



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Υπουργείο Παιδείας, Θρησκευμάτων
και Αθλητισμού



Τεχνικός Λογισμικού Η/Υ

ΕΚΠΑΙΔΕΥΤΙΚΟ ΕΓΧΕΙΡΙΔΙΟ ΣΑΕΚ

ΣΕΠΤΕΜΒΡΙΟΣ 2026

ΕΙΔΙΚΟΤΗΤΑ

Τεχνικός Λογισμικού Η/Υ

ΣΑΕΚ

ΣΕΠΤΕΜΒΡΙΟΣ 2026

ΣΥΓΓΡΑΦΗ ΕΚΠΑΙΔΕΥΤΙΚΟΥ ΕΓΧΕΙΡΙΔΙΟΥ ΣΤΗΝ ΕΙΔΙΚΟΤΗΤΑ «ΤΕΧΝΙΚΟΣ ΛΟΓΙΣΜΙΚΟΥ Η/Υ»

Συγγραφική ομάδα

Ιωάννης Δαβγιώτης

Ελένη Καλύβα

Φώτιος Λαζαρίνης

Κωνσταντίνος Παπαδάκης

Σύμβουλος μεθοδολογίας ανάπτυξης του εκπαιδευτικού εγχειριδίου

Ιωάννα Προφύρη σε συνεργασία

με Κωνσταντίνο Φασνάκη

Γλωσσική επιμέλεια

Εκδόσεις Σοκόλη

Σχεδιασμός-σελιδοποίηση

Mr. Moschos / WINK DESIGN

Το περιεχόμενο του εκπαιδευτικού εγχειριδίου της ειδικότητας διαμορφώθηκε από τη συγγραφική ομάδα με βάση μεθοδολογικές προδιαγραφές και ειδικά πρότυπα που επεξεργάστηκε το ΚΑΝΕΠ/ΓΣΕΕ στο πλαίσιο της Πράξης «ΔΙΑΜΟΡΦΩΣΗ ΟΔΗΓΩΝ ΚΑΤΑΡΤΙΣΗΣ ΚΑΙ ΕΚΠΑΙΔΕΥΤΙΚΩΝ ΕΓΧΕΙΡΙΔΙΩΝ ΙΝΣΤΙΤΟΥΤΩΝ ΕΠΑΓΓΕΛΜΑΤΙΚΗΣ ΚΑΤΑΡΤΙΣΗΣ (ΙΕΚ)» (κωδικός ΟΠΣ [MIS] 5069281) του Επιχειρησιακού Προγράμματος «Ανάπτυξη Ανθρώπινου Δυναμικού, Εκπαίδευση και Διά Βίου Μάθηση 2014-2020», που υλοποιήθηκε με σύμπραξη των κοινωνικών εταίρων και, ειδικότερα, από το ΚΑΝΕΠ/ΓΣΕΕ (επικεφαλής εταίρο της κοινοπραξίας), το ΙΝΕ/ΓΣΕΕ, το ΙΜΕ/ΓΣΕ-ΒΕΕ, το ΚΑΕΛΕ/ΕΣΕΕ, το ΙΝΣΕΤΕ και τον ΕΟΠΠΕΠ, και συγχρηματοδοτήθηκε από την Ελλάδα και την Ευρωπαϊκή Ένωση (Ε.Κ.Τ.). Το παρόν εκπαιδευτικό εγχειρίδιο αναπτύχθηκε με την ευθύνη του ΙΜΕ/ΓΣΕΒΕΕ, τα επιστημονικά στελέχη του οποίου παρακολούθησαν και υποστήριξαν τον συντονισμό του έργου σε όλες του τις φάσεις.

Περιεχόμενα

1	Πρόλογος	7
2	Εισαγωγή	9
A	Περιγραφή ειδικότητας	11
1	Τίτλος της ειδικότητας και ομάδα προσανατολισμού	11
1 . 1	Τίτλος ειδικότητας	11
1 . 2	Ομάδα προσανατολισμού (επαγγελματικός τομέας)	11
2	Συνοπτική περιγραφή ειδικότητας	12
2 . 1	Ορισμός ειδικότητας	12
2 . 2	Αρμοδιότητες/Καθήκοντα	12
2 . 3	Προοπτικές απασχόλησης στον κλάδο ή τομέα	13
2 . 4	Πρόσθετες πηγές πληροφόρησης	15
B	Σκοπός και προσδοκώμενα μαθησιακά αποτελέσματα του προγράμματος κατάρτισης της ειδικότητας	16
1	Σκοπός του προγράμματος κατάρτισης της ειδικότητας	16
2	Προσδοκώμενα μαθησιακά αποτελέσματα του προγράμματος κατάρτισης	17
Γ	Βασικές εκπαιδευτικές προδιαγραφές και αναλυτικό περιεχόμενο του προγράμματος της θεωρητικής και εργαστηριακής κατάρτισης	21
1	Ωρολόγιο πρόγραμμα	22
2	Αναλυτικό περιεχόμενο προγράμματος θεωρητικής και εργαστηριακής κατάρτισης	24
2 . 1	ΕΞΑΜΗΝΟ Α΄	24
2 . 1 . A	ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΚΑΙ ΥΛΙΚΟ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ	24
2 . 1 . B	ΣΥΓΧΡΟΝΑ ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ	29
2 . 1 . Γ	ΕΡΓΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ	35
2 . 1 . Δ	ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Ι (C11)-ΑΛΓΟΡΙΘΜΙΚΗ ΚΑΙ ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ	41
2 . 1 . E	ΕΦΑΡΜΟΓΕΣ ΓΡΑΦΕΙΟΥ ΚΑΙ ΨΗΦΙΑΚΕΣ ΔΕΞΙΟΤΗΤΕΣ ΣΤΗ ΣΥΓΧΡΟΝΗ ΑΓΟΡΑ ΕΡΓΑΣΙΑΣ	49
2 . 2	ΕΞΑΜΗΝΟ Β΄	55
2 . 2 . A	ΓΛΩΣΣΑ CLIENT-SIDE ΔΙΑΔΙΚΤΥΑΚΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ (JAVASCRIPT)	55

2 . 2 . B	ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ	61
2 . 2 . Γ	ΕΙΣΑΓΩΓΗ ΣΤΗ ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΡΥΘΜΟΝ	68
2 . 2 . Δ	ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ II (C++14)-ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ	73
2 . 2 . Ε	ΑΝΟΙΚΤΑ ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ ΒΑΣΙΣΜΕΝΑ ΣΤΟ LINUX	82
2 . 3	ΕΞΑΜΗΝΟ Γ΄	89
2 . 3 . A	ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ (PL/SQL)	89
2 . 3 . B	ΓΛΩΣΣΑ SERVER-SIDE ΔΙΑΔΙΚΤΥΑΚΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ (PHP)	94
2 . 3 . Γ	ΕΞΕΙΔΙΚΕΥΜΕΝΑ ΖΗΤΗΜΑΤΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΚΑΙ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ ΜΕ ΡΥΘΜΟΝ	100
2 . 3 . Δ	ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ III (JAVA9/JAVAFX)- ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΕΦΑΡΜΟΓΩΝ ΓΡΑΦΙΚΗΣ ΔΙΕΠΑΦΗΣ ΧΡΗΣΤΗ (GUI) I	106
2 . 3 . Ε	ΣΥΣΤΗΜΑΤΑ ΕΛΕΓΧΟΥ ΕΚΔΟΣΕΩΝ-WORKFLOWS ΚΑΙ ΣΥΝΕΡΓΑΤΙΚΕΣ ΤΕΧΝΙΚΕΣ (GIT, GITLAB, GITHUB)	111
2 . 4	ΕΞΑΜΗΝΟ Δ΄	117
2 . 4 . A	ΑΝΑΛΥΣΗ ΚΑΙ ΣΧΕΔΙΑΣΜΟΣ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ	117
2 . 4 . B	ΤΕΧΝΟΛΟΓΙΑ ΛΟΓΙΣΜΙΚΟΥ	122
2 . 4 . Γ	ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ IV (ASP.NET, MVC, C#10)- ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΕΦΑΡΜΟΓΩΝ ΓΡΑΦΙΚΗΣ ΔΙΕΠΑΦΗΣ ΧΡΗΣΤΗ (GUI) II	128
2 . 4 . Δ	ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ V (VB.NET 16, MVC)- ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΕΦΑΡΜΟΓΩΝ ΓΡΑΦΙΚΗΣ ΔΙΕΠΑΦΗΣ ΧΡΗΣΤΗ (GUI) III	135
2 . 4 . Ε	ΕΦΑΡΜΟΣΜΕΝΕΣ ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΣΤΗ ΣΥΓΧΡΟΝΗ ΑΓΟΡΑ ΕΡΓΑΣΙΑΣ	142
2 . 4 . ΣΤ	ΔΙΑΘΕΜΑΤΙΚΗ ΕΡΓΑΣΙΑ	149
3	Διδακτική μεθοδολογία	155
4	Υγεία και ασφάλεια κατά τη διάρκεια της κατάρτισης	158
4 . 1	Βασικοί κανόνες υγείας και ασφάλειας	158
4 . 2	Μέσα ατομικής προστασίας	159
Δ	Προδιαγραφές υλοποίησης πρακτικής άσκησης	160
1	Ο θεσμός της πρακτικής άσκησης	161
2	Οδηγίες για τον/την πρακτικά ασκούμενο/-η	163
2 . 1	Προϋποθέσεις εγγραφής στο πρόγραμμα πρακτικής άσκησης	163
2 . 2	Δικαιώματα και υποχρεώσεις του/της πρακτικά ασκούμενου/-ης	163
2 . 3	Φορείς υλοποίησης πρακτικής άσκησης	165

3	Ο ρόλος του/της εκπαιδευτή/-τριας της πρακτικής άσκησης	166
4	Προσδοκώμενα μαθησιακά αποτελέσματα της πρακτικής άσκησης	167
5	Αναλυτικό περιεχόμενο προγράμματος πρακτικής άσκησης	172
5 . 1 . Α	ΧΡΗΣΗ ΣΥΣΤΗΜΑΤΩΝ ΛΟΓΙΣΜΙΚΟΥ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗ ΨΗΦΙΑΚΩΝ ΤΕΧΝΟΛΟΓΙΩΝ	172
5 . 1 . Β	ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗ ΔΕΔΟΜΕΝΩΝ	173
5 . 1 . Γ	ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ ΚΑΙ ΔΙΑΔΙΚΤΥΑΚΩΝ ΕΦΑΡΜΟΓΩΝ	174
5 . 1 . Δ	ΑΝΑΛΥΣΗ ΣΥΣΤΗΜΑΤΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑ ΛΟΓΙΣΜΙΚΟΥ	175
5 . 2	Μεθοδολογία υλοποίησης πρακτικής άσκησης	176
	Βιβλιογραφία	178
A	Βιβλιογραφικές αναφορές σχετικές με την ειδικότητα	179
B	Βιβλιογραφικές αναφορές σχετικές με τη μεθοδολογία ανάπτυξης των εκπαιδευτικών εγχειριδίων	187
Γ	Σχετική εθνική νομοθεσία	189

1 | Πρόλογος

Το εκπαιδευτικό εγχειρίδιο της Ειδικότητας «Στέλεχος Επιχειρήσεων Τυποποίησης, Μεταποίησης και Εμπορίας Αγροτικών Προϊόντων» αναπτύχθηκε στο πλαίσιο της Πράξης «ΔΙΑΜΟΡΦΩΣΗ ΟΔΗΓΩΝ ΚΑΤΑΡΤΙΣΗΣ ΚΑΙ ΕΚΠΑΙΔΕΥΤΙΚΩΝ ΕΓΧΕΙΡΙΔΙΩΝ ΙΝΣΤΙΤΟΥΤΩΝ ΕΠΑΓΓΕΛΜΑΤΙΚΗΣ ΚΑΤΑΡΤΙΣΗΣ (ΙΕΚ)» (κωδικός ΟΠΣ [MIS] 5069281), η οποία υλοποιήθηκε με σύμπραξη των κοινωνικών εταίρων και, ειδικότερα, το ΚΑΝΕΠ/ΓΣΕΕ (επικεφαλής εταίρος της κοινοπραξίας), το ΙΝΕ/ΓΣΕΕ, το ΙΜΕ/ΓΣΕΒΕΕ, το ΚΑΕΛΕ/ΕΣΕΕ, το ΙΝΣΕΤΕ και τον ΕΟΠΠΕΠ, και συγχρηματοδοτήθηκε από το ΕΠ «Ανάπτυξη Ανθρώπινου Δυναμικού, Εκπαίδευση και Διά Βίου Μάθηση».

Το έργο αυτό αποτέλεσε μια ολοκληρωμένη παρέμβαση για τη βελτίωση και την ενίσχυση του θεσμού της αρχικής επαγγελματικής κατάρτισης σε μια περίοδο κατά την οποία, περισσότερο από ποτέ, το αίτημα της διασύνδεσής του με την αγορά εργασίας είναι επιτακτικό και επίκαιρο. Ιδιαίτερα, μετά τη μακρά περίοδο οικονομικής κρίσης και ύφεσης την οποία αντιμετώπισε η ελληνική κοινωνία, αλλά και τις συνέπειες από την πανδημική κρίση Covid-19, οι αναδυόμενες προκλήσεις καθιστούν αναγκαία στοχευμένα μέτρα εκσυγχρονισμού του θεσμού. Το συγκεκριμένο έργο αποτέλεσε μια συστηματική προσπάθεια αντιμετώπισης χρόνιων αδυναμιών του πεδίου, αναβάθμισης του επιπέδου των παρεχόμενων γνώσεων, δεξιοτήτων και ικανοτήτων, και βελτίωσης των μαθησιακών αποτελεσμάτων που απορρέουν από την επαγγελματική κατάρτιση σε συγκεκριμένες ειδικότητες.

Εμπερικλείοντας μια καινοτόμο δέσμη αλληλοσυμπληρούμενων δράσεων, μεθόδων και πρακτικών, επιδίωξε να συμβάλει με πολλαπλασιαστικό τρόπο στην ενίσχυση της ποιότητας και της αποτελεσματικότητας του πεδίου της αρχικής επαγγελματικής κατάρτισης. Ειδικότερα, στο πλαίσιο του έργου:

- Διεξήχθη ποιοτική έρευνα με στόχο τη διερεύνηση των χαρακτηριστικών και της δυναμικής που διέπει το πεδίο της αρχικής επαγγελματικής κατάρτισης στη χώρα μας και στον ευρωπαϊκό χώρο, με στόχο τη διαμόρφωση σχετικών προτάσεων πολιτικής.
- Αναπτύχθηκαν:
 - Επικαιροποιημένοι «οδηγοί κατάρτισης» για 130 ειδικότητες αρχικής κατάρτισης,
 - Αντίστοιχα εκπαιδευτικά εγχειρίδια, για την υποστήριξη της κατάρτισης/εκπαίδευσης των σπουδαστών/-τριών,
 - Συναφείς τράπεζες θεμάτων για κάθε ειδικότητα.
 - Συναφείς τράπεζες θεμάτων για κάθε ειδικότητα.

- Το σύνολο των παραπάνω στηρίχθηκε σε ένα ενιαίο μεθοδολογικό πλαίσιο, μέσω του οποίου επιδιώχθηκε η σύνδεση της κοινωνικής εμπειρίας της εργασίας, της εκπαίδευσης και της πιστοποίησής της, λαμβάνοντας υπόψη του το ισχύον θεσμικό πλαίσιο, καθώς και τις ιδιαιτερότητες του πεδίου της αρχικής επαγγελματικής κατάρτισης.
- Τέλος, με γνώμονα την ενίσχυση της θετικής επενέργειας του έργου σε θεσμικό επίπεδο, αναπτύχθηκε μια μεθοδολογία ευέλικτης περιοδικής επανεξέτασης και επικαιροποίησης των περιεχομένων των οδηγών κατάρτισης, των εγχειριδίων και των τραπεζών θεμάτων, έτσι ώστε αυτά να βρίσκονται –κατά το δυνατόν– σε αντιστοιχία με τα νέα τεχνολογικά, οργανωσιακά, εργασιακά, περιβαλλοντικά, κοινωνικά και οικονομικά δεδομένα και τις ανάγκες της αγοράς εργασίας και των εκπαιδευομένων.

2 | Εισαγωγή

Το παρόν εγχειρίδιο, εκκινώντας από την περιγραφή της ειδικότητας, εστιάζεται στο αναλυτικό περιεχόμενο και τις βασικές εκπαιδευτικές προδιαγραφές ενός προγράμματος κατάρτισης στην ειδικότητα «Τεχνικός Λογισμικού Η/Υ». Στοχεύει στην ενημέρωση του συνόλου των συντελεστών του, λαμβάνοντας υπόψη τα περιεχόμενα των καθηκόντων και τις ιδιαιτερότητές της, καθώς και τους ισχύοντες θεσμικούς περιορισμούς στο πεδίο.

Απευθύνεται κυρίως στους/στις εκπαιδευόμενους/-ες, αλλά και στους/στις εκπαιδευτές/-τριες των προγραμμάτων αρχικής επαγγελματικής κατάρτισης. Επιπλέον, αποτελεί ένα χρήσιμο εγχειρίδιο για τα στελέχη σχεδιασμού, τους σχετικούς φορείς υλοποίησής τους, τις Σχολές Ανώτερης Επαγγελματικής Κατάρτισης, αλλά και για το σύνολο των υπόλοιπων δυνάμει συντελεστών ενός προγράμματος αρχικής επαγγελματικής κατάρτισης, ιδιαίτερα για όσους/-ες συμμετέχουν στην υλοποίηση της πρακτικής άσκησης.

Το εγχειρίδιο αυτό αποτελεί μια συστηματική βάση, η οποία περιλαμβάνει βασικά στοιχεία για την κατανόηση του πεδίου της συγκεκριμένης ειδικότητας, αλλά κυρίως επικεντρώνεται στην ανάλυση των μαθησιακών ενοτήτων και των εκπαιδευτικών προδιαγραφών της θεωρητικής και της εργαστηριακής κατάρτισης, καθώς και στην υλοποίηση της πρακτικής άσκησης.

Στην κατεύθυνση αυτή, για το κάθε πρόγραμμα αρχικής επαγγελματικής κατάρτισης το οποίο δύναται να υλοποιηθεί, είναι απαραίτητο να ληφθούν συστηματικά υπόψη τα εκπαιδευτικά περιεχόμενα αλλά και οι μεθοδολογικές προδιαγραφές που περιλαμβάνονται.

Ειδικότερα, το εκπαιδευτικό εγχειρίδιο αποτελείται από τέσσερα (Α'-Δ') Μέρη.

- **Το Μέρος Α'** παρέχει συνοπτικά τις πληροφορίες που αφορούν την περιγραφή της ειδικότητας.

Περιλαμβάνει την περιγραφή της ειδικότητας, των κύριων εργασιακών καθηκόντων της, των προοπτικών απασχόλησης σε αυτήν, ενώ γίνεται αναφορά σε βασικές πηγές πληροφόρησης σε σχέση με την ειδικότητα.

- **Το Μέρος Β'** εστιάζεται στον καθορισμό των ευρύτερων αλλά και των επιμέρους ενοτήτων των προσδοκώμενων μαθησιακών αποτελεσμάτων του προγράμματος κατάρτισης.

Αναφέρεται στις δραστηριότητες που θα είναι σε θέση να επιτελέσουν οι εκπαιδευόμενοι/-ες μετά το πέρας της συνολικής κατάρτισής τους στη συγκεκριμένη ειδικότητα.

- **Το Μέρος Γ΄ εστιάζεται στο περιεχόμενο και τη διάρθρωση του προγράμματος θεωρητικής και εργαστηριακής κατάρτισης, όπως και σε βασικές εκπαιδευτικές προδιαγραφές υλοποίησής του.**

Το Μέρος Γ΄ περιλαμβάνει το ωρολόγιο πρόγραμμα, την περίληψη, τα προσδοκώμενα μαθησιακά αποτελέσματα, τις ώρες διδασκαλίας ανά εβδομάδα, τις βασικές έννοιες και λέξεις-κλειδιά, ένα κείμενο αναφοράς και τις προτεινόμενες πηγές μελέτης κάθε μαθησιακής ενότητας. Επιπλέον, αναφέρεται στην προτεινόμενη διδακτική μεθοδολογία, καθώς και σε σημαντικές πληροφορίες που αφορούν την υγεία και την ασφάλεια κατά τη διάρκεια της κατάρτισης.

- **Το Μέρος Δ΄ εστιάζεται στην περιγραφή του περιεχομένου, των χαρακτηριστικών και των προδιαγραφών υλοποίησης της πρακτικής άσκησης.**

Περιλαμβάνεται η περιγραφή του θεσμού της πρακτικής άσκησης και παρέχονται χρήσιμες οδηγίες για τους/τις εκπαιδευόμενους/-ες και τους/τις εκπαιδευτές/-τριες στον χώρο εργασίας. Στα περιεχόμενα συγκαταλέγονται επίσης οι ενότητες μαθησιακών αποτελεσμάτων της πρακτικής άσκησης, οι ενδεικτικές εργασίες τις οποίες θα εκτελέσουν οι εκπαιδευόμενοι/-ες, και οι ειδικές προδιαγραφές υλοποίησης που αφορούν τον εξοπλισμό. Αναλύονται οι ενότητες μαθησιακών αποτελεσμάτων, ενώ προτείνονται βασικές μεθοδολογικές αρχές για την υλοποίηση της πρακτικής άσκησης.

Το εκπαιδευτικό εγχειρίδιο και η τράπεζα θεμάτων της ειδικότητας συνδέονται οργανικά με τον αντίστοιχο οδηγό κατάρτισης της ειδικότητας.¹

Για τη συγγραφή της μεθοδολογίας ανάπτυξης των εκπαιδευτικών εγχειριδίων συνεργάστηκαν οι Ρένα Βαρβιτσιώτη, Ελένη Θεοδωρή, Κωνσταντίνος Μαρκίδης και Δέσποινα Μπαμπανέλου, η οποία ανέλαβε και τη συντακτική της επιμέλεια. Αυτή εκπονήθηκε στο πλαίσιο της «Οριζόντιας Εκπαιδευτικής Μεθοδολογίας» της εν λόγω Πράξης «ΔΙΑΜΟΡΦΩΣΗ ΟΔΗΓΩΝ ΚΑΤΑΡΤΙΣΗΣ ΚΑΙ ΕΚΠΑΙΔΕΥΤΙΚΩΝ ΕΓΧΕΙΡΙΔΙΩΝ ΙΝΣΤΙΤΟΥΤΩΝ ΕΠΑΓΓΕΛΜΑΤΙΚΗΣ ΚΑΤΑΡΤΙΣΗΣ (ΙΕΚ)» και με βάση μια κατάλληλα προσαρμοσμένη εκδοχή του προτύπου ανάπτυξης εκπαιδευτικών υλικών του ΙΝΕ/ΓΣΕΕ (Γούλας, Μαρκίδης & Μπαμπανέλου, 2021).

¹ Δαβγιώτης, Ι., Λαζαρίνης, Φ. & Παπαδάκης, Κ. (2023). *Οδηγός κατάρτισης «Τεχνικός Λογισμικού Η/Υ»*. Αθήνα: Γενική Γραμματεία Επαγγελματικής Εκπαίδευσης, Κατάρτισης, Διά Βίου Μάθησης και Νεολαίας.

A

Περιγραφή ειδικότητας

1 | Τίτλος της ειδικότητας και ομάδα προσανατολισμού

1.1 • Τίτλος ειδικότητας

«Τεχνικός Λογισμικού Η/Υ».²

1.2 • Ομάδα προσανατολισμού (επαγγελματικός τομέας)

Η ειδικότητα «Τεχνικός Λογισμικού Η/Υ» ανήκει στην ομάδα προσανατολισμού «Τεχνολογικών Εφαρμογών» και στον Τομέα «Πληροφορικής».

2 ΦΕΚ Α΄ 193/17-9-2013: https://www.minedu.gov.gr/publications/docs2018/N_4186_2013_fek193.pdf

2 | Συνοπτική περιγραφή ειδικότητας

2.1 • Ορισμός ειδικότητας

Οι απόφοιτοι της ειδικότητας «Τεχνικός Λογισμικού Η/Υ» συμμετέχουν με καίριο ρόλο στη βιομηχανία/αγορά Πληροφορικής, έχοντας ως βασικό επαγγελματικό αντικείμενο τη σχεδίαση, την ανάπτυξη και τη διαχείριση συστημάτων Πληροφορικής. Σκοπός της ειδικότητας «Τεχνικός Λογισμικού Η/Υ» είναι να μεταδώσει στους/στις σπουδαστές/-τριες τις γενικές αρχές της επιστήμης και να τους βοηθήσει να κατανοήσουν και να αφομοιώσουν στο μέλλον νέες τεχνολογίες, εργαλεία και συστήματα. Οι απόφοιτοι της ειδικότητας θα ασχοληθούν κυρίως με τη σχεδίαση, την ανάπτυξη, την υποστήριξη λειτουργίας και τη συντήρηση αυτόνομων ή διαδικτυακών εφαρμογών, χρησιμοποιώντας τις δημοφιλέστερες γλώσσες προγραμματισμού και μια πληθώρα εργαλείων διαχείρισης, ελέγχου και τεκμηρίωσης. Οι εφαρμογές θα καλύπτουν κάθε τομέα της καθημερινής ζωής, καθώς η ανάπτυξη λογισμικού αποτελεί μια εξειδίκευση της Πληροφορικής που σημειώνει αλματώδη αύξηση τα τελευταία χρόνια. Το λογισμικό εφαρμογών έχει διεισδύσει σε ένα ευρύ φάσμα δραστηριοτήτων, καθιστώντας το επάγγελμα του «Τεχνικού Λογισμικού Η/Υ» πρόκληση για το μέλλον.

2.2 • Αρμοδιότητες/Καθήκοντα

Ο/Η κάτοχος διπλώματος της ειδικότητας «Τεχνικός Λογισμικού Η/Υ» ασχολείται με την ανάλυση, τη σχεδίαση και την υλοποίηση λογισμικού που:

- Αναπτύσσει, σύμφωνα με συγκεκριμένες προδιαγραφές και με βάση την προστασία των προσωπικών δεδομένων και την ασφάλεια της πληροφορίας,
- Χρησιμοποιείται για την επίλυση προβλημάτων της πραγματικής ζωής σε διάφορα πεδία της ανθρώπινης δραστηριότητας.

Ο/Η «Τεχνικός Λογισμικού Η/Υ» ασκεί (ενδεικτικά και όχι περιοριστικά) τις παρακάτω αρμοδιότητες/καθήκοντα:

- Κατανόηση των αναγκών μιας επιχείρησης σε υποδομές και υπηρεσίες σε ΤΠΕ και αξιολόγηση διαφόρων συνθέσεων υλικού και λογισμικού, σύμφωνα με τις κατάλληλες προδιαγραφές,
- Εγκατάσταση, ρύθμιση, έλεγχο, συντήρηση, χρήση εφαρμογών λογισμικού Η/Υ,
- Εκτέλεση τεχνικών εργασιών, διατήρηση αντιγράφων ασφάλειας και επαναφορά δεδομένων επί του λογισμικού του πληροφοριακού συστήματος ή των ανεξάρτητων Η/Υ,
- Διάγνωση, αναγνώριση και επίλυση προβλημάτων σε συστήματα λογισμικού,
- Εγκατάσταση και διαμόρφωση κατάλληλου λογισμικού και λειτουργιών σύμφωνα με τις προδιαγραφές,

- Υποστήριξη χρηστών και παροχή καθοδήγησης σε αυτούς, σχετικά με τον τρόπο λειτουργίας νέου λογισμικού,
- Αναβάθμιση και επικαιροποίηση εφαρμογών λογισμικού και υποστήριξη των αντίστοιχων χρηστών των εφαρμογών λογισμικού-δικτύων, βάσει υφιστάμενων μελετών και διαδικασιών,
- Επέκταση και βελτιστοποίηση των εργαλείων και δυνατοτήτων του περιβάλλοντος διεπαφής χρήστη (user interface),
- Αξιολόγηση σε επίπεδο εφαρμογής-υλοποίησης-συντήρησης εφαρμογών λογισμικού, εφαρμόζοντας υφιστάμενες αναλύσεις και σχέδια ανάπτυξης,
- Επικοινωνία και ανταλλαγή δεδομένων με άλλες εφαρμογές Πληροφορικής,
- Ανάλυση, σχεδιασμό και ανάπτυξη διαδικτυακών εφαρμογών,
- Σχεδιασμό και ανάπτυξη συστημάτων λογισμικού με χρήση εργαλείων και βοηθητικών προγραμμάτων, διαδραστικών γραφικών και γλωσσών προγραμματισμού, κατανοώντας τις λειτουργικές απαιτήσεις που προέρχονται από τις οδηγίες του αναλυτή,
- Δημιουργία και σύνδεση βάσεων δεδομένων με το περιβάλλον προγραμματισμού και τις διαδικτυακές εφαρμογές,
- Διαχείριση συστήματος βάσης δεδομένων των πληροφοριακών συστημάτων,
- Σχεδιασμό, ανάπτυξη και ενσωμάτωση κώδικα υπολογιστή με άλλα εξειδικευμένα δεδομένα, για την παραγωγή, τη συντήρηση και την υποστήριξη ιστοτόπων,
- Σύνταξη μελετών, εκθέσεων, οδηγιών υλοποίησης και τεκμηρίωσης, σχετικών με τις ΤΠΕ,
- Βοήθεια στην ανάλυση, τον καθορισμό και την ανάπτυξη στρατηγικών Διαδικτύου και μεθοδολογιών που βασίζονται στο Διαδίκτυο και σχεδίων ανάπτυξης,
- Προώθηση και πώληση προϊόντων ή υπηρεσιών Πληροφορικής (Hardware, Software, προϊόντα ψηφιακών τεχνολογιών).

2.3 • Προοπτικές απασχόλησης στον κλάδο ή τομέα

Με την πανδημία να έχει δημιουργήσει νέα μοντέλα εργασίας, οι επιχειρήσεις μετατοπίζουν το ενδιαφέρον τους από το Hardware στο Software και στο Cloud. Μάλιστα, το 2021, το λογισμικό ήταν η δεύτερη μεγαλύτερη δαπάνη με μερίδιο 29%.

Η ζήτηση για τις υπηρεσίες Πληροφορικής και για το λογισμικό εξαρτάται πλέον, σε μεγάλο βαθμό, από την ανάγκη των επιχειρήσεων-πελατών του κλάδου να μετασχηματιστούν ψηφιακά όσο το δυνατόν γρηγορότερα, ώστε να καταφέρουν να παραμείνουν ανταγωνιστικές. Η τεχνολογική ανάπτυξη και ο ψηφιακός μετασχηματισμός φέρνουν νέα δεδομένα για την εργασία, δημιουργώντας νέες προκλήσεις για τους/τις εργαζομένους/-ες, καθώς, σύμφωνα με έκθεση που συνέταξε το Παρατηρητήριο Ψηφιακού Μετασχηματισμού του ΣΕΒ, με τη συνεργασία της Deloitte, περισσότερες από 85 εκατομμύρια θέσεις εργασίας θα μετεξελιχθούν μέχρι το 2025. Περίπου τέσσερις στις δέκα θέσεις εργασίας στην ΕΕ αναμένεται να μετασχηματιστούν εξαιτίας των νέων τεχνολογιών και παγκοσμίως το 14% θα εξαλειφθούν, ενώ θα διαταραχθεί το 32% των θέσεων εργασίας παγκοσμίως.

Επισημαίνεται πως το 85% των θέσεων εργασίας που θα υπάρχουν το 2030 δεν έχουν δημιουργηθεί ακόμη, με 97 εκατομμύρια θέσεις εργασίας που θα δημιουργηθούν να είναι τελείως καινούριες σε σχέση με αυτές που γνωρίζουμε σήμερα. Οι ψηφιακές τεχνολογίες δημιουργούν νέα επαγγέλματα, διαφοροποιώντας τις ανάγκες που απαιτούνται από τους/τις εργαζομένους/-ες, των οποίων η γνώση και οι δεξιότητες εκτιμάται πως απαξιώνονται κατά 30% κάθε χρόνο³.

Σύμφωνα με έρευνες, τα «υπερ-επαγγέλματα» που συνδυάζουν τη χρήση τεχνολογίας, π.χ. τη γνώση προγραμματισμού και ανάλυσης δεδομένων, μαζί με ένα ευρύ φάσμα κοινωνικών δεξιοτήτων και ικανοτήτων, διατρέχουν τον μικρότερο κίνδυνο για να χαθούν από την αυτοματοποίηση.

Ο/Η κάτοχος διπλώματος της ειδικότητας «Τεχνικός Λογισμικού Η/Υ» έχει πολύ θετικές προοπτικές απασχόλησης και μπορεί να εργαστεί ενδεικτικά στους ακόλουθους οργανισμούς:

- Εταιρείες ανάπτυξης, υποστήριξης και συντήρησης εφαρμογών λογισμικού,
- Εταιρείες τεχνικής υποστήριξης Υπολογιστικών Συστημάτων,
- Εταιρείες παροχής υπηρεσιών Ασφάλειας Πληροφοριακών Συστημάτων, Διαχείρισης Συστημάτων ή Βάσεων Δεδομένων,
- Εταιρείες κινητής τηλεφωνίας,
- Εταιρείες παραγωγής τηλεπικοινωνιακού υλικού και λογισμικού,
- Εταιρείες εγκατάστασης και συντήρησης υλικού υπολογιστών,
- Εταιρείες παροχής υπηρεσιών Διαδικτύου,
- Internet Providers,
- Τμήματα Πληροφορικής και μηχανογραφικά κέντρα οργανισμών δημόσιου τομέα και επιχειρήσεων ιδιωτικού τομέα,
- Επιχειρήσεις κατασκευής ή υποστήριξης προϊόντων Πληροφορικής,
- Επιχειρήσεις παροχής υπηρεσιών φιλοξενίας ιστοτόπων,
- Εταιρείες κατασκευής ιστοτόπων,
- Επιχειρήσεις προβολής εταιρειών στο Διαδίκτυο,
- Διαφημιστικές εταιρείες,
- Επιχειρήσεις παροχής υπηρεσιών νεφοϋπολογιστικής,
- Επιχειρήσεις ηλεκτρονικού εμπορίου (e-shop),
- Επιχειρήσεις προώθησης προϊόντων και υπηρεσιών Πληροφορικής,
- Επιχειρήσεις συμβουλευτικής στην επιστήμη των υπολογιστών,
- Εκπαιδευτικούς οργανισμούς,
- Υπηρεσίες του Δημόσιου Τομέα που χρησιμοποιούν κέντρο Πληροφορικής.

3 https://www.businessdaily.gr/oikonomia/44093_dimiorygoyn-57-ekat-theseis-ergasias-oiependyseis-stin-tehnologia

Τέλος, ο/η κάτοχος διπλώματος της ειδικότητας «Τεχνικός Λογισμικού Η/Υ» μπορεί να εργαστεί, δημιουργώντας τη δική του επιχείρηση, αυτόνομα ή ως μέλος κάποιας ομάδας, η οποία έχει ένα ευρύτερο αντικείμενο απασχόλησης στον χώρο της Πληροφορικής.

2.4 • Πρόσθετες πηγές πληροφόρησης

Ηλεκτρονικές διευθύνσεις συναφών οργανώσεων εργαζομένων, επαγγελματικών οργανώσεων και οργανισμών:

- Εθνικός Οργανισμός Πιστοποίησης Προσόντων και Επαγγελματικού
- Προσανατολισμού (ΕΟΠΠΕΠ): <https://www.eoppep.gr/>
- Κατάλογος Επαγγελματικών Περιγραμμάτων ΕΟΠΠΕΠ:
<https://ergonesti.eoppep.gr/>
- ΙΕΚ-ΟΑΕΔ-Αρχική Επαγγελματική Κατάρτιση:
- <https://www.oaed.gr/iek-oaed-arkhiki-epaghghematiki-katartisi>
- Ελληνική Εταιρεία Επιστημόνων και Επαγγελματιών Πληροφορικής
- και Επικοινωνιών ΕΠΥ: <http://www.epy.gr>
- Ένωση Πληροφορικών Ελλάδος: <https://www.epe.org.gr/>
- Πανελλήνιος Σύνδεσμος Επιχειρήσεων Ηλεκτρονικών Εφαρμογών,
- Πληροφορικής και Νέων Τεχνολογιών: <https://psict.gr/>
- Ελληνικό Δίκτυο Επαγγελματιών Πληροφορικής ΗΕΠΙΣ: <https://www.hepis.gr>
- Εταιρεία Ελεύθερου Λογισμικού/Λογισμικού Ανοικτού Κώδικα: <https://ellak.gr/>

B

Σκοπός και προσδοκώμενα μαθησιακά αποτελέσματα του προγράμματος κατάρτισης της ειδικότητας

1 | Σκοπός του προγράμματος κατάρτισης της ειδικότητας

Ο βασικός σκοπός του προγράμματος κατάρτισης της ειδικότητας είναι να προετοιμάσει τους/τις εκπαιδευόμενους/-ες για την επαγγελματική τους σταδιοδρομία στην ειδικότητα «Τεχνικός Λογισμικού Η/Υ».

2 | Προσδοκώμενα μαθησιακά αποτελέσματα του προγράμματος κατάρτισης

Οι γνώσεις, οι δεξιότητες και οι ικανότητες που θα αποκτήσουν οι εκπαιδευόμενοι/-ες κατά τη διάρκεια της κατάρτισής τους, οργανώνονται σε Ενότητες προσδοκώμενων μαθησιακών αποτελεσμάτων.

Στον πίνακα που ακολουθεί, παρουσιάζονται τα επιμέρους προσδοκώμενα μαθησιακά αποτελέσματα ανά ενότητα.

Πίνακας 1 /// Ενότητες προσδοκώμενων μαθησιακών αποτελεσμάτων

ΕΝΟΤΗΤΕΣ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	
ΕΝΟΤΗΤΑ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	Επιμέρους προσδοκώμενα μαθησιακά αποτελέσματα Με την ολοκλήρωση του προγράμματος κατάρτισης, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:
Α. «Χρήση συστημάτων λογισμικού και διαχείριση ψηφιακών τεχνολογιών»	• Αναγνωρίζουν τη δομή του υπολογιστή και τη λειτουργικότητα των διαφόρων υποσυστημάτων του, • Αξιολογούν διάφορες συνθέσεις υλικού και λογισμικού σύμφωνα με τις κατάλληλες προδιαγραφές, • Εξηγούν τις δυνατότητες του κοινού λογισμικού για εφαρμογές γραφείου και βοηθητικών εργαλείων λογισμικού, • Διαχειρίζονται διαφορετικά λειτουργικά συστήματα και λογισμικό εφαρμογών γραφείου, • Αναπτύσσουν τρόπους βέλτιστης παραμετροποίησης των λειτουργικών συστημάτων, • Εγκαθιστούν και αναβαθμίζουν λογισμικό συστημάτων (system Software) και περιφερειακές συσκευές, • Εγκαθιστούν και συντηρούν λογισμικό εφαρμογών σύμφωνα με τις οδηγίες των κατασκευαστών του, • Διατηρούν αντίγραφα ασφάλειας (backup) και επαναφέρουν (restore) δεδομένα από αντίγραφα ασφάλειας, θέτοντας σε κανονική κατάσταση λειτουργίας εγκαταστάσεις εφαρμογών, • Εκτελούν διαγνωστικούς ελέγχους σε εγκαταστάσεις υπολογιστών και αναγνωρίζουν τις αιτίες των βλαβών, • Χρησιμοποιούν υπηρεσίες νεφούπολογιστικής, • Επιλέγουν τα κατάλληλα μέτρα για τη φυσική και λογική προστασία των δεδομένων, • Αναγνωρίζουν τις ενδεχόμενες απειλές (ιοί, hackers) και να λαμβάνουν τα απαραίτητα μέτρα για την προστασία των συστημάτων λογισμικού, • Συντάσσουν μελέτες, εκθέσεις, οδηγούς υλοποίησης και τεκμηρίωσης, σχετικούς με τις ΤΠΕ.

συνέχεια >>

Πίνακας 1 /// Ενότητες προσδοκώμενων μαθησιακών αποτελεσμάτων

ΕΝΟΤΗΤΕΣ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	
ΕΝΟΤΗΤΑ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	Επιμέρους προσδοκώμενα μαθησιακά αποτελέσματα Με την ολοκλήρωση του προγράμματος κατάρτισης, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:
Β. «Προγραμματισμός και διαχείριση δεδομένων»	• Αποκτούν αντίληψη των διαφορετικών στιλ προγραμματισμού (διαδικαστικός, αντικειμενοστρεφής, συναρτησιακός), • Παραθέτουν αλγοριθμικές λύσεις για την επίλυση ενός προβλήματος, • Χρησιμοποιούν τις κυριότερες γλώσσες προγραμματισμού, • Σχεδιάζουν προγραμματιστικές λύσεις για επίλυση πραγματικών προβλημάτων, • Εκσφαλματώνουν προγράμματα που εμφανίζουν συντακτικά ή λογικά σφάλματα, • Βελτιστοποιούν τον κώδικα των προγραμμάτων για την καλύτερη απόδοσή τους, • Εφαρμόζουν τεχνικές αμυντικού προγραμματισμού, • Επεξεργάζονται δεδομένα με χρήση βάσεων δεδομένων, μεριμνώντας για την ασφάλειά τους και να εξαγουν χρήσιμα συμπεράσματα, • Εισάγουν περιεχόμενο μεταξύ διαφορετικών συστημάτων λογισμικού, εξασφαλίζοντας την ομαλή μετάπτωση των δεδομένων.
Γ. «Ανάπτυξη λογισμικού και διαδικτυακών εφαρμογών»	• Κατηγοριοποιούν τις τεχνολογίες ανάπτυξης λογισμικού συστήματος, • Προβλέπουν τους κινδύνους που ενδέχεται να προκύψουν σε ένα έργο ανάπτυξης λογισμικού, • Αναπτύσσουν διαδικτυακές εφαρμογές με χρήση HTML5 και CSS3, • Υλοποιούν διαδικτυακές εφαρμογές, χρησιμοποιώντας τεχνολογίες από την πλευρά του/της πελάτη/-ισσας (client-side), • Αναπτύσσουν δυναμικές εφαρμογές του Παγκόσμιου Ιστού με τεχνολογίες από την πλευρά του διακομιστή (server-side) και βάσεις δεδομένων, • Σχεδιάζουν γραφικά περιβάλλοντα με βάση την προστασία των προσωπικών δεδομένων και την ασφάλεια της πληροφορίας (information security), • Τεκμηριώνουν τη λειτουργικότητα μιας εφαρμογής, • Χρησιμοποιούν τοπικά και απομακρυσμένα συστήματα ελέγχου εκδόσεων για ανάπτυξη λογισμικού.

Πίνακας 1 /// Ενότητες προσδοκώμενων μαθησιακών αποτελεσμάτων

ΕΝΟΤΗΤΕΣ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	
ΕΝΟΤΗΤΑ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	Επιμέρους προσδοκώμενα μαθησιακά αποτελέσματα Με την ολοκλήρωση του προγράμματος κατάρτισης, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:
Δ. «Ανάλυση συστημάτων και τεχνολογία λογισμικού»	- Αναγνωρίζουν τις βασικές έννοιες της τεχνολογίας λογισμικού (λογισμικό, εργαλεία, μεθοδολογίες, κύκλο ζωής κ.λπ.), - Περιγράφουν τα στάδια του κύκλου ανάπτυξης και ζωής μιας εφαρμογής λογισμικού, - Αναγνωρίζουν τις απαιτήσεις μιας εφαρμογής λογισμικού, χρησιμοποιώντας τις αρχές της δομημένης ανάλυσης, - Αναπαριστούν τις απαιτήσεις πελατών/-τισσών, χρησιμοποιώντας εργαλεία και μεθοδολογίες ανάλυσης και σχεδίασης, - Παραθέτουν τους ελέγχους που διενεργούνται σε κάθε επίπεδο ελέγχου κώδικα λογισμικού (levels of testing), - Εξηγούν τις διαφορετικές τεχνικές που χρησιμοποιούνται για τον έλεγχο ενός συστήματος λογισμικού, - Χρησιμοποιούν εργαλεία ελέγχου ενός συστήματος λογισμικού, - Ελέγχουν την ποιότητα ενός συστήματος λογισμικού, - Σχεδιάζουν βάσεις δεδομένων, χρησιμοποιώντας αποτελεσματικά τις τεχνικές του σημασιολογικού, λογικού και φυσικού σχεδιασμού, - Προσαρμόζουν τον κώδικα και τη δομή των βάσεων δεδομένων, ανάλογα με τις προδιαγραφές, - Ενσωματώνουν βάσεις δεδομένων σε εφαρμογές.

συνέχεια »

Πίνακας 1 /// Ενότητες προσδοκώμενων μαθησιακών αποτελεσμάτων

ΕΝΟΤΗΤΕΣ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	
ΕΝΟΤΗΤΑ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	Επιμέρους προσδοκώμενα μαθησιακά αποτελέσματα Με την ολοκλήρωση του προγράμματος κατάρτισης, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:
Ε. «Οικονομική διαχείριση και μάρκετινγκ επιχειρήσεων μεταποίησης και τυποποίησης αγροτικών προϊόντων»	• Επιλέγουν το όνομα της επιχείρησης και ονόματα προϊόντων, μελετώντας τις σχετικές νομικές οδηγίες ειδικών, • Επιλέγουν σχέδια ετικετών σήμανσης τροφίμων, μελετώντας βασικές αρχές νομοθεσίας, • Επιλέγουν σύγχρονα συστήματα λογιστικής διαχείρισης, μελετώντας τις ανάγκες της επιχείρησης, • Περιγράφουν βασικά οικονομικά μεγέθη, • Εκτιμούν τη βιωσιμότητα της επιχείρησης, μελετώντας επιχειρηματικά σχέδια, • Διαχειρίζονται μεθόδους αντιμετώπισης επιχειρηματικών κινδύνων, • Αναλύουν οικονομικά δεδομένα σε συνεργασία με ειδικούς, • Εκμεταλλεύονται ορθολογικά αναπτυξιακές επιδοτήσεις, • Διακρίνουν επιχειρηματικούς κινδύνους, • Αναλύουν οικονομικά στοιχεία και κοστολογούν προϊόντα.



Βασικές εκπαιδευτικές προδιαγραφές και αναλυτικό περιεχόμενο του προγράμματος της θεωρητικής και εργαστηριακής κατάρτισης

1 | Ωρολόγιο πρόγραμμα

Παρατίθεται το ωρολόγιο πρόγραμμα της ειδικότητας «Τεχνικός Λογισμικού Η/Υ», με παρουσίαση των εβδομαδιαίων ωρών θεωρίας (Θ), εργαστηρίων (Ε), καθώς και του συνόλου (Σ) αυτών ανά μάθημα και ανά εξάμηνο:

Πίνακας 2 /// Ωρολόγιο πρόγραμμα

ΕΞΑΜΗΝΟ		Α			Β			Γ			Δ		
Α/Α	ΜΑΘΗΣΙΑΚΕΣ ΕΝΟΤΗΤΕΣ	Θ	Ε	Σ	Θ	Ε	Σ	Θ	Ε	Σ	Θ	Ε	Σ
1	ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΚΑΙ ΥΛΙΚΟ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ	1	1	2									
2	ΣΥΓΧΡΟΝΑ ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ		3	3									
3	ΕΡΓΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ (HTML5, CSS3)		5	5									
4	ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Ι (C11)-ΑΛΓΟΡΙΘΜΙΚΗ ΚΑΙ ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ	1	5	6									
5	ΕΦΑΡΜΟΓΕΣ ΓΡΑΦΕΙΟΥ ΚΑΙ ΨΗΦΙΑΚΕΣ ΔΕΞΙΟΤΗΤΕΣ ΣΤΗ ΣΥΓΧΡΟΝΗ ΑΓΟΡΑ ΕΡΓΑΣΙΑΣ		4	4									
6	ΓΛΩΣΣΑ CLIENT-SIDE ΔΙΑΔΙΚΤΥΑΚΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ (JAVASCRIPT)					4	4						
7	ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ					4	4						
8	ΕΙΣΑΓΩΓΗ ΣΤΗ ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΡΥΤΗΘΝ					3	3						
9	ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΙΙ (C++14)-ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ					6	6						
10	ΑΝΟΙΚΤΑ ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ ΒΑΣΙΣΜΕΝΑ ΣΤΟ LINUX					3	3						
11	ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ (PL/SQL)								4	4			
12	ΓΛΩΣΣΑ SERVER-SIDE ΔΙΑΔΙΚΤΥΑΚΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ (PHP)								4	4			

[συνέχεια >>](#)

Πίνακας 2 /// Ωρολόγιο πρόγραμμα

ΕΞΑΜΗΝΟ		Α			Β			Γ			Δ		
Α/Α	ΜΑΘΗΣΙΑΚΕΣ ΕΝΟΤΗΤΕΣ	Θ	Ε	Σ	Θ	Ε	Σ	Θ	Ε	Σ	Θ	Ε	Σ
13	ΕΞΕΙΔΙΚΕΥΜΕΝΑ ΖΗΤΗΜΑΤΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΚΑΙ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ ΜΕ ΡΥΘΜΟΝ								4	4			
14	ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ III (JAVA9/JAVAFX)- ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΕΦΑΡΜΟΓΩΝ ΓΡΑΦΙΚΗΣ ΔΙΕΠΑΦΗΣ ΧΡΗΣΤΗ (GUI) I								6	6			
15	ΣΥΣΤΗΜΑΤΑ ΕΛΕΓΧΟΥ ΕΚΔΟΣΕΩΝ, WORKFLOWS ΚΑΙ ΣΥΝΕΡΓΑΤΙΚΕΣ ΤΕΧΝΙΚΕΣ (GIT, GITLAB, GITHUB)								2	2			
16	ΑΝΑΛΥΣΗ ΚΑΙ ΣΧΕΔΙΑΣΜΟΣ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ											3	3
17	ΤΕΧΝΟΛΟΓΙΑ ΛΟΓΙΣΜΙΚΟΥ										2	2	4
18	ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ IV (ASP.NET, MVC, C#10)- ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΕΦΑΡΜΟΓΩΝ ΓΡΑΦΙΚΗΣ ΔΙΕΠΑΦΗΣ ΧΡΗΣΤΗ (GUI) II											4	4
19	ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ V (VB.NET 16, MVC)- ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΕΦΑΡΜΟΓΩΝ ΓΡΑΦΙΚΗΣ ΔΙΕΠΑΦΗΣ ΧΡΗΣΤΗ (GUI) III											5	5
20	ΕΦΑΡΜΟΓΕΣ ΓΛΩΣΣΩΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΣΤΗ ΣΥΓΧΡΟΝΗ ΑΓΟΡΑ ΕΡΓΑΣΙΑΣ											2	2
21	ΔΙΑΘΕΜΑΤΙΚΗ ΕΡΓΑΣΙΑ											2	2
ΣΥΝΟΛΟ		2	18	20	-	20	20	2	18	20	0	20	20

2 | Αναλυτικό περιεχόμενο προγράμματος θεωρητικής και εργαστηριακής κατάρτισης

2.1 • ΕΞΑΜΗΝΟ Α΄

2.1.A. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΚΑΙ ΥΛΙΚΟ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Αρχιτεκτονική και υλικό ηλεκτρονικών υπολογιστών» εισάγει τους/τις εκπαιδευόμενους/-ες στη δομή των υλικών μερών (bit level, gate level, circuit level) από τα οποία αποτελείται ένας ηλεκτρονικός υπολογιστής, και στον τρόπο επικοινωνίας τους. Ο/Η εκπαιδευόμενος/-η θα διδαχθεί ιστορικές αρχιτεκτονικές Η/Υ και τη σημασία τους, τις πλέον πρόσφατες τεχνολογίες υλικού (Hardware), καθώς και τον τρόπο που αυτές συνδυάζονται σε επίπεδο αποθήκευσης και σε επίπεδο επεξεργασίας δεδομένων. Επιπλέον, θα διδαχθούν τεχνικές σελιδοποίησης/κατάτμησης της Κύριας Μνήμης. Τέλος, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με τα εισαγωγικά στοιχεία μιας Συμβολικής Γλώσσας (Assembly Language) και της μετατροπής της σε Γλώσσα Μηχανής (Machine Language).

Προσδοκώμενα μαθησιακά αποτελέσματα

Ολοκληρώνοντας τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναλύουν τους τρόπους μετατροπής μεταξύ των αριθμητικών συστημάτων δυαδικού, δεκαδικού (μη προσημασμένου μέτρου, προσημασμένου μέτρου, προσημασμένου συμπληρώματος ως προς 1 και 2, BCD, GRAY), οκταδικού και δεκαεξαδικού συστήματος, με ακέραιο ή/και δεκαδικό μέρος, και την κωδικοποίηση mantissa,
- Κατονομάζουν τις περιπτώσεις υπερχείλισης καταχωρητή (overflow),
- Αναγνωρίζουν τους διαφορετικούς τύπους λογικών πυλών,
- Αναγνωρίζουν τη συμβολή του Alan Turing στην Επιστήμη των Υπολογιστών,
- Αναφέρουν τα υλικά μέρη από τα οποία αποτελείται ο υπολογιστής IAS του Von Neumann,
- Διακρίνουν τους τρόπους διευθυνσιοδότησης (Immediate, Direct, Indirect, Register Direct, Register Indirect, Displacement/Indexing),
- Επιλέγουν τα κατάλληλα εξαρτήματα για τη συναρμολόγηση ενός ηλεκτρονικού υπολογιστή,
- Αναγνωρίζουν τη σημασία του συμβατού υλικού (Hardware),
- Αναπτύσσουν απλά προγράμματα σε Assembly MIPS,
- Αναλύουν τις πιθανές αιτίες βλάβης υλικού ενός Ηλεκτρονικού Υπολογιστή,
- Υιοθετούν θετική στάση ως προς τις νέες τεχνολογίες υλικού και της χρήσης τους,
- Συσχετίζουν τη συμβατότητα υλικών μερών ενός Ηλεκτρονικού Υπολογιστή μεταξύ τους.

Βασικές έννοιες και λέξεις-κλειδιά

Συστήματα αρίθμησης / Μετατροπή μεταξύ αριθμητικών συστημάτων / Υλικό υπολογιστή / Δομή ΚΜΕ / Κωδικοποίηση χαρακτήρων / Είδη μνήμης / Λειτουργία καταχωρητών / Λογικές πύλες / Άλγεβρα Boole / Γλώσσα Assembly.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (1), Εργαστήριο (1), Σύνολο (2).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΑΝΑΠΑΡΑΣΤΑΣΗ ΔΕΔΟΜΕΝΩΝ ΣΤΟΝ ΥΠΟΛΟΓΙΣΤΗ
2	ΥΛΙΚΟ ΥΠΟΛΟΓΙΣΤΗ
3	ΚΕΝΤΡΙΚΗ ΜΟΝΑΔΑ ΕΠΕΞΕΡΓΑΣΙΑΣ
4	ΛΟΓΙΚΕΣ ΠΥΛΕΣ ΚΑΙ ΠΡΑΞΕΙΣ
5	ΜΝΗΜΗ ΚΑΙ ΜΕΤΑΦΟΡΑ ΔΕΔΟΜΕΝΩΝ
6	ΕΙΣΑΓΩΓΗ ΣΤΗ ΣΥΜΒΟΛΙΚΗ ΓΛΩΣΣΑ ASSEMBLY

Κείμενο αναφοράς

2.1.A.1 | ΑΝΑΠΑΡΑΣΤΑΣΗ ΔΕΔΟΜΕΝΩΝ ΣΤΟΝ ΥΠΟΛΟΓΙΣΤΗ

Σκοπός της υποενότητας «Αναπαράσταση δεδομένων στον υπολογιστή» είναι να διδαχθούν οι εκπαιδευόμενοι/-ες τα αριθμητικά συστήματα που χρησιμοποιούνται στον υπολογιστή και τις μονάδες μέτρησης που χρησιμοποιούνται για την αποθήκευση της πληροφορίας. Αρχικά, εισάγονται στην έννοια του δυαδικού (binary) συστήματος και μαθαίνουν να μετατρέπουν από το δεκαδικό στο δυαδικό σύστημα και αντίστροφα. Γίνεται ανάλυση του τρόπου αναπαράστασης αριθμών κινητής υποδιαστολής και προσημασμένων αριθμών. Στη συνέχεια, γίνεται παρουσίαση του οκταδικού (octal) και του δεκαεξαδικού (hexadecimal) συστήματος αρίθμησης, δίνοντας έμφαση στο δεκαεξαδικό σύστημα, και περιγράφεται η αναγκαιότητα και τα πλεονεκτήματα κάθε τρόπου αναπαράστασης. Η υποενότητα συνεχίζει με τις αριθμητικές πράξεις μεταξύ δυαδικών αριθμών· πιο συγκεκριμένα, αναλύονται οι πράξεις της πρόσθεσης και αφαίρεσης μη προσημασμένων και προσημασμένων δυαδικών ακεραίων, η τεχνική συμπληρώματος με βάση το 1 και το 2, και εξηγείται ο τρόπος πολλαπλασιασμού και διαίρεσης, εστιάζοντας την προσοχή στους μη προσημασμένους αριθμούς. Επιπρόσθετα, αναφέρονται οι κωδικοποιήσεις BCD, GRAY και mantissa. Στη συνέχεια, οι εκπαιδευόμενοι/-ες, έχοντας εξοικειωθεί με το δυαδικό σύστημα αρίθμησης, μαθαίνουν τις μονάδες μέτρησης για αποθήκευση δεδομένων από το bit στο byte, στο ZB (zettabyte) και στο QB (quettabyte), ώστε να γνωρίζουν τις σύγχρο-

νες μονάδες μέτρησης. Είναι σημαντικό να μάθουν για τα συστήματα κωδικοποίησης χαρακτήρων ANSI, UTF και Unicode, ώστε να κατανοήσουν τους τρόπους αναπαράστασης των δεδομένων. Έτσι, οι εκπαιδευόμενοι/-ες κατανοούν την αναπαράσταση δεδομένων στον υπολογιστή, το δυαδικό σύστημα αρίθμησης και τις μονάδες μέτρησης για την αποθήκευση των δεδομένων.

2.1.A.2 | ΥΛΙΚΟ ΥΠΟΛΟΓΙΣΤΗ

Στόχος της τρέχουσας υποενότητας είναι οι εκπαιδευόμενοι/-ες να μάθουν για το υλικό (Hardware) του υπολογιστή και να μπορούν να αναγνωρίσουν τα εσωτερικά εξαρτήματα και τις περιφερειακές συσκευές. Επίσης, θα μάθουν για την εσωτερική δομή του υπολογιστή, δηλαδή τη μητρική πλακέτα, την Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ), τη μνήμη κ.λπ. Αρχικά, θα εξεταστούν οι βασικές συσκευές του υπολογιστή και οι περιφερειακές. Στο πλαίσιο του μαθήματος, εξετάζονται οι ιδιότητες κάθε συσκευής. Ιδιαίτερη έμφαση πρέπει να δοθεί στις κάρτες γραφικών και δικτύου και στις ιδιότητές τους. Μελετώνται οι τρόποι σύνδεσης των περιφερειακών συσκευών και πώς συμβάλλει η κάθε συσκευή στη συνολική ταχύτητα του υπολογιστή. Κατά τη διάρκεια του μαθήματος, οι εκπαιδευόμενοι/-ες μαθαίνουν να συναρμολογούν έναν υπολογιστή, να αλλάζουν εξαρτήματά του και να ελέγχουν την καλή λειτουργία τους με τη χρήση των απαραίτητων εργαλείων. Αφού ολοκληρωθεί το πρώτο τμήμα, έμφαση πρέπει να δοθεί στο εσωτερικό του υπολογιστή. Αναλύεται ο ρόλος της μητρικής πλακέτας (motherboard) και γίνεται μελέτη της ΚΜΕ, της μνήμης, και των τεχνολογιών δευτερεύουσας μνήμης. Περιγράφονται οι ιδιότητες του κάθε εξαρτήματος, συζητούνται θέματα συμβατότητας και αναλύονται τρόποι αναβάθμισης παλαιότερων υπολογιστών. Η υποενότητα καταλήγει στην παρουσίαση των βασικών τμημάτων της ΚΜΕ και τους τύπους μνήμης που θα αναλυθούν σε επόμενες υποενότητες. Έχοντας ολοκληρώνει την υποενότητα, οι εκπαιδευόμενοι/-ες θα είναι σε θέση να προτείνουν και να συνθέσουν νέους υπολογιστές, να εξηγούν τις ιδιότητες και να ελέγχουν τη λειτουργία του υλικού ενός σύγχρονου υπολογιστή.

2.1.A.3 | ΚΕΝΤΡΙΚΗ ΜΟΝΑΔΑ ΕΠΕΞΕΡΓΑΣΙΑΣ

Η υποενότητα εστιάζεται στις υπομονάδες της ΚΜΕ και τις διαφορετικές αρχιτεκτονικές τους. Αρχικά, γίνεται αναφορά σε ένα γενικό μοντέλο και παρουσιάζονται οι υπομονάδες τους, δηλαδή η Αριθμητική και λογική μονάδα (ALU-Arithmetic Logic Unit), η μονάδα ελέγχου (CU-Control Unit) και η μονάδα μνήμης που διαθέτει. Ο ρόλος και ο τρόπος συνεργασίας και επικοινωνίας της ΚΜΕ με τη μνήμη αναλύονται, ώστε να γίνει κατανοητός ο τρόπος εκτέλεσης εντολών. Ο ρόλος των καταχωρητών της ΚΜΕ, το μέγεθός τους και οι τύποι (π.χ. μετρητής προγράμματος, καταχωρητής γενικής χρήσης [R0, R1, R2...] κ.λπ.) περιγράφονται, για να γίνει κατανοητός ο τρόπος εκτέλεσης των εντολών. Προτείνεται να γίνει αναφορά στον επεξεργαστή γραφικών (GPU-Graphics Processing Unit) και στον ρόλο του στην επιτάχυνση των εργασιών.

Ο τρόπος μέτρησης της ταχύτητας της ΚΜΕ επεξεργασίας εξηγείται στην ενότητα. Στην υποενότητα, αναλύεται ο ρόλος του Alan Turing, της θεωρητικής μηχανής του και του πρώτου υπολογιστή που κατασκεύασε. Το μηχάνημα IAS, ο πρώτος ψηφιακός υπολογιστής που δημιουργήθηκε από το Institute for Advanced Study (IAS), ο ρόλος του von Neumann που τον κατασκεύασε και η αρχιτεκτονική von Neumann είναι σημαντικό να συζητηθούν, ώστε να γνωρίζουν οι εκπαιδευόμενοι/-ες κάποιους από τους θεμελιωτές του τομέα. Προτείνεται να γίνει σύντομη αναφορά στους κβαντικούς υπολογιστές. Έτσι, οι εκπαιδευόμενοι/-ες αποκτούν καλή κατανόηση της δομής και λειτουργίας της ΚΜΕ και των αρχιτεκτονικών που αποτέλεσαν ορόσημο στην επιστήμη της Πληροφορικής.

2.1.A.4 | ΛΟΓΙΚΕΣ ΠΥΛΕΣ ΚΑΙ ΠΡΑΞΕΙΣ

Η υποενότητα «Λογικές πύλες και πράξεις» αποτελεί σημαντικό τμήμα των βασικών μαθημάτων στην Πληροφορική και στην ηλεκτρονική. Αυτό το μάθημα επικεντρώνεται στη θεωρία και στην εφαρμογή των λογικών πυλών και πράξεων κατά την επεξεργασία της πληροφορίας. Οι εκπαιδευόμενοι/-ες κατανοούν καλύτερα τον ρόλο της ALU και των καταχωρητών του επεξεργαστή. Στην υποενότητα αυτή, οι εκπαιδευόμενοι/-ες εξετάζουν τις βασικές λογικές πύλες (NOT, AND, OR, XOR, NAND και NOR) και μαθαίνουν πώς λειτουργούν και ποιος είναι τελικά ο ρόλος τους στα ηλεκτρονικά συστήματα του υπολογιστή. Μαθαίνουν τη σχηματική αναπαράσταση των λογικών πυλών, ώστε να μπορούν να κατανοούν σύνθετα σχήματα με λογικές πύλες, και μαθαίνουν να εκτελούν λογικές πράξεις με την άλγεβρα Boole – σε αυτό το σημείο εξετάζεται η έννοια της προτεραιότητας των πράξεων και ο τρόπος με τον οποίο οι λογικές πράξεις επηρεάζουν τον μετασχηματισμό των δεδομένων εισόδου. Μαθαίνουν ακόμη να μετασχηματίζουν ένα κύκλωμα που έχει υλοποιηθεί (για παράδειγμα, με πύλες AND, OR, NOT σε ισοδύναμο κύκλωμα με πύλες NAND, NOR). Τέλος, προτείνεται να παρουσιαστούν τα κυκλώματα των βασικών κυκλωμάτων αριθμητικών πράξεων της άλγεβρας (κύκλωμα ημιαθροιστή, κύκλωμα αθροιστή σειριακού και παράλληλου, κύκλωμα αφαιρέτη, κύκλωμα δυαδικού πολλαπλασιασμού). Η υποενότητα βοηθά τους/τις καταρτιζόμενους/-ες να εξασκηθούν στο σχεδιασμό και την ανάλυση των λογικών συναρτήσεων, που αποτελούν τη βάση για τα ηλεκτρονικά κυκλώματα.

2.1.A.5 | ΜΝΗΜΗ ΚΑΙ ΜΕΤΑΦΟΡΑ ΔΕΔΟΜΕΝΩΝ

Σε αυτή την υποενότητα, οι εκπαιδευόμενοι/-ες θα κατανοήσουν τα είδη μνήμης και τον τρόπο διευθυνσιοδότησης και μεταφοράς δεδομένων από και προς τον επεξεργαστή από τις άλλες εσωτερικές μονάδες που βρίσκονται ενσωματωμένες στη μητρική ή συνδέονται πάνω σε αυτή. Αρχικά, παρουσιάζονται αναλυτικά οι τεχνολογίες μνήμης, ώστε να γίνουν κατανοητές οι διαφορές των τεχνολογιών (π.χ. DDR4 και DDR5 και γενικά ο τύπος, η συχνότητα, και τα υπόλοιπα χαρακτηριστικά των μνημών). Ο ρόλος της μνήμης στην ταχύτητα του υπολογιστή και ο τρόπος χρήσης της από τον

επεξεργαστή είναι σημαντικό να εξηγηθούν αναλυτικά στο πλαίσιο του μαθήματος. Σε τεχνικό επίπεδο, προτείνεται να εξηγηθούν τα κυκλώματα, η δομή και οι τεχνολογίες που χρησιμοποιούνται για τη δημιουργία των μνημών (π.χ. ημιαγωγικών, μαγνητικών και οπτικών). Παρουσιάζεται η έννοια και η σημαντικότητά της κρυφής μνήμης, εξηγούνται σύντομα οι τρόποι διευθυνσιοδότησης, και αναλύονται τα χαρακτηριστικά των τεχνολογιών μόνιμης αποθήκευσης (π.χ. SSD και M2) που έχουν σύντομα περιγραφεί σε προηγούμενη υποενότητα της ενότητας αυτής. Συζητούνται επίσης οι διάφοροι τύποι αρτηριών μεταφοράς δεδομένων μεταξύ δευτερεύουσας μνήμης και επεξεργαστή, και εξηγείται ο τρόπος που επηρεάζουν τη συνολική ταχύτητα του υπολογιστή. Μέσω της υποενότητας αυτής, οι εκπαιδευόμενοι/-ες κατανοούν τις τεχνολογίες προσωρινής και μόνιμης αποθήκευσης, τη δομή τους, τα τεχνικά χαρακτηριστικά τους και είναι σε θέση να κατανοούν τις εξελίξεις στον συγκεκριμένο τομέα.

2.1.A.6 | ΕΙΣΑΓΩΓΗ ΣΤΗ ΣΥΜΒΟΛΙΚΗ ΓΛΩΣΣΑ ASSEMBLY

Στην υποενότητα αυτή, γίνεται εισαγωγή στη Συμβολική Γλώσσα Assembly, ώστε να γίνει σαφής η λειτουργία των υπολογιστών μέσα από την εκτέλεση εντολών σε χαμηλό επίπεδο που απαιτεί γνώση της αρχιτεκτονικής του υπολογιστή. Η γλώσσα Assembly είναι πολύ κοντά στη γλώσσα μηχανής και στο υλικό του υπολογιστή. Κάθε συγκεκριμένη αρχιτεκτονική συνόλου εντολών, δηλαδή κάθε οικογένεια επεξεργαστών, έχει τη δική της συμβολική γλώσσα, η οποία δίνεται συνήθως από τον κατασκευαστή της. Συνεπώς, στην υποενότητα αυτή χρησιμοποιείται η Assembly MIPS, που βασίζεται στην αρχιτεκτονική MIPS (Microprocessor without Interlocked Pipeline Stages) και χρησιμοποιεί έναν σχετικά μικρό αριθμό απλών εντολών επεξεργαστή, προσπαθώντας να επιτύχει υψηλή απόδοση (RISC-Reduced Instruction Set Computer). Εξηγούνται οι εντολές της συμβολικής γλώσσας που περιλαμβάνουν βασικές λειτουργίες, όπως φόρτωση (load), αποθήκευση (store), αριθμητικές πράξεις (π.χ. πρόσθεση, αφαίρεση, πολλαπλασιασμός) και ελέγχους ροής μέσα από παραδείγματα, και παρουσιάζεται η δομή ενός προγράμματος σε Assembly. Η γλώσσα διδάσκεται μέσω προσομοιωτή και, εφόσον η υποενότητα έχει εισαγωγικό χαρακτήρα, ο βασικός σκοπός είναι οι εκπαιδευόμενοι/-ες να μπορούν να εκτελέσουν βασικά προγράμματα και να χρησιμοποιήσουν τους καταχωρητές. Παρουσιάζονται οι διαφορετικοί καταχωρητές και δίνονται παραδείγματα με πράξεις, συγκρίσεις αριθμών, εκτυπώσεις σε δεκαδικό και δεκαεξαδικό σύστημα και πρόσβαση στη μνήμη του υπολογιστή. Συμπερασματικά, στη συγκεκριμένη υποενότητα, οι εκπαιδευόμενοι/-ες μαθαίνουν την ανάπτυξη βασικών προγραμμάτων στη γλώσσα Assembly και κατανοούν περαιτέρω τη δομή και λειτουργία της ΚΜΕ.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Hennessy, L. J. & Patterson, D. A. (2020). *Αρχιτεκτονική υπολογιστών: μια ποσοτική προσέγγιση*. Αθήνα: Εκδόσεις Κλειδάριθμος.
2. Patterson, A. D. & Hennessy, L. J. (2023). *Οργάνωση και σχεδίαση υπολογιστών MIPS: η διασύνδεση υλικού και λογισμικού*. Αθήνα: Εκδόσεις Κλειδάριθμος.

Συμπληρωματικές

1. Δρόσος, Δ., Βουγιούκας, Δ., Καλλίγερος, Ε., Κοκολάκης, Σ. & Σκιάνης, Χ. (2015). *Εισαγωγή στην επιστήμη των υπολογιστών & επικοινωνιών*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/4582>
2. Norton, P. & Socha, J. (2009). *Το βιβλίο της Assembly για τα PC*. Αθήνα: Εκδόσεις Κλειδάριθμος.

2.1.B • ΣΥΓΧΡΟΝΑ ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Σύγχρονα λειτουργικά συστήματα» εισάγει τους/τις εκπαιδευόμενους/-ες στην έννοια του λειτουργικού συστήματος και στη χρήση του σε όλο το φάσμα των επιστημών ανάπτυξης λογισμικού και δικτύων επικοινωνίας. Ο/Η εκπαιδευόμενος/-η θα έρθει σε επαφή με τις πιο γνωστές και πλέον πρόσφατες τεχνολογίες λειτουργικών συστημάτων, τον τρόπο που αυτές χρησιμοποιούνται για τη διαχείριση διεργασιών και αρχείων, την εγκατάσταση/απεγκατάσταση εφαρμογών και τη συμπίεση/αποσυμπίεση δεδομένων. Επιπροσθέτως, αναλύονται οι αλγόριθμοι χρονοπρογραμματισμού διεργασιών και τα διαφορετικά κριτήρια, βάσει των οποίων διαχωρίζονται. Τέλος, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με τη χρήση εικονικών μηχανών, docker container και kubernetes, και τον πολλαπλά εφαρμοζόμενο ρόλο που αυτές κατέχουν στη βιομηχανία της Πληροφορικής.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναγνωρίζουν τα πλεονεκτήματα των λειτουργικών συστημάτων (των ΛΣ που απαιτούν άδεια χρήσης και αυτών που βασίζονται στο ΕΛ/ΛΑΚ),
- Επιλέγουν το κατάλληλο λειτουργικό σύστημα που θα πρέπει να εγκαταστήσουν, αναλόγως της χρήσης του,
- Αναγνωρίζουν τη διαφορά μεταξύ των οικογενειών λειτουργικών συστημάτων και

την αντιστοίχιση της κάθε μιας, ανάλογα με τις ανάγκες της επιχείρησης/οργανισμού,

- Αναπτύσσουν τρόπους βέλτιστης παραμετροποίησης του λειτουργικού συστήματός τους, προκειμένου να εξυπηρετούν τον σκοπό χρήσης του,
- Αναλύουν τις πιθανές αιτίες βλάβης του λειτουργικού συστήματος ενός ηλεκτρονικού υπολογιστή,
- Αποφασίζουν για τη χρήση ή μη κατάλληλων εργαλείων διαχείρισης υλικού/εφαρμογών/διεργασιών, εμφάνισης κ.λπ., που προσφέρει ένα λειτουργικό σύστημα,
- Υιοθετούν θετική στάση ως προς τις νέες τεχνολογίες λειτουργικών συστημάτων, τη σημασία τους στις Τεχνολογίες Πληροφορικής και Επικοινωνιών και της χρήσης τους,
- Αποφασίζουν για το είδος των εντολών περιβάλλοντος κονσόλας που θα πρέπει να χρησιμοποιήσουν,
- Διακρίνουν τις διαφορές μεταξύ των οικογενειών λειτουργικών συστημάτων που βασίζονται σε Windows, Linux και MAC-OS (συγκρίνοντας επιτραπέζιες εκδόσεις και εκδόσεις για φορητές συσκευές [tablet/smartphone κ.λπ.]),
- Διακρίνουν τις διαφορές μεταξύ εικονικών μηχανών, docker container και kubernetes,
- Εγκαθιστούν και παραμετροποιούν εικονικές μηχανές με τη χρήση κατάλληλου λογισμικού.

Βασικές έννοιες και λέξεις-κλειδιά

Αλγόριθμοι χρονοδρομολόγησης / Γραμμή εντολών / Εικονικοποίηση / Android / Mobile development / Linux / Windows.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (3), Σύνολο (3).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΔΟΜΗ ΚΑΙ ΒΑΣΙΚΕΣ ΛΕΙΤΟΥΡΓΙΕΣ ΛΕΙΤΟΥΡΓΙΚΟΥ ΣΥΣΤΗΜΑΤΟΣ
2	ΑΛΓΟΡΙΘΜΟΙ ΧΡΟΝΟΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΔΙΕΡΓΑΣΙΩΝ
3	ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ MICROSOFT
4	ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ LINUX (UNIX SHELL & GUI)
5	ΤΕΧΝΟΛΟΓΙΕΣ ΕΙΚΟΝΙΚΟΠΟΙΗΣΗΣ
6	ΑΛΛΕΣ ΟΙΚΟΓΕΝΕΙΕΣ ΛΕΙΤΟΥΡΓΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Κείμενο αναφοράς

2.1.B.1 | ΔΟΜΗ ΚΑΙ ΒΑΣΙΚΕΣ ΛΕΙΤΟΥΡΓΙΕΣ ΛΕΙΤΟΥΡΓΙΚΟΥ ΣΥΣΤΗΜΑΤΟΣ

Η εκπαιδευτική υποενότητα «Δομή και βασικές λειτουργίες λειτουργικού συστήματος» μελετά το τι είναι το λειτουργικό σύστημα, εξηγώντας τις αρμοδιότητες και τους βασικούς στόχους που εξυπηρετεί. Κρίνεται σκόπιμη η παρουσίαση της εξέλιξης των λειτουργικών συστημάτων, αναλύοντας τα χαρακτηριστικά κάθε γενιάς. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα εστιάσουν την προσοχή τους στη διάκριση και την ταξινόμηση των λειτουργικών συστημάτων σε συστήματα ομαδικής επεξεργασίας (multiprogramming), καταμερισμού χρόνου (time sharing), πραγματικού χρόνου (real time), κατανεμημένων (distributed) και ενσωματωμένων (embedded), αναλύοντας τις διαφορές μεταξύ τους. Στο πλαίσιο της υποενότητας, αναλύεται ο όρος της διεργασίας και η διαφορά μεταξύ διεργασίας και προγράμματος. Οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με τον ρόλο της διεργασίας στην κατηγοριοποίηση των λειτουργικών συστημάτων σε πολυ-διεργασιακά (multi-tasking) και μονο-διεργασιακά (single tasking). Επιπρόσθετα, αναλύονται οι έννοιες του πολυπρογραμματισμού και της εναλλαγής, καθώς και η διαχείριση των διεργασιών από ένα λειτουργικό σύστημα. Τέλος, τονίζονται οι διακριτές καταστάσεις στις οποίες μπορεί να βρεθεί κάθε διεργασία κατά τη διάρκεια ζωής της, και εξετάζονται οι λειτουργίες επί διεργασιών, όπως η αλλαγή προτεραιότητας, το μπλοκάρισμα/ξεμπλοκάρισμα και η νάρκωση/αφύπνιση μιας διεργασίας. Ο στόχος της εκπαιδευτικής υποενότητας είναι να εισαγάγει τους/τις εκπαιδευόμενους/-ες στις βασικές λειτουργίες που επιτελεί ένα λειτουργικό σύστημα και τον ρόλο που διαδραματίζει στην εκτέλεση πολλών προγραμμάτων παράλληλα.

2.1.B.2 | ΑΛΓΟΡΙΘΜΟΙ ΧΡΟΝΟΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΔΙΕΡΓΑΣΙΩΝ

Η εκπαιδευτική υποενότητα «Αλγόριθμοι χρονοπρογραμματισμού διεργασιών» ασχολείται με τον τρόπο και τη σειρά που τα σύγχρονα λειτουργικά συστήματα εκτελούν πολλές διεργασίες. Οι εκπαιδευόμενοι/-ες θα εστιάσουν την προσοχή στην κατανόηση της σημασίας του χρονοδρομολογητή (scheduler) και των αλγορίθμων που χρησιμοποιούνται για την εκτέλεση των εργασιών. Αρχικά, εξετάζονται οι έννοιες της μακροχρόνιας και της βραχυχρόνιας χρονοδρομολόγησης και επισημαίνονται οι μεταξύ τους διαφορές. Στη συνέχεια, αναλύεται το ποιος είναι ο βασικός στόχος ενός αλγόριθμου χρονοδρομολόγησης και τονίζονται τα χαρακτηριστικά/κριτήρια που θα πρέπει να συγκεντρώνει, προκειμένου να θεωρείται ανταγωνιστικός. Συγκεκριμένα, κρίνεται σκόπιμη η ανάλυση των χαρακτηριστικών του βαθμού χρησιμοποίησης της κεντρικής μονάδας επεξεργασίας (CPU Utilization), της ρυθμαπόδοσης (throughput), του μέσου χρόνου αναμονής και του μέσου χρόνου απόκρισης. Επίσης, οι εκπαιδευόμενοι/-ες εξοικειώνονται με τις έννοιες της προεκχωρητικής και μη προεκχωρητικής χρονοδρομολόγησης και, μέσα από πρακτικά παραδείγματα, έρχονται σε επαφή με τους βασικούς αλγόριθμους χρονοδρομολόγησης. Στο πλαίσιο της υποενότητας,

αναλύονται οι μη προεκχωρητικοί αλγόριθμοι, με βάση τη σειρά άφιξης (First Come First Serve) και με βάση τη μικρότερη διάρκεια (Shortest Job First). Ακόμη, αναλύονται οι προεκχωρητικοί αλγόριθμοι, με βάση τη μικρότερη υπολειπόμενη διάρκεια (Shortest Remaining Time First) και την εξυπηρέτηση εκ περιτροπής (Round Robin). Επιπρόσθετα, αναλύεται ο αλγόριθμος εξυπηρέτησης προτεραιοτήτων (priority scheduling) που συναντάται και σε προεκχωρητική και σε μη προεκχωρητική μορφή. Η υποεπάρκεια αυτή παρέχει την απαραίτητη βάση για την κατανόηση και εφαρμογή των αλγορίθμων χρονοδρομολόγησης για την πιο αποτελεσματική εκτέλεση των διεργασιών που επιτελούνται σε ένα σύγχρονο λειτουργικό σύστημα.

2.1.B.3 | ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ MICROSOFT

Η εν λόγω εκπαιδευτική υποεπάρκεια εξετάζει την οικογένεια λειτουργικών συστημάτων της εταιρείας Microsoft. Αρχικά, εξετάζεται το λειτουργικό σύστημα MS-DOS, επισημαίνοντας ότι τα πρώτα χρόνια των υπολογιστών δεν υπήρχαν παράθυρα ή γραφικά στους υπολογιστές. Οι εκπαιδευόμενοι/-ες εξοικειώνονται με το σύστημα εντολών του MS-DOS και εστιάζουν την προσοχή στην κατανόηση των εντολών (και στις παραμέτρους που δέχονται), οι οποίες σχετίζονται με τη διαχείριση αρχείων και φακέλων. Επίσης, κατανοούν ότι η γραμμή εντολών δίνει τη δυνατότητα εκτέλεσης προγραμμάτων που δεν είναι διαθέσιμα στο γραφικό περιβάλλον (έλεγχος αρχείων συστήματος, εμφάνιση πληροφοριών σύνδεσης και εμφάνιση λίστας με τους εγκατεστημένους drivers). Στη συνέχεια, αναλύονται τα χαρακτηριστικά των διαφορετικών εκδόσεων του γραφικού περιβάλλοντος Windows, από την έκδοση Windows 1.x μέχρι και την έκδοση Windows 11 (η οποία είναι η τελευταία έκδοση, κατά τη διάρκεια συγγραφής του παρόντος εγχειριδίου). Μέσα από πρακτικά παραδείγματα, οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με τη χρήση των παραθύρων, τις βασικές λειτουργίες και ρυθμίσεις του λειτουργικού συστήματος, τη δομή και τη διαχείριση φακέλων και αρχείων. Ακόμη, οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με δημοφιλή εργαλεία των Windows, όπως είναι η αναζήτηση αρχείων και φακέλων, η συμπίεση/αποσυμπίεση αρχείων και η διαχείριση συνδεδεμένων εκτυπωτών και εργασιών εκτύπωσης. Επιπρόσθετα, θα κατανοήσουν τη σημασία των αντικών εφαρμογών και θα διδαχθούν τη διαδικασία εγκατάστασης/απεγκατάστασης εφαρμογών λογισμικού στον Η/Υ. Τέλος, κρίνεται σκόπιμη η αναφορά στη σημασία της τεχνητής νοημοσύνης και η εξάσκηση σε εργαλεία όπως το ChatGPT. Επιπλέον, στην υποεπάρκεια αυτή, αναλύονται οι πιθανές αιτίες βλάβης του λειτουργικού συστήματος ενός ηλεκτρονικού υπολογιστή. Οι εκπαιδευόμενοι/-ες θα διδαχθούν να διαχειρίζονται εφαρμογές που δεν ανταποκρίνονται, και θα εξοικειωθούν με τους τρόπους βέλτιστης παραμετροποίησης του λειτουργικού συστήματος. Συνοψίζοντας, στόχος της υποεπάρκειας είναι να προσφέρει τις γνώσεις και τις δεξιότητες που απαιτούνται, προκειμένου οι εκπαιδευόμενοι/-ες να είναι σε θέση να κατανοήσουν και να εφαρμόσουν τις δυνατότητες των λειτουργικών συστημάτων της οικογένειας Microsoft.

2.1.B.4 | ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ LINUX (UNIX SHELL & GUI)

Η συγκεκριμένη εκπαιδευτική υποενότητα ασχολείται με το λειτουργικό σύστημα Linux, το οποίο ανήκει στην οικογένεια των λειτουργικών συστημάτων ανοικτού κώδικα Unix. Αρχικά, εξετάζονται οι διαφορετικές διανομές του συστήματος Linux και οι εκπαιδευόμενοι/-ες έρχονται σε επαφή με το γραφικό περιβάλλον (GUI) των πιο διαδεδομένων από αυτές, εστιάζοντας την προσοχή στις διαφορές που έχουν με το αντίστοιχο γραφικό περιβάλλον των Windows, που αναλύθηκε στην προηγούμενη υποενότητα. Στο πλαίσιο της υποενότητας, οι εκπαιδευόμενοι/-ες θα διδαχθούν το πώς μέσα από τη γραμμή εντολών του Linux (η οποία ονομάζεται και κονσόλα ή τερματικό) μπορούν να εκτελεστούν αρκετά πιο προηγμένες εργασίες σε σχέση με το αντίστοιχο γραφικό περιβάλλον. Επομένως, θα λάβουν γνώσεις και θα αναπτύξουν δεξιότητες για την αποτελεσματική χρήση της γραμμής εντολών σε πληθώρα εργασιών, με σκοπό τη βέλτιστη παραμετροποίηση και διαχείριση του λειτουργικού συστήματος. Μέσω της υποενότητας, οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με τη δομή του συστήματος αρχείων του Linux και θα γνωρίσουν τις βασικές εντολές για λειτουργίες σε αρχεία και καταλόγους (directories), όπως είναι η διαγραφή και η δημιουργία, η αντιγραφή, η μετακίνηση, η αλλαγή ονόματος. Επίσης, θα κατανοήσουν τη χρήση των εξουσιοδοτήσεων και τη σημασία τους στην προστασία των αρχείων του/της χρήστη/-τριας από μη εξουσιοδοτημένη πρόσβαση από άλλους χρήστες/-τριες που χρησιμοποιούν το σύστημα. Επιπλέον, επισημαίνεται ότι υπάρχουν κοινότητες που επικεντρώνονται σε συγκεκριμένες διανομές του Linux, στις οποίες οι χρήστες/-τριες μπορούν να καταφύγουν για διάφορα θέματα υποστήριξης. Συνοψίζοντας, στόχος της υποενότητας είναι να προσφέρει τις γνώσεις και τις δεξιότητες που απαιτούνται, προκειμένου οι εκπαιδευόμενοι/-ες να είναι σε θέση να κατανοήσουν και να εφαρμόσουν τις δυνατότητες του λειτουργικού συστημάτων Linux.

2.1.B.5 | ΤΕΧΝΟΛΟΓΙΕΣ ΕΙΚΟΝΙΚΟΠΟΙΗΣΗΣ

Η εν λόγω εκπαιδευτική υποενότητα αφιερώνεται στη σημασία της εικονικοποίησης (virtualization) και τα οφέλη που προσδίδει στις σύγχρονες επιχειρήσεις, επιτρέποντάς τους να επεκτείνουν τις δυνατότητές τους και να βελτιώσουν την παραγωγικότητά τους. Αρχικά, αναλύεται η τεχνολογία που επιτρέπει την ταυτόχρονη χρησιμοποίηση πολλών λειτουργικών συστημάτων σε έναν κεντρικό υπολογιστή, γνωστή ως Hypervisor ή Virtual Machine Monitor. Οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με τη διαχείριση ορισμένων από τα πιο συχνά σε χρήση Hypervisors, όπως είναι το VMware και το VirtualBox, και θα δημιουργήσουν στιγμιότυπα από πολλαπλές εικονικές μηχανές, με τα πιο γνωστά λειτουργικά συστήματα. Στο πλαίσιο της υποενότητας, αναλύεται η έννοια και οι τεχνολογίες του container και εξετάζονται οι διαφορές που παρουσιάζει με μια εικονική μηχανή. Επίσης, οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με τον μηχανισμό ανοικτού κώδικα Docker. Οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με τις δυνατότητες που παρέχει η εν λόγω πλατφόρμα δημιουργίας container, προ-

κειμένου να εκτελούν και να διαχειρίζονται εφαρμογές σε διαφορετικά περιβάλλοντα με εύκολο και αποτελεσματικό τρόπο. Επιπλέον, κρίνεται σκόπιμο να επισημανθεί ότι η εξέλιξη της τεχνολογίας container οδήγησε στην ανάπτυξη εργαλείων (π.χ. kubernetes) για τη δημιουργία και τη διαχείριση εφαρμογών πολλαπλών container σε πολλούς servers. Συνοψίζοντας, σκοπός της εκπαιδευτικής υποενότητας είναι να παράσχει την απαραίτητη βάση για την απόκτηση γνώσεων και δεξιοτήτων, που σχετίζονται με τις τεχνολογίες εικονικοποίησης. Οι εκπαιδευόμενοι/-ες θα είναι σε θέση να χρησιμοποιήσουν τα κατάλληλα εργαλεία για την εγκατάσταση και παραμετροποίηση εικονικών μηχανών, και να διακρίνουν τις διαφορές μεταξύ εικονικών μηχανών, docker container και kubernetes, αλλά και τον ρόλο που διαδραματίζουν στη βιομηχανία της Πληροφορικής.

2.1.B.6 | ΑΛΛΕΣ ΟΙΚΟΓΕΝΕΙΕΣ ΛΕΙΤΟΥΡΓΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Η μαθησιακή υποενότητα «Άλλες οικογένειες λειτουργικών συστημάτων» αφιερώνεται στα λειτουργικά συστήματα που δεν αναλύθηκαν στις προηγούμενες υποενότητες. Πιο συγκεκριμένα, αναλύεται το λειτουργικό σύστημα MAC-OS, όπου συγκρίνονται οι επιτραπέζιες εκδόσεις του με τις αντίστοιχες οικογένειες λειτουργικών συστημάτων Microsoft και Unix (εξετάστηκαν στις προηγούμενες υποενότητες), με κριτήριο την παρουσία μητρώου (registry), την οργάνωση του συστήματος αρχείων, τις τεχνολογίες Remote Access, το πλήθος χρηστών/-τριών και την ένταξή τους ή μη στις κατηγορίες του ΕΛ/ΛΑΚ (Ελεύθερου Λογισμικού/Λογισμικού Ανοικτού Κώδικα). Επίσης, οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με τις εκδόσεις των λειτουργικών συστημάτων που χρησιμοποιούνται στις φορητές συσκευές (smartphone, tablet κ.ά.). Έτσι, στο πλαίσιο της υποενότητας, εξετάζονται τα λειτουργικά συστήματα Android, iOS, Windows Mobile (Windows Phone), Blackberry OS και Symbian, πραγματοποιώντας μια ανάλυση SWOT (Strengths, Weaknesses, Opportunities, Threats) για το καθένα από αυτά. Στη συγκεκριμένη υποενότητα, οι εκπαιδευόμενοι/-ες λαμβάνουν γνώσεις και κατανοούν τη σημασία που έχουν οι εφαρμογές κινητών συσκευών στην καθημερινότητα των ανθρώπων. Επιπλέον, θα πρέπει να γίνει κατανοητή η ανάγκη του mobile development στο κομμάτι των επιχειρήσεων, αφού μέσω αυτού μπορούν να προκύψουν ευκαιρίες ανάπτυξης και τρόποι με τους οποίους μια επιχείρηση μπορεί να αυξήσει την παραγωγικότητά της. Ως εκ τούτου, οι εκπαιδευόμενοι/-ες θα αντιληφθούν τη σημασία του λειτουργικού συστήματος με μοναδικά χαρακτηριστικά, που να μπορούν να αξιοποιήσουν οι εφαρμογές για κινητές συσκευές. Συνοψίζοντας, η εκπαιδευτική υποενότητα επιδιώκει να παράσχει τις απαραίτητες γνώσεις σχετικά με τις διαφορετικές εκδόσεις των λειτουργικών συστημάτων, επισημαίνοντας τον ρόλο τους στις σύγχρονες τεχνολογικές προκλήσεις και ιδιαίτερα στον χώρο των κινητών συσκευών.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Silberschatz, A., Galvin, P. B. & Gagne, G. (2009). *Λειτουργικά Συστήματα*. Αθήνα: Εκδόσεις Ίων.
2. Tanenbaum, A. S. & Herbert, B. (2018). *Σύγχρονα Λειτουργικά Συστήματα*. Αθήνα: Εκδόσεις Κλειδάριθμος.

Συμπληρωματικές

1. Negus, C. (2020). *Linux Bible*. New Jersey: Wiley & Sons.
2. Stallings, W. (2017). *Λειτουργικά Συστήματα*. Θεσσαλονίκη: Εκδόσεις Τζιόλα.

2.1.Γ • ΕΡΓΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Εργαλεία ανάπτυξης εφαρμογών Διαδικτύου» εισάγει τους/τις εκπαιδευόμενους/-ες στη γλώσσα σήμανσης υπερκειμένου (HTML) και στη γλώσσα φύλλων ύφους (CSS), ώστε να αποκτήσουν τις βάσεις για τη δημιουργία περιεχομένου, κατάλληλου για χρήση στον Παγκόσμιο Ιστό. Ειδικότερα, παρουσιάζονται οι πιο πρόσφατες εκδόσεις των προαναφερόμενων γλωσσών (HTML5, CSS3), το συντακτικό, η δομή και τα χαρακτηριστικά τους. Παρουσιάζονται οι ετικέτες της HTML5 για τη δημιουργία επικεφαλίδων, παραγράφων, συνδέσμων, εικόνων, λιστών, πινάκων, φορμών, καθώς και τα στυλ της CSS3 για τη μορφοποίηση των ετικετών και τη δημιουργία πλαισίων, περιγραμμάτων, περιθωρίων, χρωμάτων, διαστάσεων, θέσεων κ.λπ. Επίσης, αναφέρονται λογισμικά σύνταξης περιεχομένου ιστού, εργαλεία απόθεσης κώδικα και ελέγχου εκδόσεων, διαδικτυακά πρωτόκολλα, εργαλεία δοκιμών ασφάλειας ιστοτόπων, δυνατότητες φιλοξενίας ιστοτόπων, καθώς και προχωρημένα θέματα ανάπτυξης εφαρμογών Διαδικτύου, όπως γραφικά, APIs, media, transforms, transitions, flexbox και δημοφιλή Frameworks. Επιπροσθέτως, αναφέρεται η σημασία του τεκμηριωμένου κώδικα και της προστασίας των δεδομένων εκ σχεδιασμού και εξορισμού. Τέλος, σχεδιάζονται και αναπτύσσονται ανταποκρίσιμοι (responsive) και ασφαλείς (secure) ιστότοποι, που υποστηρίζουν οι σύγχρονοι φυλλομετρητές.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Επιλέγουν τα κατάλληλα στοιχεία και χαρακτηριστικά για τη δημιουργία περιεχομένου που θα χρησιμοποιηθεί στον Παγκόσμιο Ιστό,

- Αναγνωρίζουν τη σημασία του δομημένου και τεκμηριωμένου κώδικα,
- Αναπτύσσουν ανταποκρίσιμους (responsive) ιστοτόπους με χρήση HTML5 και CSS3,
- Συνεισφέρουν με πρωτότυπο κώδικα στην κοινότητα των προγραμματιστών ιστοτόπων,
- Τροποποιούν υπάρχοντα κώδικα, ανάλογα με τις προδιαγραφές των ιστοτόπων που αναπτύσσουν,
- Αποφασίζουν για τη χρήση ή μη κατάλληλων λογισμικών, σχετικών με την ανάπτυξη ιστοτόπων,
- Συνεργάζονται με άλλους/-ες εκπαιδευόμενους/-ες, συγκροτώντας ομάδες, με σκοπό τη δημιουργία και την επίλυση προβλημάτων των ιστοτόπων,
- Υιοθετούν θετική στάση ως προς τις νέες τεχνολογίες ανάπτυξης ιστοτόπων.

Βασικές έννοιες και λέξεις-κλειδιά

Ανάπτυξη εφαρμογών Διαδικτύου / Ιστοσελίδα/Ιστότοπος / Δυναμικές ιστοσελίδες / HTML5/CSS3 / Πρωτόκολλα επικοινωνίας / Application Programming Interface (Διεπαφή Προγραμματισμού Εφαρμογών) / Framework (Πλαίσιο εργασίας) / Bootstrap Framework / Foundation Framework / Github / Web Hosting (Φιλοξενία Ιστοσελίδων).

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (5), Σύνολο (5).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ ΣΤΟ ΔΙΑΔΙΚΤΥΟ
2	ΓΛΩΣΣΑ ΑΝΑΠΤΥΞΗΣ ΙΣΤΟΣΕΛΙΔΩΝ HTML
3	ΕΙΣΑΓΩΓΗ ΣΤΑ CSS
4	ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ ΑΝΑΠΤΥΞΗΣ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ
5	ΧΡΗΣΙΜΑ ΕΡΓΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ
6	ΤΕΧΝΙΚΑ ΘΕΜΑΤΑ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ

Κείμενο αναφοράς

2.1.Γ.1 | ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ ΣΤΟ ΔΙΑΔΙΚΤΥΟ

Στη συγκεκριμένη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες εισάγονται στον τρόπο ανάπτυξης εφαρμογών Διαδικτύου και αποκτούν μια ολοκληρωμένη εικόνα για την ανάπτυξη εφαρμογών για το Διαδίκτυο, τις αρχιτεκτονικές και τις τεχνολογίες που χρησιμοποιούνται. Εξηγείται, αρχικά, ο τρόπος λειτουργίας του Διαδικτύου, γίνεται αποσαφήνιση όρων, όπως είναι το Διαδίκτυο (Internet) και ο Παγκόσμιος Ιστός (World Wide Web), η ιστοσελίδα, ο ιστότοπος, η χρησιμότητα των υπερσυνδέσμων,

και αναλύεται η έννοια του εξυπηρετητή. Παρουσιάζονται, επίσης, τα πρωτόκολλα που χρησιμοποιούνται στο Διαδίκτυο, όπως το IP, HTTP, FTP, SMTP, POP3, IMAP, IMAP, SSL, που βοηθούν τους/τις εκπαιδευόμενους/-ες να κατανοήσουν ότι η επικοινωνία στο Διαδίκτυο διέπεται από διαφορετικούς κανόνες, ανάλογα με το είδος της επικοινωνίας, και να καταλάβουν τα συγκεκριμένα ακρώνυμα και την ορολογία. Επιπρόσθετα, οι γνώσεις αυτές είναι απαραίτητες στο πρακτικό κομμάτι της δημιουργίας διαδικτυακών εφαρμογών. Στη συνέχεια, γίνεται εισαγωγή στη γλώσσα περιγραφής HTML και του τρόπου εμφάνισης του μορφοποιημένου και ενεργού κειμένου σε κάποιον φυλλομετρητή και γίνεται διάκριση του τρόπου ανάπτυξης στατικών ιστοσελίδων και δυναμικών ιστοσελίδων, αναφέροντας σύντομα κάποια από τα εργαλεία που χρησιμοποιούνται για τις δυναμικές ιστοσελίδες. Στην υποενότητα αυτή, είναι σημαντικό να συζητηθούν θέματα που αφορούν θέματα συμπεριφοράς στο Διαδίκτυο, τόσο κατά την επικοινωνία όσο και κατά τον διαμοιρασμό πληροφοριών, για να αποκτήσουν οι εκπαιδευόμενοι/-ες την κατάλληλη κουλτούρα και να κατανοούν τους κινδύνους και τα προβλήματα που μπορούν να δημιουργηθούν.

2.1.Γ.2 | ΓΛΩΣΣΑ ΑΝΑΠΤΥΞΗΣ ΙΣΤΟΣΕΛΙΔΩΝ HTML

Σκοπός της υποενότητας είναι η απόκτηση δεξιοτήτων ανάπτυξης ιστοσελίδων και ιστοτόπων με τη χρήση της HTML, και πιο συγκεκριμένα της HTML5, αλλά και η εκμάθηση επεξεργαστών και ολοκληρωμένων περιβαλλόντων για την ανάπτυξη ιστοσελίδων και ιστοτόπων. Αρχικά, γίνεται μια ιστορική αναδρομή στη δημιουργία της HTML, ώστε να γνωρίζουν οι εκπαιδευόμενοι/-ες για τον καταλυτικό ρόλο του Τιμ Μπέρνερς Λι και να μάθουν για τις εκδόσεις της. Στη συνέχεια, μαθαίνουν για τη λογική των ετικετών και τα δομικά στοιχεία ενός εγγράφου HTML, δηλαδή το <HEAD> και το <BODY>. Μαθαίνουν τις βασικές ετικέτες <H1> έως <H5>, , <i>,
, <p>, τις λίστες, τους πίνακες και τις ετικέτες για το τμήμα <HEAD> και, με τη βοήθεια κάποιου απλού επεξεργαστή κειμένου, δημιουργούν έγγραφα HTML, ώστε να έχουν πλήρη κατανόηση της δομής και του τρόπου λειτουργίας των ετικετών. Στη συνέχεια, οι εκπαιδευόμενοι/-ες ενημερώνονται για επεξεργαστές WYSIWYG (What You See Is What You Get) και χρησιμοποιούν κάποιον αντίστοιχο επεξεργαστή HTML. Επιπρόσθετα, μαθαίνουν για τα ολοκληρωμένα περιβάλλοντα ανάπτυξης ιστοσελίδων και διαχείρισης ιστοτόπων, ώστε να κατανοήσουν ότι η διαχείριση ιστοτόπων είναι μια πολύπλοκη διαδικασία. Η υποενότητα συνεχίζει με την παρουσίαση των ετικετών, για ενσωμάτωση εικόνων και συνδέσμων, και των δυνατοτήτων της HTML για φόρμες. Ακολούθως, γίνεται εισαγωγή στην HTML5 και στις νέες ετικέτες για ήχο <audio>, βίντεο <video>, εικόνες <figure> κ.λπ. Γίνεται εκτενής παρουσίαση και χρήση των ετικετών <canvas> και <svg> για τη σχεδίαση σχημάτων.

2.1.Γ.3 | ΕΙΣΑΓΩΓΗ ΣΤΑ CSS

Η μαθησιακή υποενότητα «Εισαγωγή στα CSS» επικεντρώνεται στην κατανόηση των CSS (Cascading Style Sheets, διαδοχικά φύλλα ύφους ή επάλληλα φύλλα ύφους), που αποτελούν βασικό εργαλείο για τη διαμόρφωση και σχεδίαση ιστοσελίδων. Τα φύλλα CSS χρησιμοποιούνται για να ορίζουν την εμφάνιση και τη διάταξη των στοιχείων HTML. Αρχικά, οι εκπαιδευόμενοι/-ες μαθαίνουν για τη σύνταξη της CSS με τον προσδιορισμό του κατάλληλου στοιχείου της HTML και τις παραμέτρους μορφοποίησης (π.χ. `p { ... }`), αλλά και τον ορισμό κλάσεων και των ιδιοτήτων τους (π.χ. `#para1 { ... }`). Στη συνέχεια, γίνεται παρουσίαση του τρόπου χρήσης CSS, είτε εμβόλιμα σε ένα αρχείο HTML είτε μέσω ενός εξωτερικού αρχείου `.css`. Τα βασικά θέματα που παρουσιάζονται στη μαθησιακή υποενότητα αφορούν τις ιδιότητες της CSS για την εμφάνιση κειμένου (π.χ. `font`, `border`), τη διάταξη των ετικετών στη σελίδα (π.χ. `float`), το μέγεθος και τα περιθώρια μεταξύ των ετικετών (π.χ. `width`, `padding`). Με την ολοκλήρωση της ενότητας, οι εκπαιδευόμενοι/-ες θα γνωρίζουν τις ιδιότητες που αφορούν τα χρώματα, τις γραμματοσειρές, τις εικόνες, το `background`, τους πίνακες, τις δυνατότητες σύνδεσης και τη μορφή της ιστοσελίδας γενικά. Στους/Στις εκπαιδευόμενους/-ες είναι σημαντικό να δοθούν έτοιμα παραδείγματα αρχείων `css`, για να έρθουν σε επαφή με προχωρημένα παραδείγματα και να πειραματιστούν με αλλαγές σε αυτά τα αρχεία. Τέλος, προτείνεται στην ενότητα να χρησιμοποιηθούν HTML και CSS υπηρεσίες για επικύρωση των αρχείων, ώστε να εμφανίζονται τα αρχεία χωρίς λάθος σε οποιονδήποτε φυλλομετρητή.

2.1.Γ.4 | ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ ΑΝΑΠΤΥΞΗΣ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ

Το πρώτο θέμα της υποενότητας είναι η παρουσίαση τεχνικών διαχείρισης γραφικών και γενικότερα πολυμεσικού υλικού από διαδικτυακές εφαρμογές. Αρχικά, εξετάζεται η διαχείριση εικόνων. Γίνεται παρουσίαση της ετικέτας ενσωμάτωσης εικόνων και των επιλογών της, καθώς και των επιλογών της CSS για τη μορφοποίηση εικόνων. Ακολουθεί η διαχείριση ήχου, και συγκεκριμένα η ετικέτα ενσωμάτωσης ήχου και οι επιλογές της, καθώς και η ετικέτα ενσωμάτωσης βίντεο και οι επιλογές της. Το δεύτερο θέμα της υποενότητας είναι οι μετασχηματισμοί σε στοιχεία HTML, με τη χρήση της ιδιότητας `transform` της CSS. Παρουσιάζεται η σύνταξη της `transform` και οι μετασχηματισμοί που υποστηρίζει, δηλαδή η μετατόπιση (`translation`), η περιστροφή (`rotation`) και η αλλαγή κλίμακας (`scaling`). Στη συνέχεια, παρουσιάζεται η ιδιότητα `transition` της CSS, η οποία αλλάζει τις ιδιότητες ενός HTML στοιχείου (π.χ. αλλαγή χρώματος, μεγέθους, θέσης) όταν συμβαίνει μια ενέργεια (π.χ. πέρασμα ποντικιού πάνω από ένα κουμπί). Συγκεκριμένα, παρουσιάζεται η σύνταξη της `transition` και δίνονται παραδείγματα χρήσης της. Τέλος, παρουσιάζεται η ιδιότητα `flexbox` της CSS, η οποία χρησιμοποιείται για τη διαμόρφωση της διάταξης των στοιχείων ενός αρχείου HTML. Παρουσιάζεται η σύνταξη της `flexbox` και δίνονται παραδείγματα χρήσης της. Το επόμενο θέμα είναι η παρουσίαση των διεπαφών προγραμματισμού εφαρμο-

γών (Application Programming Interfaces, APIs) για την HTML. Τα APIs είναι ιδιαίτερα χρήσιμα, διότι επιτρέπουν σε μια ιστοσελίδα, γραμμένη σε HTML, να επικοινωνεί με άλλα συστήματα. Παρουσιάζονται δημοφιλή APIs, όπως τα APIs της Google (π.χ. Google maps API) και τα APIs κοινωνικών δικτύων. Επίσης, παρουσιάζονται τεχνικές ενσωμάτωσης APIs σε μια ιστοσελίδα. Τέλος, γίνεται αναφορά στη σημασία του τεκμηριωμένου κώδικα και παρουσιάζονται τεχνικές τεκμηρίωσης, όπως σχόλια και περιγραφικοί τίτλοι.

2.1.Γ.5 | ΧΡΗΣΙΜΑ ΕΡΓΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ

Το πρώτο θέμα της υποενότητας αυτής είναι η παρουσίαση δημοφιλών πλαισίων εργασίας (frameworks) για την ανάπτυξη εφαρμογών Διαδικτύου. Το framework είναι μια συλλογή έτοιμου επαναχρησιμοποιήσιμου κώδικα, που διευκολύνει τους/τις προγραμματιστές/-τριες κατά την ανάπτυξη λογισμικού. Στο πλαίσιο του μαθήματος, προτείνεται να παρουσιαστούν τα ακόλουθα frameworks: το Bootstrap (<https://getbootstrap.com/>), το οποίο χρησιμοποιείται για την ανάπτυξη ανταποκρισιμων (responsive) ιστοτόπων και παρέχει πλήθος εργαλείων για τη βελτίωση της εμφάνισης των ιστοσελίδων, και το Foundation (<https://get.foundation/>), το οποίο δίνει μεγαλύτερη ελευθερία κατά τον σχεδιασμό του ιστοτόπου. Το δεύτερο θέμα της υποενότητας είναι η παρουσίαση των λογισμικών σύνταξης περιεχόμενου ιστού. Τα λογισμικά αυτά επιτρέπουν στον/στη χρήστη/-τρια την ανάπτυξη κώδικα μέσα σε ένα γραφικό περιβάλλον εργασίας, που συνδυάζει διάφορα εργαλεία και λειτουργίες προς διευκόλυνση των προγραμματιστών/-τριών. Προτείνεται να παρουσιαστεί στους/στις εκπαιδευόμενους/-ες ένα εκ των ακόλουθων εργαλείων: το Visual Studio Code (<https://code.visualstudio.com/>), το Sublime Text (<https://www.sublimetext.com/>) ή το Atom (<https://github.com/atom/atom>). Το επόμενο θέμα της υποενότητας είναι τα online εργαλεία απόθεσης κώδικα και ελέγχου εκδόσεων. Γίνεται σύντομη αναφορά στα ψηφιακά αποθετήρια και στη σημασία τους για τους νέους επαγγελματίες. Ιδιαίτερα τονίζεται το πιο δημοφιλές online αποθετήριο για την απόθεση κώδικα, το Github (<https://github.com/>). Γίνεται αναφορά στο Github και παρουσιάζεται η διαδικασία δημιουργίας αποθετηρίου (repository) για την απόθεση κώδικα, καθώς και της τεκμηρίωσης αυτού. Στη συνέχεια, η ηλεκτρονική διεύθυνση του αποθετηρίου μπορεί να χρησιμοποιηθεί από τον/την εκπαιδευόμενο/-η (για παράδειγμα στο βιογραφικό του).

2.1.Γ.6 | ΤΕΧΝΙΚΑ ΘΕΜΑΤΑ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ

Η υποενότητα ξεκινά με την έννοια του domain name (όνομα τομέα), ο τρόπος με τον οποίο ελέγχεται η διαθεσιμότητα ενός domain name (π.χ. μέσω <http://www.gr>), και ο τρόπος με τον οποίο μπορεί να κατοχυρωθεί ένα domain name (μέσω εταιριών registrars). Επίσης, παρουσιάζεται η έννοια του web hosting (φιλοξενία ιστοσελίδων). Γίνεται αναφορά στο σύστημα client-server (πελάτη-εξυπηρετητή) και στην

έννοια του web server. Τέλος, παρουσιάζονται τα κριτήρια, βάσει των οποίων επιλέγεται η εταιρεία και το πακέτο web hosting. Το επόμενο θέμα είναι το Responsive Web Design (RWD), μια νέα τάση στη σχεδίαση ιστοσελίδων, ώστε να είναι κατάλληλες για πολλές συσκευές (υπολογιστές, tablets, smartphones). Παρουσιάζεται ο τρόπος με τον οποίο ελέγχεται αν ένας ιστότοπος είναι φιλικός προς κινητά (μέσω του Mobile Friendly Test-Google Search Console) και ο τρόπος βελτιστοποίησής του, βάσει παρατηρήσεων και προτάσεων. Τέλος, τεκμηριώνεται η σημασία της ασφάλειας των ιστοτόπων και της προστασίας των δεδομένων εκ σχεδιασμού (κατά τη σχεδίαση της εφαρμογής) και εξορισμού (κατά τη λειτουργία της εφαρμογής). Παρουσιάζονται, ακόμη, οι βασικές τεχνικές προστασίας δεδομένων (κρυπτογραφία, διαβαθμισμένη πρόσβαση χρηστών και προώθηση της ανωνυμίας) και δημοφιλή εργαλεία εντοπισμού προβλημάτων ασφάλειας σε ιστότοπους. Προτείνεται το OWASP ZAP (<https://www.zaproxy.org/>), ένα εργαλείο ανοικτού κώδικα, το οποίο χρησιμοποιείται για την ανίχνευση προβλημάτων ασφάλειας τόσο σε ιστοσελίδες όσο και σε web εφαρμογές. Εναλλακτικά, προτείνουμε το Nessus (<https://www.tenable.com/products/nessus>), το οποίο παρέχει δωρεάν εκπαιδευτική έκδοση.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Αβούρης, Ν. & Σιντόρης, Χ. (2023). *Προγραμματισμός Διαδικτύου*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://dx.doi.org/10.57713/kallipos-337>
2. Lemay, L., Colburn, R. & Kyrnin, J. (2016). *Πλήρες Εγχειρίδιο HTML5, CSS και JavaScript*. Αθήνα: Εκδόσεις Μ. Γκιούρδα.

Συμπληρωματικές

1. Βενιέρης, Ι. Σ. (2003). *Σημειώσεις για τη γλώσσα HTML (HyperText Markup Language) [Ηλεκτρονική έκδοση]*. Αθήνα: Εθνικό Μετσόβιο Πολυτεχνείο, Εργαστήριο Ευφύων Επικοινωνιών και Δικτύων Ευρείας Ζώνης. <http://www.icbnet.ntua.gr/Mathimata/IntroInternetTech/HTML%20Notes.pdf>
2. Χειλάς, Κ., Βακαλούδης, Α. & Πολίτης, Α. (2015). *Η γλώσσα HTML. Πίνακες, Λίστες, Φόρμες*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/1774>

2.1.Δ • ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Ι (C11)-ΑΛΓΟΡΙΘΜΙΚΗ ΚΑΙ ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Γλώσσα προγραμματισμού Ι (C11)-Αλγοριθμική και δομές δεδομένων» εισάγει τους/τις εκπαιδευόμενους/-ες στον δομημένο, διαδικαστικό προγραμματισμό με τη γλώσσα προγραμματισμού C σε μια από τις πιο πρόσφατες προτυποποιήσεις της (2011). Ο/Η εκπαιδευόμενος/-η θα έρθει σε επαφή με τις πιο κλασικές τεχνικές αλγορίθμων, την εκμάθηση βασικών και προηγμένων αλγοριθμικών τεχνικών και τη σύνταξη ολοκληρωμένων εφαρμογών περιβάλλοντος κονσόλας. Σκοπός της μαθησιακής ενότητας είναι να εισαγάγει τους/τις εκπαιδευόμενους/-ες στη γλώσσα προγραμματισμού C11, ώστε να αποκτήσουν τις βάσεις για τη δημιουργία δυναμικών προγραμματιστικών εφαρμογών. Ειδικότερα, παρουσιάζεται το συντακτικό της γλώσσας C11 και αναφέρονται βασικές έννοιες, όπως είσοδος και έξοδος δεδομένων, σχόλια, δηλώσεις, τύποι μεταβλητών, τελεστές (αριθμητικοί, σύγκρισης, λογικοί, δυαδικοί, ολίσθησης), εκχώρηση δεδομένων, τύποι δεδομένων, εισαγωγή δεδομένων στην κύρια συνάρτηση (main), πίνακες, αλγόριθμοι επεξεργασίας πινάκων (συγχώνευση, διάσπαση, διαγραφή/εισαγωγή στοιχείων, αναζήτηση, ταξινόμηση), συμβολοσειρές, συναρτήσεις, μακροεντολές προεπεξεργαστή, βιβλιοθήκες, βρόγχοι, δομές επιλογής και επανάληψης, απαριθμητές, δομές δεδομένων (structures), μετατροπή τύπων (casting), δέσμευση και αποδέσμευση μνήμης, κλήση μέσω τιμής και κλήση μέσω αναφοράς, εμβέλεια, λογικά και συντακτικά σφάλματα, επεξεργασία αρχείων κειμένου (read, write, append), εκσφαλμάτωση και απόδοση. Μέσα από τα παραπάνω, ο/η εκπαιδευόμενος/-η θα αποκτήσει βασικές γνώσεις και δεξιότητες για τον προγραμματισμό πολύπλοκων αλγορίθμων, ώστε να μεταβεί σε γλώσσες αντικειμενοστρεφούς προγραμματισμού στα επόμενα εξάμηνα.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναγνωρίζουν τις βασικές προγραμματιστικές αρχές, βάσει των οποίων έχει υλοποιηθεί μια εφαρμογή,
- Επιλέγουν τα κατάλληλα στοιχεία (μακροεντολές, μεταβλητές, συναρτήσεις κ.ά.) που θα χρησιμοποιήσουν στις εφαρμογές που θα αναπτύξουν,
- Αναγνωρίζουν τη σημασία του διαδικαστικού προγραμματισμού και του τεκμηριωμένου κώδικα,
- Αναπτύσσουν εφαρμογές με χρήση της γλώσσας προγραμματισμού C11,
- Συνεισφέρουν με πρωτότυπο κώδικα στην κοινότητα των προγραμματιστών εφαρμογών,

- Τροποποιούν υπάρχοντα κώδικα, ανάλογα με τις προδιαγραφές των εφαρμογών που αναπτύσσουν,
- Αποφασίζουν για τη χρήση ή μη κατάλληλων βιβλιοθηκών, σχετικών με την ανάπτυξη εφαρμογών,
- Συνεργάζονται με άλλους/-ες εκπαιδευόμενους/-ες, συγκροτώντας ομάδες, με σκοπό τη δημιουργία και την επίλυση προβλημάτων των εφαρμογών,
- Υιοθετούν θετική στάση ως προς τις υπάρχουσες τεχνολογίες ανάπτυξης εφαρμογών.

Βασικές έννοιες και λέξεις-κλειδιά

Σταθερές/Μεταβλητές / Πηγαίος κώδικας / Μεταγλωττιστής/Διερμηνευτής / Στατικές/Δυναμικές δομές δεδομένων / Δομές επιλογής/Δομές επανάληψης / Σχόλια Εκσφαλμάτωση / Αλγόριθμοι αναζήτησης / Αλγόριθμοι ταξινόμησης / Αρχεία κειμένου / BGI Graphics.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (1), Εργαστήριο (5), Σύνολο (6).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΔΟΜΙΚΑ ΣΤΟΙΧΕΙΑ ΠΡΟΓΡΑΜΜΑΤΟΣ
2	ΤΕΛΕΣΤΕΣ ΠΡΟΓΡΑΜΜΑΤΟΣ
3	ΔΟΜΕΣ ΕΛΕΓΧΟΥ ΡΟΗΣ ΠΡΟΓΡΑΜΜΑΤΟΣ
4	ΡΟΠΟΓΕΝΝΗΤΡΙΑ ΠΑΡΑΓΩΓΗΣ ΨΕΥΔΟΤΥΧΑΙΩΝ ΑΡΙΘΜΩΝ-SRAND()/RAND() ΜΕ ΤΗ ΧΡΗΣΗ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ TIME.H
5	ΣΤΑΤΙΚΟΙ ΠΙΝΑΚΕΣ ΜΙΑΣ/ΔΥΟ ΔΙΑΣΤΑΣΕΩΝ ΚΑΙ ΠΙΝΑΚΕΣ ΜΕΓΑΛΥΤΕΡΗΣ ΔΙΑΣΤΑΣΗΣ
6	ΣΥΝΑΡΤΗΣΕΙΣ-ΔΕΙΚΤΕΣ
7	ΑΛΓΟΡΙΘΜΟΙ ΤΑΞΙΝΟΜΗΣΗΣ/ΑΝΑΖΗΤΗΣΗΣ
8	ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ
9	ΑΡΧΕΙΑ ΚΕΙΜΕΝΟΥ (BINARY, ANSI/ASCII)
10	ΓΡΑΦΙΚΑ ΣΤΗ C11

Κείμενο αναφοράς

2.1.Δ.1 | ΔΟΜΙΚΑ ΣΤΟΙΧΕΙΑ ΠΡΟΓΡΑΜΜΑΤΟΣ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τις έννοιες του αλγορίθμου και των κριτηρίων του. Αναφορικά με το χρησιμοποιούμενο IDE (Integrated Development Environment), θα πρέπει να υποστηρίξει τις αλλαγές που περιέχει το πρότυπο C11 του 2011 και προτείνεται ένα από τα Dev C++ 5.11/ CodeBlocks και όχι τα DevC++ 6/Visual Studio. Επίσης, προτείνεται η κατάληξη των

πηγαίων αρχείων κώδικα –σε κάποιες περιπτώσεις– να μην έχουν κατάληξη .c αλλά .cpp, ώστε να ενσωματώνουν τις λειτουργίες του προτύπου C11, όπως είναι η νέα δομή επανάληψης for(each) κ.λπ. Επιπλέον, οι σπουδαστές/-τριες θα διδαχθούν τις διαφορές μεταγλωττιστή (προτείνεται ο GCC 4.9.2 64-bit) και διερμηνευτή. Στη συνέχεια, θα διδαχθούν τη δημιουργία εκτελέσιμου αρχείου, τη σχέση εκτελέσιμου αρχείου και αρχείου πηγαίου κώδικα, τις μακροεντολές προεπεξεργαστή και τα προκαθορισμένα αρχεία βιβλιοθήκης (.h). Στη συνέχεια, οι σπουδαστές/-τριες θα διδαχθούν τη χρήση των μεταβλητών αποθήκευσης (σταθερών [χρήση εντολών define/const] και μη), όπως είναι οι char, wchar_t, int, short, long, unsigned, size_t, float, double, bool (χρήση stdbool.h), τις μεθόδους στατικού casting και τις συμβάσεις ονοματοδοσίας (snake_case, χρήση ουσιαστικών). Επιπλέον, θα διδαχθούν συναρτήσεις εισόδου/εξόδου δεδομένων των βιβλιοθηκών stdio.h/conio.h, όπως είναι οι printf, scanf, puts, gets, getch, putch, getchar, putchar, ενώ θα γίνει και χρήση των console system commands (Microsoft/UNIX), όπως είναι οι “CHCP 1253” για εμφάνιση ελληνικού character code-page, “PAUSE”, “CLS”, “clear” κ.ά. Τέλος, θα διδαχθούν τη συνάρτηση main και τη λογική της επιστρεφόμενης τιμής της στο λειτουργικό σύστημα, ενώ θα αναλυθούν οι τρόποι σχολιασμού κώδικα και ο ρόλος της εκσφαλμάτωσης (debugging) των λαθών του προγράμματος (συντακτικών, λογικών).

2.1.Α.2 | ΤΕΛΕΣΤΕΣ ΠΡΟΓΡΑΜΜΑΤΟΣ

Σε αυτή τη μαθησιακή υποενοότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τα είδη τελεστών που έχει η γλώσσα προγραμματισμού C11, ξεκινώντας με τους τελεστές διαφυγής (escape sequence characters: \n, \t, \a, \\\, \", \', \b, \r, \f, \v, %%) και με τους τελεστές εμφάνισης (%x, %X, %o, %d, %f, %lf, %c, %s). Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν τους αριθμητικούς τελεστές –δυναμικοί/μοναδιαίοι– (+, -, *, /, %, =, +=, -=, *=, /=, %+=, ++, --), με έμφαση να δίνεται στον ρόλο του τελεστή «/» κατά τη χρήση του από ακέραιες μεταβλητές, έναντι της χρήσης του από πραγματικές μεταβλητές απλής/διπλής ακρίβειας, καθώς και στις περιπτώσεις κατά τις οποίες επιτρέπεται η χρήση του αριθμητικού τελεστή «%». Στη συνέχεια, θα παρουσιαστούν ονομαστικά οι συγκριτικοί τελεστές (<, <=, >, >=, !=, ==) και οι λογικοί τελεστές (&&/and, ||/or, !/not), μαζί με τους αντίστοιχους πίνακες αληθείας τους. Επιπλέον, θα παρουσιαστούν οι τελεστές Bitwise/τελεστές ολίσθησης (&, |, ^, <<, >>, ~) –αριθμητική και λογική χρήση τους– μαζί με τη διαφορά χρήσης τους από τους προαναφερθέντες λογικούς τελεστές, ενώ θα παρουσιαστεί και η χρήση του τελεστή κόμμα (,) σε όλες τις χρήσεις του. Επιπρόσθετα, θα γίνει μια σύγκριση της σημασίας των τελεστών όταν δεν χρησιμοποιούνται παρενθέσεις, κάνοντας αναφορά στην προτεραιότητα κάθε είδους, συγκρινόμενου με τα υπόλοιπα. Τέλος, θα γίνει παρουσίαση της βιβλιοθήκης math.h, μαζί με τις πλέον σημαντικές συναρτήσεις που ενυπάρχουν σε αυτή: sqrt, pow, abs/fabs, ceil, floor, fmod κ.ά., ενώ η μαθησιακή ενότητα θα ολοκληρωθεί με τη βιβλιοθήκη μέγιστων/ελάχιστων τιμών κάθε είδους μεταβλητής limits.h.

2.1.Δ.3 | ΔΟΜΕΣ ΕΛΕΓΧΟΥ ΡΟΗΣ ΠΡΟΓΡΑΜΜΑΤΟΣ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τα είδη των δομών επιλογής και τα αντίστοιχα διαγράμματα ροής δεδομένων κάθε είδους δομής ελέγχου ροής, ξεκινώντας με τις δομές επιλογής, όπως είναι η απλή/σύνθετη επιλογή (if, if-else, if-else-if-else κ.ά.), ο τριαδικός τελεστής/τελεστής υπό συνθήκη –ternary operator/if conditional (: ? -), switch-case-break- default, μαζί με τις περιπτώσεις κατά τις οποίες η εντολή switch δεν λειτουργεί. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν τα τέσσερα από τα συνολικά έξι είδη των δομών επανάληψης. Πιο συγκεκριμένα, θα γίνει θεωρητική αναφορά στη δομή επανάληψης goto και στα προβλήματα που αυτή παρουσιάζει στο δομημένο ακολουθιακό διαδικαστικό προγραμματισμό, και στη συνέχεια θα γίνει αναλυτική παρουσίαση των δομών επανάληψης while, for και do-while, καθώς και των περιπτώσεων καταλληλότητας κάθε δομής. Επιπλέον, θα πραγματοποιηθούν εργαστηριακές ασκήσεις που θα απαιτούν τη μετατροπή μιας δομής επανάληψης στις υπόλοιπες δύο, καθώς και τα λογικά διαγράμματα ροής δεδομένων για κάθε δομή επανάληψης. Κατόπιν, θα πραγματοποιηθούν συνδυαστικές ασκήσεις απλών/εμφωλευμένων δομών επιλογής/επανάληψης, και θα αξιοποιηθούν οι λογικοί τελεστές ή/και τελεστές σύγκρισης. Επιπρόσθετα, θα πραγματοποιηθεί χρήση των εντολών exit και return, ενώ θα γίνει αντιπαραβολή με τις εντολές break/continue. Τέλος, ο/η σπουδαστής/-τρια θα διδαχθεί τον τελεστή EOF-End Of File, καθώς και τους τρόπους μέσω των οποίων θα τερματίζεται η είσοδος δεδομένων από το πληκτρολόγιο (Ctrl+D/Ctrl+Z). Επιπλέον, θα πραγματοποιηθεί αναφορά στις δύο εναπομείνουσες δομές επανάληψης που θα αναλυθούν αργότερα: στη for(each) και στην αναδρομική γραφή συνάρτησης, ενώ θα πραγματοποιηθεί αναφορά στη βιβλιοθήκη assert.h ως αυτοματοποιημένου εργαλείου αμυντικού προγραμματισμού.

2.1.Δ.4 | ΡΟΠΟΓΕΝΝΗΤΡΙΑ ΠΑΡΑΓΩΓΗΣ ΨΕΥΔΟΤΥΧΑΙΩΝ ΑΡΙΘΜΩΝ-SRAND()/RAND() ΜΕ ΤΗ ΧΡΗΣΗ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ TIME.H

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη χρήση της βιβλιοθήκης time.h και τη χρήση των κυριότερων συναρτήσεων της – π.χ. time(NULL) ή time(0). Στη συνέχεια, θα παρουσιαστεί στους/στις εκπαιδευόμενους/-ες η ροπογεννήτρια συνάρτηση παραγωγής ψευδοτυχαίων αριθμών (rand), και οι λόγοι για τους οποίους χρησιμοποιείται η srand πριν από την κλήση της εντολής rand μέσα σε ένα πρόγραμμα. Ιδιαίτερη προσοχή θα δοθεί στον αριθμό των κλήσεων της εντολής srand πριν από την κλήση της αντίστοιχης rand, και στη σύνδεση της srand με την εντολή time(NULL) κατά την κλήση της πρώτης. Κατόπιν, θα γίνει παρουσίαση των τρόπων με τους οποίους θα μπορεί να μεταβληθεί το πεδίο τιμών εξόδου που έχει η rand. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα συνδυάσουν τις δομές επιλογής και επανάληψης που έχουν διδαχτεί έως τώρα, ώστε να δημιουργήσουν προγράμματα που να υλοποιούν τυχαία πειράματα, όπως τη ρίψη ενός εξάεδρου ζαριού, ή τη ρίψη ενός νομίσματος, ή τη διαδικασία που ακολουθείται κατά την επι-

λογή ενός τυχαίου τραπουλόχαρτου/αριθμού ρουλέτας σε μια εφαρμογή τυχερών παιχνιδιών. Ειδικότερα, για την υλοποίηση παραδειγμάτων σχετικών με την επιλογή τραπουλόχαρτου, ενδείκνυται, για λόγους εικαστικής αρμονίας, η χρήση των τελεστών χαρακτήρα '\3', '\4', '\5' και '\6' κατά την εκτύπωση των συμβόλων κάθε κάρτας, καθώς και η εντολή συστήματος color, η οποία θα τροποποιεί το χρώμα που θα βλέπει ο/η πελάτης/-ισσα. Προτείνεται η γραμματοσειρά Raster του command prompt, ώστε να εμφανίζονται τα σύμβολα που θα προκύψουν από τους τελεστές χαρακτήρα που αναφέρθηκαν προηγουμένως.

2.1.Δ.5 | ΣΤΑΤΙΚΟΙ ΠΙΝΑΚΕΣ ΜΙΑΣ/ΔΥΟ ΔΙΑΣΤΑΣΕΩΝ ΚΑΙ ΠΙΝΑΚΕΣ ΜΕΓΑΛΥΤΕΡΗΣ ΔΙΑΣΤΑΣΗΣ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τους λόγους για τους οποίους χρησιμοποιούνται οι στατικές δομές δεδομένων ενός σταθερού αριθμού μεταβλητών του ίδιου είδους. Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες εξοικειώνονται με τη μορφή των εντολών με τις οποίες γίνεται η δήλωση ενός πίνακα, και με τους διαφορετικούς τρόπους με τους οποίους δηλώνεται η διάσταση ενός πίνακα μιας, δύο ή περισσότερων διαστάσεων. Επιπλέον, γίνεται αναφορά στις δομές επανάληψης που είναι κατάλληλες για να διαχειρίζονται πίνακες, μαζί με μια εκτενή αναφορά στη νέα δομή επανάληψης for(each) και στους περιορισμούς της. Επιπλέον, μελετάται η δέσμευση της μνήμης για έναν πίνακα που δηλώνεται, οι τρόποι με τους οποίους γίνεται αρχικοποίηση ενός πίνακα και ο τρόπος με τον οποίο πραγματοποιείται η προσπέλαση στα στοιχεία αυτού. Στη συνέχεια, πραγματοποιούνται ασκήσεις εφαρμογής στη χρήση πινάκων, όπως η διαπέραση πίνακα, η συγχώνευση και ο διαχωρισμός πινάκων, η εισαγωγή και η διαγραφή στοιχείου από πίνακα. Στη συνέχεια, οι εκπαιδευόμενοι/-ες διδάσκονται μια σημαντική κατηγορία πινάκων, τους πίνακες χαρακτήρων (strings, αλφαριθμητικά, συμβολοσειρές) μέσω της χρήσης της βιβλιοθήκης string.h και των έτοιμων συναρτήσεων που ενυπάρχουν σε αυτή, όπως των strlen/strnlen, strcpy/strncpy, strcmp/strncmp, strcat/strncat, memcpy, strstr κ.ά. Επίσης, διδάσκεται η χρήση της συνάρτησης gets() για είσοδο αλφαριθμητικού και τα πλεονεκτήματά της. Τέλος, γίνεται αναφορά στη βιβλιοθήκη ctype.h και στη συσχέτισή της με τους πίνακες χαρακτήρων μέσα από τις έτοιμες συναρτήσεις, όπως τις toupper, tolower, τις isdigit, isalpha, μαζί με τις atoi, atol και atof της βιβλιοθήκης stdlib.h. Τέλος, γίνεται αναφορά στη χρήση των define/const και του παραμετροποιήσιμου τρόπου σύνταξης και αναφοράς σε μεγέθη πινάκων.

2.1.Δ.6 | ΣΥΝΑΡΤΗΣΕΙΣ-ΔΕΙΚΤΕΣ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τον τρόπο δημιουργίας ενός υποπρογράμματος/διαδικασίας, μέσα από τη χρήση αυτοσχέδιων συναρτήσεων που θα υλοποιεί ο/η εκπαιδευόμενος/-η. Αρχικά, θα διδαχτεί η εμβέλεια (scope) των μεταβλητών και η διάκρισή τους σε καθολικές, τοπικές και στατικές, και οι

κανόνες που διέπουν τη σύνταξη συναρτήσεων (είδος επιστρεφόμενης τιμής, πλήθος/είδος παραμέτρων/ορισμάτων εισόδου σε συναρτήσεις, συμβάσεις ονοματοδοσίας (camelCase τρόπος δήλωσης, χρήση ρημάτων) κ.λπ. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν τις αναδρομικές (recursive) συναρτήσεις, κάνοντας αναφορά στον τρόπο λειτουργίας τους και επισημαίνοντας σε ποιες περιπτώσεις θα είναι προτιμότερη η χρήση τους, καθώς και στις συναρτήσεις που δέχονται πίνακα ως είσοδο, ενώ ακολούθως θα διδαχθούν συναρτήσεις που δέχονται άλλες συναρτήσεις ως είσοδο. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν τον τρόπο εισόδου δεδομένων ως παραμέτρους στη συνάρτηση `main()`, τα ορίσματα `argc` και `argv` της συνάρτησης `main()` και τον τρόπο που αυτά λειτουργούν. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν την κλήση μεταβλητών μέσω τιμής και μέσω αναφοράς (call by value/reference) και θα γίνει αναφορά στους τελεστές `star (*)` και `ampersand (&)`, οι οποίοι χρησιμοποιούνται για τη δημιουργία δεικτών μεταβλητών, και στην αντιστοίχισή τους σε κάποια διεύθυνση μνήμης. Επιπλέον, θα παρουσιαστεί η σχέση δεικτών/πινάκων, οι πράξεις μεταξύ δεικτών, οι συναρτήσεις με είσοδο δείκτη και που επιστρέφουν δείκτη, οι δείκτες δεικτών και, τέλος, η δημιουργία δυναμικής δέσμευσης/αποδέσμευσης μνήμης μέσω των εντολών `malloc/calloc/realloc`, συνδυαζόμενες με την εντολή `sizeof`, καθώς και την εντολή αποδέσμευσης μνήμης `free`. Όλα αυτά θα βοηθήσουν στη δημιουργία πινάκων μιας ή και περισσότερων διαστάσεων με δυναμικό και όχι στατικό μέγεθος.

2.1.Δ.7 | ΑΛΓΟΡΙΘΜΟΙ ΤΑΞΙΝΟΜΗΣΗΣ/ΑΝΑΖΗΤΗΣΗΣ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν αλγορίθμους ταξινόμησης ενός πίνακα και αναζήτησης ενός στοιχείου από έναν πίνακα. Πιο συγκεκριμένα, στην ταξινόμηση των στοιχείων ενός πίνακα οι εκπαιδευόμενοι/-ες διδάσκονται ότι ο κάθε αλγόριθμος ταξινόμησης καλείται να ταξινομήσει τα στοιχεία του υπό επεξεργασία πίνακα είτε σε αύξουσα είτε σε φθίνουσα σειρά. Στην κατηγορία αυτή, κρίνεται σκόπιμη η μελέτη των γνωστών αλγορίθμων «Ταξινόμηση Φυσαλίδας (Bubble Sort)», «Ταξινόμηση με Επιλογή (Selection Sort)», «Ταξινόμηση με Εισαγωγή (Insertion Sort)», «Ταξινόμηση με Συγχώνευση (Merge Sort)» και «Γρήγορη Ταξινόμηση (Quick Sort)», πραγματοποιώντας και μια συνολική σύγκριση των παραπάνω αλγορίθμων, εφαρμόζοντάς τους σε όμοια data sets. Τα κριτήρια σύγκρισης αφορούν τον αριθμό προσβάσεων μέσα στον πίνακα και τον αριθμό των συγκρίσεων που πραγματοποιήθηκαν σε κάθε αλγόριθμο, και κάθε φορά αποτιμάται ο αλγόριθμος που είχε το μικρότερο αριθμό και στα δύο κριτήρια. Στη συνέχεια, οι εκπαιδευόμενοι/-ες διδάσκονται τρόπους αναζήτησης δεδομένων μέσα σε έναν πίνακα. Πιο συγκεκριμένα, στην αναζήτηση ενός στοιχείου μέσα σε έναν πίνακα, διδάσκονται ότι ο αλγόριθμος καλείται να εντοπίσει αν το ζητούμενο στοιχείο υπάρχει ή όχι στον πίνακα. Στο πλαίσιο της αναζήτησης ενός στοιχείου, οι εκπαιδευόμενοι/-ες διδάσκονται τους αλγορίθμους «Σειριακής Αναζήτησης (Linear Search)», ενώ παράλληλα διδάσκονται και τον αλγόριθμο της «Διαδικής Αναζήτησης (Binary

Search)», πραγματοποιώντας αναφορά στα πλεονεκτήματα και στα μειονεκτήματα του κάθε αλγορίθμου αναζήτησης. Οι παραπάνω αλγόριθμοι ταξινόμησης και αναζήτησης θα διδαχθούν εντός ή/και εκτός κατάλληλων συναρτήσεων, εφαρμοζόμενοι σε πίνακες μιας και δύο διαστάσεων. Ειδικότερα, για τους αλγορίθμους ταξινόμησης, θα διδαχθεί η χρήση βοηθητικής συνάρτησης αντιμετάθεσης τιμών 2 στοιχείων – π.χ. `void swapTwoValues(..., ...)`.

2.1.Δ.8 | ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν την έννοια της σύνθετης μεταβλητής, η οποία διαθέτει μια ή και περισσότερες από τις αρχέτυπες (primitive) μεταβλητές ως μέλη της. Πιο αναλυτικά, θα πραγματοποιηθεί θεωρητική ανάλυση στη δομή δεδομένων `union`, καθώς και στον λόγο εγκατάλειψής της ως προβληματικής ως προς τη δέσμευση μνήμης. Στη συνέχεια, θα γίνει πλήρης περιγραφή της δεσμευμένης λέξης `enum`, η οποία δηλώνει απαριθμητές, μαζί με τους κανόνες προκαθορισμένων τιμών εντός της λίστας ενός απαριθμητή, ενώ παράλληλα θα παρουσιαστούν και τεχνικές αλλαγής των τιμών ενός απαριθμητή. Στη συνέχεια, θα πραγματοποιηθεί η διδασκαλία της δήλωσης δομής δεδομένων `struct`, μαζί με το προσδιοριστικό `typedef` και τις διαφορές που προκύπτουν με/χωρίς αυτό, κατά τη δήλωση μιας `struct` δομής δεδομένων. Επιπλέον, θα διδαχθούν κανόνες ονοματοδοσίας μιας δομής δεδομένων (π.χ. `CamelCase` τρόπος δήλωσης, με το πρώτο γράμμα κάθε λέξης κεφαλαίο και όχι από τη δεύτερη λέξη και μετά, όπως συνέβαινε στις συναρτήσεις και χρήση ουσιαστικών), ενώ θα γίνει ανάλυση του τρόπου δήλωσης μιας δομής δεδομένων μέσα στη μνήμη και στους λόγους καλύτερης διαλειτουργικότητας μιας εφαρμογής που χρησιμοποιεί δομές δεδομένων εν συγκρίσει με απλές –μη ομαδοποιημένες– μεταβλητές. Στη συνέχεια, θα γίνει αναφορά στον τελεστή τελείας (.) που χρησιμοποιείται για να αναφερθούμε στο μέλος μιας δομής δεδομένων, ενώ θα γίνει αναλυτική αναφορά και στον τελεστή βέλος (→), ο οποίος χρησιμοποιείται για τη δήλωση μελών δεικτών σε δομές δεδομένων. Τέλος, θα πραγματοποιηθεί αναφορά στις λίστες της C11, αναλύοντάς τις σε συνδεδεμένες και μη συνδεδεμένες λίστες, ενώ θα αναφερθούν και τα πλεονεκτήματα των λιστών έναντι των πινάκων.

2.1.Δ.9 | ΑΡΧΕΙΑ ΚΕΙΜΕΝΟΥ (BINARY, ANSI/ASCII)

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους αλληλεπίδρασης μιας προγραμματιστικής εφαρμογής με αρχεία κειμένου. Πιο αναλυτικά, θα διδαχθούν το είδος μεταβλητών δείκτη αρχείου `FILE`, καθώς και των συναρτήσεων `fopen` και `fclose`, οι οποίες ανοίγουν και κλείνουν την πρόσβαση σε ένα αρχείο κειμένου στον σκληρό δίσκο αντίστοιχα. Στη συνέχεια, θα πραγματοποιηθεί αναφορά στις εντολές εισόδου και εξόδου δεδομένων σε αρχεία κειμένου, όπως είναι οι `fprintf`, `fscanf`, `fgets`, `fputs` κ.ά., ενώ στη συνέχεια οι εκπαιδευόμενοι/-ες θα διδα-

χθούν και τρόπους χειρισμού αρχείων κειμένου μέσω των εντολών `fseek`, `ftell`, `rewind` κ.λπ. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν τους τρεις διακριτούς τρόπους πρόσβασης σε αρχεία κειμένου: την ανάγνωση δεδομένων από αρχεία κειμένου ("`r`"-read mode), καταστροφική εγγραφή δεδομένων σε αρχεία κειμένου ("`w`"-write mode), όπου αν το αρχείο κειμένου υπάρχει ήδη, τότε διαγράφεται και δημιουργείται ένα νέο, κενό αρχείο, και την επέκταση ενός αρχείου κειμένου, χωρίς να χαθεί η αποθηκευμένη πληροφορία ("`a`"-append mode). Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν και τρόπους σύνθετης πρόσβασης σε αρχεία κειμένου, μέσω των προσδιοριστικών "`r+` - read and write mode", "`w+`-read and write mode" και "`a+`-read and append mode". Τέλος, θα πραγματοποιηθεί αντιπαραβολή και αντιστοίχιση των παραπάνω εντολών που αφορούν σε ASCII/ANSI αρχεία κειμένου με τις αντίστοιχες εντολές και λειτουργίες για επεξεργασία δυαδικών αρχείων κειμένου ("`rb`"-read mode binary, "`ab`"-append mode binary, "`wb`"-write mode binary, "`rb+`-read and write mode binary", "`wb+`-read and write mode binary" και "`ab+`-read and append mode binary").

2.1.Δ.10 | ΓΡΑΦΙΚΑ ΣΤΗΣ 11

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους δημιουργίας γραφικών σχημάτων και αναπαραστάσεων δισδιάστατων σχημάτων, τα οποία θα υποστηρίζουν και εφέ κίνησης, ώστε να μπορούν να δημιουργήσουν απλές εφαρμογές κονσόλας που θα υποστηρίζουν χρώμα και κίνηση εκτός από λειτουργικότητα. Στο πλαίσιο αυτό, θα πραγματοποιηθεί αναφορά στη βιβλιοθήκη Borland Graphics Interface (BGI), η οποία είναι ενσωματωμένη στα αρχεία βιβλιοθήκης `graphics.h` και `winbgim.h`, που περιλαμβάνουν τα πρωτότυπα των συναρτήσεων γραφικών που θα χρησιμοποιηθούν. Τα αρχεία αυτά δεν περιλαμβάνονται ήδη σε κάποιο IDE και θα πρέπει να γίνει λήψη τους και αποθήκευσή τους στον αντίστοιχο κατάλογο που υπάρχουν και τα υπόλοιπα αρχεία βιβλιοθήκης συστήματος, μαζί με το αντίστοιχο αρχείο `libbgi.a`. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν εντολές, όπως οι `circle`, `line`, `lineto`, `moveto`, `ellipse`, `arc`, `rectangle`, `bar`, `drawpoly`, `pieslice`, `sector`, `outtext`, `outtextxy`, `fillellipse`, `setlinestyle`, `setcolor`, `setfillstyle`, `floodfill`, `fillpoly`, `bar3d`, `getimage`, `putimage` κ.λπ. Οι παραπάνω εντολές θα υπάρχουν ανάμεσα σε εντολές `initgraph`-`closegraph`, ενώ θα μπορεί να χρησιμοποιηθεί και η εντολή `cleardevice` για ανανέωση των γραφικών κονσόλας, διαγράφοντας ό,τι έχει εμφανιστεί. Με τις παραπάνω λειτουργίες, οι εκπαιδευόμενοι/-ες θα σχεδιάζουν παιχνίδια, όπως π.χ. μια 3x3 τρίλιζα με χρώματα, ενώ θα είναι σε θέση να σχεδιάσουν και εφαρμογές 2D, όπως η επίλυση 3 πλευρών ενός 3x3 κύβου ρούμπικ κ.ά.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Χατζηγιαννάκης, Ν. Μ. (2017). *Η γλώσσα C σε βάθος*. Αθήνα: Εκδόσεις Κλειδάριθμος.
2. Hanly, J. R. & Koffman, E. B. (2021). *Αρχές και τεχνικές προγραμματισμού με τη γλώσσα C*. Αθήνα: Εκδόσεις Κριτική.

Συμπληρωματικές

1. Μαστοροκώστας, Π. (2015). *Διαδικαστικός Προγραμματισμός: η Γλώσσα C*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις, ΣΕΑΒ.
2. *C tutorial*. (n.d.). Retrieved from: <https://www.w3schools.com/c/index.php>

2.1.Ε • ΕΦΑΡΜΟΓΕΣ ΓΡΑΦΕΙΟΥ ΚΑΙ ΨΗΦΙΑΚΕΣ ΔΕΞΙΟΤΗΤΕΣ ΣΤΗ ΣΥΓΧΡΟΝΗ ΑΓΟΡΑ ΕΡΓΑΣΙΑΣ

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Εφαρμογές γραφείου και ψηφιακές δεξιότητες στη σύγχρονη αγορά εργασίας» εισάγει τους/τις εκπαιδευόμενους/-ες στο περιβάλλον των εφαρμογών γραφείου, οι οποίες θεωρούνται απαραίτητες από την αγορά εργασίας ως κεκτημένη γνώση και πρακτική εμπειρία. Ο/Η εκπαιδευόμενος/-η θα έρθει σε επαφή με εφαρμογές επεξεργασίας κειμένου, επεξεργασίας λογιστικών φύλλων, δημιουργίας παρουσιάσεων, δημιουργίας βάσεων δεδομένων και των πλέον σύγχρονων τεχνολογιών χρήσης και λειτουργίας Διαδικτύου: ηλεκτρονική αλληλογραφία, εξοικείωση με δικτυακές συσκευές και κανόνες διευθυνσιοδότησης/ονοματοδοσίας στο Διαδίκτυο. Μέσα από τα παραπάνω, ο/η εκπαιδευόμενος/-η θα αποκτήσει τις βασικές γνώσεις και δεξιότητες για τη διαχείριση καθημερινών προκλήσεων, σχετικών με τη διαδικτυακή επικοινωνία και την επεξεργασία ευρέως διαδεδομένων τύπων εφαρμογών που κάθε περιβάλλον εργασίας απαιτεί.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Εργάζονται με αρχεία εφαρμογών γραφείου, αποθηκεύοντάς τα σε διάφορες μορφές αρχείων, τοπικά ή διαδικτυακά, σε υπηρεσίες νεφούπολογιστικής,
- Αναγνωρίζουν τις διάφορες διαθέσιμες εφαρμογές γραφείου, τη συμβατότητα χρήσης τους και πότε να τις χρησιμοποιούν,
- Επιλέγουν διάφορες διατάξεις και πρότυπα σχεδίασης αρχείων εφαρμογών γραφείου,

- Μορφοποιούν το κείμενο και τους πίνακες ενός εγγράφου,
- Αναγνωρίζουν την καλή πρακτική χρήσης εφαρμογών γραφείου με συνέπεια,
- Επεξεργάζονται γραφήματα για να επικοινωνούν τα δεδομένα κατάλληλα και με τη σωστή τους σημασία,
- Επεξεργάζονται εικόνες και αντικείμενα σχεδίασης,
- Εφαρμόζουν διορθώσεις στο περιεχόμενο ενός αρχείου πριν από την τελική εκτύπωσή του,
- Αναγνωρίζουν πιθανούς λόγους βλάβης της δικτυακής σύνδεσης,
- Επιλέγουν τις κατάλληλες εφαρμογές δοσοληψίας ηλεκτρονικού ταχυδρομείου,
- Προσδιορίζουν τη λειτουργία του Διαδικτύου και των ηλεκτρονικών μηνυμάτων, δίνοντας έμφαση στη διαδικτυακή ασφάλεια.

Βασικές έννοιες και λέξεις-κλειδιά

Αυτοματοποίηση εργασιών γραφείου / Λογισμικό υποστήριξης / Σύνταξη κειμένων / Διαχείριση δεδομένων / Παρουσίαση περιεχομένου / Διαδίκτυο / Επικοινωνία.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (4), Σύνολο (4).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΠΕΞΕΡΓΑΣΤΗΣ ΚΕΙΜΕΝΟΥ
2	ΛΟΓΙΣΤΙΚΑ ΦΥΛΛΑ
3	ΔΙΑΧΕΙΡΙΣΗ ΠΑΡΟΥΣΙΑΣΕΩΝ
4	ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ
5	ΕΠΙΚΟΙΝΩΝΙΑ ΜΕΣΩ ΔΙΑΔΙΚΤΥΟΥ
6	ΣΥΓΧΡΟΝΕΣ ΥΠΗΡΕΣΙΕΣ ΔΙΑΔΙΚΤΥΟΥ

Κείμενο αναφοράς

2.1.Ε.1 | ΕΠΕΞΕΡΓΑΣΤΗΣ ΚΕΙΜΕΝΟΥ

Σκοπός της υποενότητας είναι να εξοικειωθούν οι εκπαιδευόμενοι/-ες με τη χρήση των πιο γνωστών εφαρμογών δημιουργίας και επεξεργασίας αρχείων κειμένου. Θα αναλυθούν διεξοδικά τα χαρακτηριστικά είτε της εφαρμογής Word του εμπορικού πακέτου MS Office (desktop ή online έκδοση) είτε αντίστοιχης εφαρμογής ανοικτού κώδικα, όπως είναι η Writer του OpenOffice ή του LibreOffice. Ο/Η καταρτιζόμενος/-η γνωρίζει τους τρόπους έναρξης της εφαρμογής και εξοικειώνεται με το περιβάλλον διεπαφής, με τις επιλογές στις γραμμές εργαλείων (ή καρτέλες), τα μενού της εφαρμογής και τις ρυθμίσεις τους. Παρουσιάζονται βασικές ρυθμίσεις της εφαρμο-

γής, όπως η προεπιλεγμένη μορφή και θέση αποθήκευσης του εγγράφου και η ρύθμιση του αυτόματου χρόνου αποθήκευσής του. Διδάσκεται η δημιουργία εγγράφου, η εισαγωγή κειμένου στο έγγραφο, η αποθήκευση του εγγράφου ως αρχείου, η αποθήκευση ενός αντιγράφου του εγγράφου, το κλείσιμο του εγγράφου και το άνοιγμά του από τη θέση αποθήκευσης. Παρουσιάζονται οι τρόποι προβολής του εγγράφου, η διαχείριση κειμένου με το ποντίκι και το πληκτρολόγιο, η επιλογή κειμένου, η αντιγραφή, η αποκοπή, η αντικατάσταση και η διαγραφή του. Αναλύονται η μορφοποίηση χαρακτήρων και παραγράφων, ο ορισμός γραμματοσειράς, η μορφοποίηση σελίδας, η οργάνωση του κειμένου με κουκκίδες και αρίθμηση, η χρήση των εσοχών και ο χάρακας του εγγράφου, οι στηλοθέτες, η εύρεση και αντικατάσταση κειμένου, ο ορθογραφικός έλεγχος. Αναφέρονται η διαχείριση της κεφαλίδας και του υποσέλιδου, οι εκτυπώσεις, τα πρότυπα εγγράφων, η εισαγωγή και επεξεργασία αντικειμένων, όπως εικόνων, αυτόματων σχημάτων και γραφημάτων. Επιπρόσθετα, αναλύονται η δημιουργία και διαχείριση πινάκων πληροφοριών, η συγχώνευση αλληλογραφίας, η διαχείριση πίνακα περιεχομένων, η δημιουργία ετικετών και φακέλων αλληλογραφίας.

2.1.Ε.2 | ΛΟΓΙΣΤΙΚΑ ΦΥΛΛΑ

Στην παρούσα υποενότητα οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με την οργάνωση και επεξεργασία αριθμητικών δεδομένων και δεδομένων κειμένου σε λογιστικά φύλλα. Θα γίνει χρήση είτε της εφαρμογής Excel του εμπορικού πακέτου MS Office (desktop ή online έκδοση) είτε αντίστοιχης εφαρμογής ανοικτού κώδικα, όπως είναι η Calc του OpenOffice ή του LibreOffice. Καταρχήν, ο/η καταρτιζόμενος/-η ενημερώνεται για το περιβάλλον εργασίας της εφαρμογής, για τις βασικές επιλογές στις γραμμές εργαλείων (ή καρτέλες), τα μενού της εφαρμογής και τις σχετικές ρυθμίσεις. Στη συνέχεια, θα αναλυθεί η δομή ενός λογιστικού φύλλου, ώστε ο/η καταρτιζόμενος/-η να εξοικειωθεί με τις έννοιες: φύλλο εργασίας (sheet), βιβλίο (book), κελί (cell), γραμμή (row), στήλη (column). Παρουσιάζεται η καταχώρηση δεδομένων, είτε αφορά εισαγωγή αριθμητικών δεδομένων είτε κείμενο, ημερομηνίες κ.ά. Ιδιαίτερη έμφαση δίνεται στον διαχωρισμό «σχετικής» και «απόλυτης» αναφοράς ενός κελιού. Αναλύεται η διαχείριση κελιών, η αυτόματη συμπλήρωση κελιών, η συμπλήρωση δεδομένων με χρήση λαβής συμπλήρωσης και η διαχείριση γραμμών, στηλών και φύλλων εργασίας (αντιγραφή, αποκοπή, μετονομασία, μετακίνηση, διαγραφή). Παρουσιάζεται η μορφοποίηση γραμμών και στηλών, η απόκρυψή τους, η σταθεροποίηση τμημάτων του φύλλου εργασίας, η εκτύπωση και η διαμόρφωσή της. Περιγράφονται η αναζήτηση και αντικατάσταση περιεχομένου, ο έλεγχος τιμών και η επικύρωση δεδομένων, η μορφοποίηση κελιών, η μορφοποίηση υπό όρους και η αυτόματη μορφοποίηση. Διδάσκεται ο υπολογισμός δεδομένων με μαθηματικούς τύπους και συναρτήσεις, καθώς και η δημιουργία γραφημάτων. Επιπλέον, θα διδαχθεί η ταξινόμηση λιστών πληροφοριών, η δημιουργία πινάκων, τα φίλτρα και η διαχείριση διπλοτύπων, όπως και η εισαγωγή εξωτερικών δεδομένων.

2.1.E.3 | ΔΙΑΧΕΙΡΙΣΗ ΠΑΡΟΥΣΙΑΣΕΩΝ

Στην υποενότητα αυτή, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη χρήση προγραμμάτων δημιουργίας παρουσιάσεων (presentation) περιεχομένου και τις βασικές αρχές που πρέπει να τηρούνται, κατά τη διαδικασία μιας παρουσίασης σε ακροατήριο. Οι πιο γνωστές εφαρμογές που θα χρησιμοποιηθούν είναι η εφαρμογή PowerPoint του εμπορικού πακέτου MS Office ή η εφαρμογή ανοικτού κώδικα Impress του OpenOffice ή του LibreOffice. Περιγράφεται το περιβάλλον εργασίας της εφαρμογής, η λειτουργία των γραμμών εργαλείων (ή καρτελών), τα μενού και οι πιο χρήσιμες γενικές ρυθμίσεις της εφαρμογής. Ορίζεται το κύριο δομικό στοιχείο μιας παρουσίασης που είναι η διαφάνεια (slide), η διαχείρισή της και περιγράφεται η χρήση των πλαισίων που αποτελούν μια διαφάνεια, στα οποία εισάγεται το περιεχόμενο της παρουσίασης. Αναφέρονται τα κυριότερα είδη περιεχομένου που μπορεί να περιλαμβάνει μια παρουσίαση (κείμενο, εικόνες, γραφικά, βίντεο κ.ά.), εξηγούνται οι επιλογές διάταξης της διαφάνειας (slide layout) για την αποτελεσματικότερη οργάνωση του περιεχομένου και ορίζεται η εισαγωγή αντικειμένων. Οι εκπαιδευόμενοι/-ες διδάσκονται τη μορφοποίηση κειμένου, την επεξεργασία των διαφανειών, τις σημειώσεις για το ακροατήριο και τα σχόλια, και την εισαγωγή κειμένου, αρίθμησης ή ημερομηνίας στο υποσέλιδο διαφανειών. Αναλύεται η τροποποίηση του φόντου διαφάνειας, τα πρότυπα σχεδίασης και η χρήση των υποδειγμάτων διαφανειών. Παρουσιάζεται η διαχείριση των κυριότερων αντικειμένων που εισάγονται σε μια διαφάνεια, δηλαδή οι εικόνες, οι πίνακες, τα γραφικά smart art, τα οργανογράμματα, τα βίντεο, τα σχήματα, τα γραφήματα κ.ά. Ιδιαίτερη σημασία δίνεται στην εφαρμογή εφέ κίνησης και ήχου, είτε στα αντικείμενα είτε στις διαφάνειες, ενώ αναλύονται οι τρόποι προβολής της παρουσίασης και ο χρονοπρογραμματισμός προβολής.

2.1.E.4 | ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με τις βασικές έννοιες των βάσεων δεδομένων και θα διδαχθούν τη λειτουργία ενός συστήματος διαχείρισης βάσης δεδομένων, ώστε να είναι σε θέση να οργανώνουν, να επεξεργάζονται δεδομένα και να ανακτούν χρήσιμη πληροφορία. Θα γίνει χρήση της εφαρμογής Access είτε του εμπορικού πακέτου MS Office είτε αντίστοιχου προγράμματος ανοικτού κώδικα, όπως είναι η Base του Open Office ή του LibreOffice. Θα διδαχθεί τι είναι μια βάση δεδομένων, θα αναλυθεί η χρησιμότητα των συστημάτων διαχείρισης βάσεων δεδομένων (ΣΔΒΔ), τα βασικά εργαλεία που παρέχουν για τη διαχείριση μιας βάσης και τα επίπεδα παρουσίασης μιας βάσης. Θα εξηγηθεί τι είναι ένα μοντέλο βάσης δεδομένων και θα δοθεί έμφαση στην επεξήγηση του σχεσιακού μοντέλου. Αναλύεται το σχεσιακό σύστημα διαχείρισης βάσης δεδομένων και θα παρατεθούν τα πιο γνωστά συστήματα που υλοποιούνται στην πράξη. Οι εκπαιδευόμενοι/-ες θα γνωρίσουν την έννοια του πίνακα (table), το πεδίο (field), τους τύπους πεδίων, την εγγραφή (record/row), και θα εξοικειωθούν με την καταχώρηση όλων των τύπων δεδομένων σε μια βάση. Αναλύεται

η ταξινόμηση εγγραφών σε έναν πίνακα, η αναζήτηση και η εφαρμογή φίλτρων. Παρουσιάζονται οι συσχετίσεις (relations) πινάκων και τα είδη τους: ένα-προς-ένα (one-to-one), ένα-προς-πολλά (one-to-many), πολλά-προς-πολλά (many-to-many), όπως και το πρωτεύον (primary) και το ξένο κλειδί (foreign key). Περιγράφεται η έννοια της κανονικοποίησης μιας βάσης δεδομένων, η δημιουργία ερωτημάτων ανάκτησης, τροποποίησης, εισαγωγής ή διαγραφής δεδομένων σε μια βάση. Παρατίθενται παραδείγματα από δραστηριότητες της κοινωνικής ζωής, όπου απαιτείται η χρήση βάσεων δεδομένων για τη διαχείριση μεγάλου όγκου δεδομένων.

2.1.E.5 | ΕΠΙΚΟΙΝΩΝΙΑ ΜΕΣΩ ΔΙΑΔΙΚΤΥΟΥ

Στην παρούσα υποενότητα, ο/η καταρτιζόμενος/-η θα έρθει σε επαφή με βασικές υπηρεσίες του Διαδικτύου (Internet) και την πλοήγησή του/της σε αυτό. Διδάσκεται το τι είναι η ιστοσελίδα και η ηλεκτρονική διεύθυνσή της (URL), τι είναι το λογισμικό περιήγησης του Διαδικτύου ή φυλλομετρητής (browser). Θα αναφερθούν οι πιο γνωστοί φυλλομετρητές, όπως είναι οι Mozilla Firefox, Google Chrome, Opera, MS Edge, Safari κ.ά. Αναλύονται βασικές ρυθμίσεις των φυλλομετρητών, ο ορισμός αρχικής σελίδας, η εμφάνιση ή απόκρυψη των μενού και της γραμμής εργαλείων κ.ά. Αναλύονται βασικά κουμπιά, όπως το κουμπί αρχικής σελίδας, ανανέωσης τρέχουσας σελίδας, μετάβασης μια σελίδα πίσω, μετάβασης μια σελίδα μπροστά. Θα εξηγηθεί η πρόσβαση στις αγαπημένες ιστοσελίδες και η διαχείρισή τους, το ιστορικό ιστοσελίδων, τα μενού, η διαχείριση γραμμών εργαλείων. Βαρύτητα δίνεται στις μηχανές αναζήτησης, στη λήψη (download) δεδομένων από μια ιστοσελίδα, στην αποθήκευση (save as) μιας ιστοσελίδας τοπικά, στην εκτύπωση πληροφοριών. Αναλύεται η ηλεκτρονική αλληλογραφία (e-mail), ενώ θα αναφερθούν οι πιο γνωστές εφαρμογές διαχείρισης ηλεκτρονικής αλληλογραφίας. Αναφέρονται η δημιουργία και αποστολή μηνυμάτων, η απάντηση, η προώθηση, η αποστολή ομαδικών μηνυμάτων, η κοινοποίηση και η ιδιαιτέρη κοινοποίηση. Αναφορά γίνεται στους φακέλους μηνυμάτων και στα εισερχόμενα, απεσταλμένα, ανεπιθύμητα, πρόχειρα και διαγραμμένα μηνύματα. Θα τονιστεί η διαχείριση επαφών, οι ομάδες επαφών, η διαχείριση ημερολογίου. Αναλύονται θέματα ασφάλειας προσωπικών δεδομένων, και οι έννοιες ιός (virus), αντιικό (antivirus), τείχος προστασίας (firewall), ενώ μπορεί να δοθεί βαρύτητα στο περιβάλλον του Gmail. Αναφέρονται οι δυνατότητες εφαρμογών για σύγχρονη επικοινωνία και οι τηλεσυναντήσεις μεταξύ χρηστών/-τριών με μετάδοση φωνής/εικόνας. Τέλος, αναλύονται πιθανά προβλήματα κατά τη σύνδεση στο Διαδίκτυο σε ένα τυπικό γραφείο.

2.1.E.6 | ΣΥΓΧΡΟΝΕΣ ΥΠΗΡΕΣΙΕΣ ΔΙΑΔΙΚΤΥΟΥ

Στην υποενότητα «Σύγχρονες υπηρεσίες Διαδικτύου» γίνεται αναφορά σε διάφορες τεχνολογίες και υπηρεσίες που μπορούν να βοηθήσουν τους/τις εκπαιδευόμενους/-ες στη διαμόρφωση του βιογραφικού και γενικά του διαμοιρασμού δεδομένων και αρχείων. Αρχικά, προτείνεται να γίνει χρήση διαδικτυακών υπηρεσιών για διαμοιρασμό

πληροφοριών και συνεργασία. Παρουσιάζονται υπηρεσίες όπως το Google Drive, το Dropbox και το wetransfer.com. Τα έγγραφα Google, τα υπολογιστικά φύλλα, οι παρουσιάσεις, η δημιουργία φορμών και η συνεργατική δημιουργία εγγράφων διδάσκονται αναλυτικά. Ο τρόπος διαμοιρασμού εγγράφων και λοιπών αρχείων σε επιλεγμένους χρήστες/-τριες αναλύεται στην υποενότητα. Στη συνέχεια, γίνεται αναφορά σε αποθετήρια εικόνων, όπως είναι το unsplash.com, το freepik.com και άλλα παρόμοια αποθετήρια, για χρήση εικόνων σε διάφορα έργα. Η δωρεάν χρήση έτοιμων templates (προτύπων) για βιογραφικά, έγγραφα, και άλλες εργασίες και ο τρόπος εντοπισμού τους στο Διαδίκτυο είναι σημαντικό να αναφερθούν στην υποενότητα. Οι προηγμένες υπηρεσίες αναζήτησης του Google είναι σημαντικές για την αποτελεσματική εύρεση πληροφοριών. Υπηρεσίες όπως ο μελετητής Google και τα βιβλία της Google θα βοηθήσουν τους/τις εκπαιδευόμενους/-ες στην εύρεση έγκυρων πληροφοριών και για τις μετέπειτα ενότητες που θα διδαχθούν. Η παρουσίαση εκπαιδευτικών υπηρεσιών με δωρεάν μαθήματα και υλικό βοηθά στον εντοπισμό διδακτικού υλικού και στην απόκτηση γνώσεων και δεξιοτήτων σε διάφορους τομείς του επαγγέλματός τους. Οι διαδικτυακές υπηρεσίες του Gov.gr (π.χ. υπεύθυνη δήλωση, ψηφιακή υπογραφή εγγράφου, αναζήτηση εγγράφων) και οι υπηρεσίες τραπεζών προτείνεται να παρουσιαστούν και, όταν είναι εφικτό, να χρησιμοποιηθούν στο πλαίσιο της εκπαίδευσης, ώστε να εξοικειωθούν οι εκπαιδευόμενοι/-ες με τις αυτές τις υπηρεσίες.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Ξαρχάκος, Κ. & Καρολίδης, Δ. (2016). *Μαθαίνετε εύκολα Microsoft Office 2016*. Αθήνα: Εκδόσεις Άβακας.
2. Συλλογικό Έργο. (2021). *7+1 Ελληνικά Microsoft Windows 10-Office 2019/365*. Αθήνα: Εκδόσεις Κλειδάριθμος.

Συμπληρωματικές

1. Δρόσος, Δ., Βουγιούκας, Δ., Καλλίγερος, Ε., Κοκολάκης, Σ. & Σκιάνης, Χ. (2015). *Εισαγωγή στην επιστήμη των υπολογιστών & επικοινωνιών*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/4582>
2. Εθνική Ακαδημία Ψηφιακών Ικανοτήτων: <https://nationaldigitalacademy.gov.gr/mathimata>

2.2 • ΕΞΑΜΗΝΟ Β΄

2.2.A • ΓΛΩΣΣΑ CLIENT-SIDE ΔΙΑΔΙΚΤΥΑΚΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ (JAVASCRIPT)

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Γλώσσα client-side διαδικτυακού προγραμματισμού (JavaScript)» εισάγει τους/τις εκπαιδευόμενους/-ες στη γλώσσα διαδικτυακού προγραμματισμού JavaScript, ώστε να αποκτήσουν τις βάσεις για τη δημιουργία δυναμικού περιεχομένου, κατάλληλου για χρήση στον Παγκόσμιο Ιστό. Η σύνταξη της γλώσσας αυτής είναι επηρεασμένη από τη γλώσσα προγραμματισμού C11, η οποία διδάσκεται στη μαθησιακή ενότητα «Γλώσσα προγραμματισμού I (C11)-Αλγοριθμική και δομές δεδομένων» του εξαμήνου Α΄. Επίσης, θα χρησιμοποιηθεί η προηγούμενη γνώση της μαθησιακής ενότητας «Εργαλεία ανάπτυξης εφαρμογών Διαδικτύου» του εξαμήνου Α΄. Ειδικότερα, παρουσιάζεται το συντακτικό της γλώσσας JavaScript. Αναφέρονται βασικές έννοιες του διαδικτυακού προγραμματισμού, όπως client-side και server-side, User Experience (UX) και User Interface (UI). Παρουσιάζονται έννοιες, όπως: έξοδος, σχόλια, δηλώσεις, τύποι μεταβλητών, τελεστές, εκχώρηση δεδομένων, τύποι δεδομένων, συναρτήσεις, αντικείμενα, γεγονότα, συμβολοσειρές και μέθοδοι συμβολοσειρών, μέθοδοι και ιδιότητες αριθμών, πίνακες, βρόγχοι πινάκων, δομές επιλογής, δομές επανάληψης, κλάσεις, μετατροπή τύπων, εμβέλεια, λογικά και συντακτικά σφάλματα, εκσφαλμάτωση και απόδοση. Επίσης, αναφέρονται προχωρημένα θέματα διαδικτυακού προγραμματισμού, όπως ασύγχρονη JavaScript (AJAX), JQUERY, JSON, form validation, δημοφιλή JS frameworks, καθώς και θέματα ασφάλειας διαδικτυακών εφαρμογών (OWASP top ten) και τρόποι προστασίας ενάντια σε απειλές. Τέλος, σχεδιάζονται και αναπτύσσονται δυναμικοί, ανταποκρίσιμοι (responsive) και ασφαλείς (secure) ιστότοποι, που υποστηρίζουν οι σύγχρονοι φυλλομετρητές.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναγνωρίζουν τα πλεονεκτήματα των client-side γλωσσών προγραμματισμού,
- Επιλέγουν τα κατάλληλα στοιχεία (μεταβλητές, συναρτήσεις κ.ά.) που θα χρησιμοποιήσουν στους ιστότοπους που θα αναπτύξουν,
- Αναγνωρίζουν τη σημασία του δομημένου και τεκμηριωμένου κώδικα,
- Αναπτύσσουν ιστότοπους με χρήση της γλώσσας προγραμματισμού JavaScript,
- Συνεισφέρουν με πρωτότυπο κώδικα στην κοινότητα των προγραμματιστών ιστοτόπων,
- Τροποποιούν υπάρχοντα κώδικα, ανάλογα με τις προδιαγραφές των ιστοτόπων που αναπτύσσουν,

- Αποφασίζουν για τη χρήση ή μη κατάλληλων λογισμικών, σχετικών με την ανάπτυξη ιστοτόπων,
- Συνεργάζονται με άλλους/-ες εκπαιδευόμενους/-ες, συγκροτώντας ομάδες, με σκοπό τη δημιουργία και την επίλυση προβλημάτων των ιστοτόπων,
- Υιοθετούν θετική στάση ως προς τις νέες τεχνολογίες ανάπτυξης ιστοτόπων.

Βασικές λέξεις και έννοιες-κλειδιά

AJAX / Client side / Constructor / JQuery / JSON / Properties / Αντικείμενο /
Δομές επανάληψης / Δομές επιλογής / Μέθοδοι/συναρτήσεις / Πίνακες.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (4), Σύνολο (4).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΗΝ CLIENT SIDE ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ JAVASCRIPT
2	ΔΟΜΙΚΑ ΣΤΟΙΧΕΙΑ ΤΗΣ JAVASCRIPT
3	ΔΟΜΕΣ ΕΠΙΛΟΓΗΣ/ΕΠΑΝΑΛΗΨΗΣ
4	ΠΙΝΑΚΕΣ
5	ΣΥΝΑΡΤΗΣΕΙΣ
6	ΑΝΤΙΚΕΙΜΕΝΑ
7	ΑΠΟΣΦΑΛΜΑΤΩΣΗ (DEBUGGING)-ΕΠΙΚΥΡΩΣΗ ΔΕΔΟΜΕΝΩΝ (DATA VALIDATION)-ΘΕΜΑΤΑ ΑΣΦΑΛΕΙΑΣ
8	ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ ΔΙΑΔΙΚΤΥΑΚΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

Κείμενο αναφοράς

2.2.A.1 | ΕΙΣΑΓΩΓΗ ΣΤΗΝ CLIENT SIDE ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ JAVASCRIPT

Η εκπαιδευτική υποενότητα αποσκοπεί στην εισαγωγή και κατανόηση των βασικών εννοιών, που σχετίζονται με τις client side γλώσσες προγραμματισμού, δηλαδή αυτές που χρησιμοποιούνται από τον φυλλομετρητή (browser) του/της χρήστη/-τριας. Αρχικά, παρουσιάζονται τα πλεονεκτήματα που έχει μια γλώσσα client side. Στο πλαίσιο της υποενότητας, οι εκπαιδευόμενοι/-ες θα γνωρίσουν τη γλώσσα JavaScript και θα μελετήσουν τις ομοιότητες και τις διαφορές που παρουσιάζει με άλλες γλώσσες προγραμματισμού. Επίσης, θα αναλυθεί η σημασία της στη δημιουργία και στον σχεδιασμό ενός User Interface, αλλά και στη βελτίωση της εμπειρίας του χρήστη (User Experience). Μέσω της υποενότητας, αναφέρονται τα βασικά στοιχεία της γλώσσας, δίνοντας έμφαση στις ιδιαιτερότητές της και στον τρόπο με τον οποίο τρέχει στον ενσωματωμένο διερμηνευτή (interpreter) του browser. Επίσης, κρίνεται σκόπιμη η

αναφορά στην έννοια και στη χρήση του parser, καθώς και στην ανάλυση του όρου DOM (Document Object Model) και της αλληλεπίδρασης της Javascript με το εν λόγω αντικείμενο. Στη συνέχεια, αναλύεται η έννοια του στοιχείου script, η χρήση του στην ενσωμάτωση κώδικα σε μια σελίδα και ο τρόπος με τον οποίο τρέχει στον browser. Είναι ιδιαίτερα ωφέλιμοι οι εκπαιδευόμενοι/-ες να επικεντρωθούν στην ενσωμάτωση scripts με χρήση εξωτερικών αρχείων, αφού αυτός είναι ο τρόπος που χρησιμοποιείται στην πράξη, όταν εργαζόμαστε με σελίδες HTML. Επίσης, αναλύονται τα χαρακτηριστικά async και defer, και η χρήση τους στην ενσωμάτωση των scripts. Τέλος, μέσω της υποενότητας, οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με τη χρήση απλών scripts και, με τη χρήση πρακτικών παραδειγμάτων, θα προβλέπουν το αποτέλεσμα που θα εμφανίζεται κάθε φορά στην οθόνη.

2.2.A.2 | ΔΟΜΙΚΑ ΣΤΟΙΧΕΙΑ ΤΗΣ JAVASCRIPT

Η εκπαιδευτική υποενότητα «Δομικά στοιχεία της JavaScript» ασχολείται με τις σταθερές, τις μεταβλητές και τους τύπους δεδομένων που χρησιμοποιούνται στην JavaScript. Αρχικά, οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με τους πρωτογενείς τύπους δεδομένων και με τους κανόνες ονοματοδοσίας σε μια τιμή. Συγκεκριμένα, αναλύονται οι τύποι δεδομένων αριθμοί (number), οι μεγάλοι ακέραιοι (bigint), οι συμβολοσειρές (string), οι λογικές μεταβλητές (boolean), τα σύμβολα (symbol) και οι ειδικοί τύποι null και undefined (συμβολίζουν την ανυπαρξία τιμής). Για κάθε έναν από τους παραπάνω τύπους δεδομένων, περιγράφεται ο τρόπος ορισμού του και παρουσιάζεται ο τρόπος εκτέλεσής του. Έπειτα, αναλύονται οι διαφορές μεταξύ σταθερών και μεταβλητών, με τη χρήση πρακτικών παραδειγμάτων δηλώνονται ονόματα σταθερών και μεταβλητών, και μελετάται η συμπεριφορά τους. Στη συνέχεια, περιγράφεται η δυνατότητα μετατροπής μιας υφιστάμενης τιμής κάποιου τύπου δεδομένων στην αντίστοιχη τιμή ενός άλλου τύπου δεδομένων. Συγκεκριμένα, αναλύονται οι καθολικές μέθοδοι μετατροπής Number, String, Boolean και BigInt. Επιπλέον, μέσω της υποενότητας αυτής, οι εκπαιδευόμενοι/-ες εξοικειώνονται με την έννοια του αντικειμένου window (γνωστού και ως global object) και μελετούν τα μέλη που περιέχει, δίνοντας ιδιαίτερη σημασία στα μέλη document, console, history, location, navigation και screen. Στο πλαίσιο της υποενότητας, αναλύεται η διάκριση των τελεστών σε: αριθμητικούς, λογικούς, σχεσιακούς, καταχώρησης και bitwise. Επίσης, μέσω πρακτικών παραδειγμάτων, αναλύεται η χρήση τους και επισημαίνεται η προτεραιότητα που έχουν όταν κατασκευάζονται σύνθετες παραστάσεις.

2.2.A.3 | ΔΟΜΕΣ ΕΠΙΛΟΓΗΣ/ΕΠΑΝΑΛΗΨΗΣ

Αρχικά, η εν λόγω εκπαιδευτική υποενότητα εξετάζει τη χρήση της δομής ελέγχου if και των παραλλαγών αυτής (if...else και if...else if...else) στην εκτέλεση εντολών, ανάλογα με το αν ισχύει μια συνθήκη. Οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με τη χρήση του τριαδικού τελεστή (?), με τη βοήθεια του οποίου μπορεί να γραφτεί ισοδύναμος κώδικας,

χωρίς να χρησιμοποιηθεί η `if..else`. Επίσης, στο πλαίσιο δημιουργίας σύνθετων εφαρμογών σε γλώσσα JavaScript, κρίνεται σκόπιμη η χρήση, περιγραφή και σύνταξη του τελεστή `??` (γνωστού ως «πρώτα ορισμένο-first defined»). Στο πλαίσιο της ανάλυσης των δομών επανάληψης, περιγράφεται το συντακτικό της δομής `switch`, η οποία χρησιμοποιείται όταν χρειάζεται να γίνουν διαφορετικές ενέργειες, ανάλογα με την τιμή που έχει μια έκφραση. Επισημαίνεται ότι η χρήση της συγκεκριμένης δομής συνιστάται όταν υπάρχουν περισσότερες από δύο επιλογές και δίνεται έμφαση στη σημασία της εντολής `break` κατά τη διάρκεια εκτέλεσης ενός προγράμματος. Επιπλέον, μέσω της υποενότητας, οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με την έννοια του μπλοκ κώδικα και θα εξετάσουν την εμβέλεια (καθολική ή τοπική) που έχει μια μεταβλητή, ανάλογα με το πού έχει οριστεί μέσα στον κώδικα του προγράμματος. Στη συνέχεια, εξετάζονται οι δομές επανάληψης `for`, `while` και `do..while`, οι οποίες επιτρέπουν την επανάληψη του ίδιου κώδικα στην JavaScript. Συγκεκριμένα, αναλύεται η χρήση και το συντακτικό των παραπάνω δομών, επισημαίνοντας τις διαφορές μεταξύ τους. Μέσα από πρακτικά παραδείγματα, οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με τη χρήση των δομών επανάληψης και θα διδαχθούν να επιλέγουν και να εφαρμόζουν την κατάλληλη δομή σε κάθε περίπτωση.

2.2.A.4 | ΠΙΝΑΚΕΣ

Στη συγκεκριμένη εκπαιδευτική υποενότητα, οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με τους πίνακες. Αρχικά, θα κατανοήσουν ότι οι πίνακες αποτελούν ένα βασικό εργαλείο για την ομαδοποίηση ομοειδών στοιχείων ή αντικείμενων διαφορετικών τύπων. Μέσω της υποενότητας, εξετάζονται οι μονοδιάστατοι και διδιάστατοι πίνακες, με ιδιαίτερη έμφαση να δίνεται στους πρώτους. Με τη χρήση πρακτικών παραδειγμάτων, οι εκπαιδευόμενοι/-ες θα διδαχθούν να δηλώνουν και να αρχικοποιούν έναν πίνακα. Στη συνέχεια, θα εξοικειωθούν με βασικές ιδιότητες του πίνακα και με βασικούς τελεστές, όπως είναι οι `delete` και `spread`. Επιπλέον, η εν λόγω εκπαιδευτική υποενότητα προσφέρει τις γνώσεις και τις δεξιότητες που απαιτούνται για την κατασκευή ενός πίνακα, χρησιμοποιώντας εναλλακτικά τον κατασκευαστή `Array([n])` ή τις στατικές μεθόδους `Array.of([n])` και `from`. Επίσης, οι εκπαιδευόμενοι/-ες θα είναι σε θέση να χρησιμοποιήσουν τις δομές επανάληψης `for..of` και `for..in` σε έναν πίνακα. Στο πλαίσιο της υποενότητας, θα εξεταστούν οι μέθοδοι που προσφέρει η JavaScript, για να εκτελεστούν ενέργειες σε έναν πίνακα. Συγκεκριμένα, θα παρουσιαστούν οι μέθοδοι που σχετίζονται με την αναζήτηση ενός στοιχείου στον πίνακα, η ταξινόμηση των στοιχείων του πίνακα σε αύξουσα ή φθίνουσα σειρά και η εισαγωγή/διαγραφή στοιχείων από την αρχή ή το τέλος ενός πίνακα. Επίσης, κρίνεται σκόπιμη η εξήγηση της μεθόδου `slice` που χρησιμοποιείται για την κατασκευή υποπινάκων, της μεθόδου `splice` η οποία αφαιρεί στοιχεία από έναν πίνακα και τα επιστρέφει σε έναν νέο πίνακα, της μεθόδου `fill` για να συμπληρώσει έναν πίνακα με τιμές, της μεθόδου `copyWithin` η οποία αντιγράφει μέρος του πίνακα στον εαυτό του, και της μεθόδου `toString` για τη μετατροπή ενός πίνακα σε μια μορφή συμβολοσειράς.

2.2.A.5 | ΣΥΝΑΡΤΗΣΕΙΣ

Η συγκεκριμένη εκπαιδευτική υποενότητα αποτελεί ένα σημαντικό μέρος της μαθησιακής ενότητας. Στόχος είναι η κατανόηση και η χρήση των συναρτήσεων σε ένα πρόγραμμα JavaScript. Αρχικά, οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με τον τρόπο που γίνεται ο ορισμός και η κλήση μιας συνάρτησης. Μέσα από πρακτικά παραδείγματα, θα αναλυθούν οι παράμετροι που μπορεί να έχει μια συνάρτηση, καθώς επίσης και η έννοια της επιστρεφόμενης τιμής. Επίσης, κρίνεται σκόπιμο να επισημανθεί ότι στην JavaScript η διοχέτευση των ορισμάτων σε μια συνάρτηση μπορεί να γίνει μόνο *by value*. Το γεγονός αυτό έρχεται σε αντίθεση με άλλες γλώσσες προγραμματισμού, οι οποίες μπορούν να διοχετεύουν αυτούσια μια μεταβλητή ως όρισμα, χρησιμοποιώντας τους τρόπους *by reference* και *by pointer*. Στο πλαίσιο της υποενότητας, παρουσιάζονται οι συναρτησιακές εκφράσεις (*function expressions*) και, με τη χρήση πρακτικών παραδειγμάτων, αναλύονται οι διαφορές ορισμού μιας συνάρτησης από μια συναρτησιακή έκφραση. Επιπλέον, δίνεται έμφαση στη χρήση μιας συναρτησιακής έκφρασης ως όρισμα σε μια συνάρτηση (γνωστή και ως *callback*). Η συγκεκριμένη εργασία θεωρείται κύριο χαρακτηριστικό του τρόπου λειτουργίας της JavaScript. Ως εκ τούτου, κρίνεται σκόπιμη η εκτέλεση αρκετών παραδειγμάτων, προκειμένου οι εκπαιδευόμενοι/-ες να εξοικειωθούν με το συγκεκριμένο τρόπο ορισμού συναρτήσεων. Τέλος, η εκπαιδευτική υποενότητα εξετάζει τον τρόπο ορισμού μιας συνάρτησης. Ο τρόπος αυτός υλοποιείται με τη βοήθεια των συναρτήσεων *arrow functions* (ή *lambda functions*). Οι εκπαιδευόμενοι/-ες κατανοούν ότι ο συγκεκριμένος τρόπος αποτελεί συντακτική σύντμηση της συναρτησιακής έκφρασης με ίδια συμπεριφορά, και είναι σε θέση να προσαρμόζουν κώδικα, ώστε να χρησιμοποιεί *arrow function* αντί για συναρτησιακές εκφράσεις.

2.2.A.6 | ΑΝΤΙΚΕΙΜΕΝΑ

Στη συγκεκριμένη εκπαιδευτική υποενότητα, οι εκπαιδευόμενοι/-ες θα κατανοήσουν την έννοια του αντικειμένου (*object*), την αντιστοιχία του με τις γλώσσες αντικειμενοστρεφούς προγραμματισμού και το γεγονός ότι πάνω σε αυτά «χτίζονται» τα προγράμματα της JavaScript. Στη συνέχεια, θα ασχοληθούν με τον τρόπο σύνταξης για τον ορισμό ενός αντικειμένου και θα εξοικειωθούν με την έννοια της ιδιότητας (*property*). Συγκεκριμένα, θα διδαχθούν πώς να διαβάζουν, να δημιουργούν, να ενημερώνουν και να διαγράφουν τα *properties* των αντικειμένων. Στόχος της υποενότητας είναι η κατανόηση και η χρήση των συναρτήσεων ως μελών των αντικειμένων (γνωστών ως μεθόδων). Επίσης, οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με τη λέξη-κλειδί *this*, μέσω της οποίας πραγματοποιείται η πρόσβαση στα μέλη του αντικειμένου. Επιπλέον, θα εξετάσουν τελεστές που έχουν οριστεί να λειτουργούν μέσα σε αντικείμενα, όπως ο τελεστής *in*. Μέσα από πρακτικά παραδείγματα, θα εξετάσουν την ανάθεση μέσω αποδόμησης (*destructuring*) σε μεταβλητές και σε σύνθετα αντικείμενα. Ακόμη, θα αναλυθούν συνηθισμένες ενέργειες αντικειμένων, όπως η επανάληψη επί αντικειμέ-

νου (με χρήση της `for...in`) και η αντιγραφή ενός αντικειμένου μέσα σε ένα άλλο. Στο πλαίσιο της υποενότητας, οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με την έννοια του κατασκευαστή (`constructor`) αντικειμένων. Συγκεκριμένα, αναλύεται ο τρόπος με τον οποίο χρησιμοποιείται ένας `constructor` για τη δημιουργία αντικειμένων ίδιου τύπου και παρουσιάζονται οι ενσωματωμένοι `constructors` που περιέχει η `JavaScript`. Τέλος, εξοικειώνονται με την έννοια και τη χρήση του πρωτοτύπου (`prototype`). Επικεντρώνονται στην έννοια της κληρονομικότητας και αναλύεται η σημασία του `Object.prototype`, καθώς και η χρήση της ιδιότητας `prototype` για την προσθήκη νέων ιδιοτήτων (`properties`) και μεθόδων (`methods`) σε έναν `constructor`.

2.2.A.7 | ΑΠΟΣΦΑΛΜΑΤΩΣΗ (DEBUGGING)-ΕΠΙΚΥΡΩΣΗ ΔΕΔΟΜΕΝΩΝ (DATA VALIDATION)-ΘΕΜΑΤΑ ΑΣΦΑΛΕΙΑΣ

Η εκπαιδευτική υποενότητα εστιάζεται στον ρόλο των φυλλομετρητών (`web browsers`) για τον εντοπισμό των λογικών και συντακτικών σφαλμάτων. Για κάθε έναν από τους πιο δημοφιλείς φυλλομετρητές (`Chrome`, `Firefox`, `Edge`, `Opera`, `Safari`), αναλύεται η ενεργοποίηση και η χρήση της δυνατότητας ελέγχου του κώδικα του προγράμματος. Οι εκπαιδευόμενοι/-ες εξοικειώνονται με τη χρήση του ενσωματωμένου εκσφαλματωτή (`debugger`) και με τη βοήθειά του ορίζουν σημεία διακοπής (`breakpoints`) για την εξέταση των μεταβλητών κατά την εκτέλεση του κώδικα. Επιπλέον, αναλύεται η σημασία που έχουν τα σχόλια (`comments`) στην αποτροπή εκτέλεσης του κώδικα. Επίσης, εξοικειώνονται με τη χρήση της λέξης κλειδί `debugger`, αλλά και τη μέθοδο `console.log()` για την εμφάνιση τιμών στο παράθυρο του `debugger`. Στο πλαίσιο της υποενότητας, οι εκπαιδευόμενοι/-ες εξοικειώνονται με τη διαδικασία της επικύρωσης δεδομένων (`data validation`), προκειμένου να διασφαλίσουν ότι πραγματοποιείται σωστή εισαγωγή δεδομένων από τον/την χρήστη/-τρια. Μέσα από πρακτικά παραδείγματα, αναλύονται συνηθισμένα λάθη μη έγκυρης καταχώρησης από τον/τη χρήστη/-τρια (π.χ. μη έγκυρη ημερομηνία, χρήση κειμένου σε αριθμητικό πεδίο) και διαχωρίζεται η επικύρωση από την πλευρά του διακομιστή (`server side`) και από την πλευρά του/της πελάτη/-ισσας (`client side validation`). Στο σημείο αυτό, κρίνεται σκόπιμη η ανάλυση της μεθόδου `checkValidity()` και των πιο συνηθισμένων ιδιοτήτων (`validity properties`). Τέλος, μέσω της υποενότητας αυτής, οι εκπαιδευόμενοι/-ες θα εξετάσουν, μέσω του προτύπου `OWASP Top 10`, τα σημαντικότερα θέματα ασφάλειας διαδικτυακών εφαρμογών και τους τρόπους προστασίας ενάντια σε απειλές. Με αυτόν τον τρόπο, θα αποκτήσουν τις απαραίτητες γνώσεις και τις δεξιότητες για να σχεδιάσουν και να αναπτύξουν δυναμικούς, ανταποκρίσιμους και ασφαλείς ιστότοπους που υποστηρίζουν οι φυλλομετρητές.

2.2.A.8 | ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ ΔΙΑΔΙΚΤΥΑΚΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

Η εκπαιδευτική υποενότητα εξετάζει τη σειριοποίηση/αποσειριοποίηση ενός αντικειμένου, δηλαδή τη μετατροπή ενός αντικειμένου σε συμβολοσειρά και το αντί-

στροφο, χρησιμοποιώντας το πρωτόκολλο σύνταξης δεδομένων JSON (JavaScript Object Notation). Αναλύεται η σημασία και ο κομβικός ρόλος που διαδραματίζει το JSON σε μια εφαρμογή, επισημαίνοντας ότι ο πιο συνηθισμένος τρόπος επικοινωνίας client-server είναι η μετατροπή των δεδομένων του αντικειμένου σε συμβολοσειρά και η αποστολή αυτής στον server. Επίσης, σκοπός της υποενότητας είναι η παρουσίαση της AJAX (Asynchronous JavaScript and XML) ως σύνολο τεχνικών για τη δημιουργία ασύγχρονων διαδικτυακών εφαρμογών. Αναλύεται ο τρόπος λειτουργίας, μέσω του οποίου είναι δυνατή η ενημέρωση τμημάτων μιας ιστοσελίδας χωρίς να χρειαστεί να γίνει επαναφόρτωση (reload) αυτής. Περιγράφονται τα πλεονεκτήματά της και αναλύονται οι τεχνολογίες που χρησιμοποιεί, επισημαίνοντας ανάμεσά τους τη χρήση JSON για την ανταλλαγή των δεδομένων. Επιπλέον, δίνεται έμφαση στη χρήση του αντικειμένου XMLHttpRequest για την ασύγχρονη επικοινωνία, παρουσιάζοντας τις συνηθέστερες μεθόδους –π.χ. `new XMLHttpRequest()`, `abort()`, `getResponseHeader()`, `open()`– και properties αυτού, π.χ. `onload`, `readyState`, `responseText`, `responseXML`, `status`. Επιπλέον, μέσω της υποενότητας, αναλύεται η JQuery, μια δημοφιλής βιβλιοθήκη της JavaScript, η οποία επιτρέπει στους/στις προγραμματιστές/-ίστριες να χρησιμοποιούν την AJAX με εύκολο τρόπο. Συγκεκριμένα, περιγράφονται τα χαρακτηριστικά από τα οποία αποτελείται η εν λόγω βιβλιοθήκη, και μέσα από πρακτικά παραδείγματα οι εκπαιδευόμενοι/-ες εξοικειώνονται με το συντακτικό και τη χρήση του στον χειρισμό τμημάτων κώδικα HTML.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Λιακέας, Γ. (2021). *Η γλώσσα JavaScript*. Αθήνα: Εκδόσεις Κλειδάριθμος.
2. Robbins, J. N. (2018). *Learning web design: A beginner's guide to HTML, CSS, JavaScript, and web graphics*. California: O'Reilly.

Συμπληρωματικές

1. Duckett, J. (2014). *JavaScript and jQuery: Interactive front-end web development*. New Jersey: Wiley & Sons.
2. JavaScript tutorial. (n.d.). Retrieved from: <https://www.w3schools.com/js/default.asp>

2.2.B • ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Συστήματα διαχείρισης βάσεων δεδομένων» εισάγει τους/τις εκπαιδευόμενους/-ες στη γλώσσα ερωταπαντήσεων (SQL) και στα συ-

στήματα διαχείρισης βάσεων δεδομένων, ώστε να αποκτήσουν τις βάσεις για τη δημιουργία και διαχείριση βάσεων δεδομένων που θα χρησιμοποιηθούν σε δι-αδικτυακές εφαρμογές. Ειδικότερα, παρουσιάζονται θεωρητικές έννοιες, όπως πλεονεκτήματα βάσεων δεδομένων, αρχιτεκτονική, δομές, οντότητες, ιδιότη-τες-χαρακτηριστικά, σχέσεις, πρωτεύον κλειδί, ξένο κλειδί, κανονικοποίηση, μο-ντέλο οντοτήτων-συσχετίσεων, άλλα μοντέλα, σχεσιακή άλγεβρα, σχεσιακός λογισμός και δημοφιλή συστήματα διαχείρισης βάσεων δεδομένων. Επίσης, πα-ρουσιάζονται βασικές εντολές του δημοφιλούς συστήματος διαχείρισης βάσεων δεδομένων MariaDB, όπως CREATE, SELECT, DROP, INSERT, UPDATE, DELETE, TRUNCATE, WHERE, LIKE, ORDER BY, GROUP BY, LIMIT, DISTINCT, ALTER, FROM, FUNCTIONS (MAX, MIN, SUM, COUNT, AVG), regular expressions, INDEXES, JOINS, UNION και INTERSECT. Επιπροσθέτως, αναφέρονται προχωρημένα θέμα-τα βάσεων δεδομένων, όπως storage engines, transactions, triggers, procedures, conditions, sequences, performance, άλλα συστήματα διαχείρισης βάσεων δε-δομένων (MongoDB, PL/SQL), αλλά και θέματα ασφάλειας, όπως SQL injection protection και σωστή διαχείριση χρηστών (user privileges). Επίσης, σχεδιάζονται και αναπτύσσονται σχεσιακές βάσεις δεδομένων και ενσωματώνονται σε δυναμι-κούς ιστότοπους. Τέλος, γίνεται αναφορά στις τεχνολογίες αποθήκευσης δεδομέ-νων RAID (με έμφαση στις τεχνολογίες 0, 1, 10, 2, 3, 4, 5) και στις εμφωλευμένες RAID τεχνολογίες.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναγνωρίζουν τους διαφορετικούς τύπους βάσεων δεδομένων,
- Αναγνωρίζουν τη χρησιμότητα των βάσεων δεδομένων,
- Διαχειρίζονται βάσεις δεδομένων,
- Εφαρμόζουν κατάλληλα ερωτήματα για την ανάκτηση ή/και τροποποίηση ή/και δι-αγραφή δεδομένων,
- Συσχετίζουν πίνακες βάσεων δεδομένων,
- Δημιουργούν κατάλληλες βάσεις δεδομένων με σκοπό τη χρήση τους σε ιστότοπους,
- Χρησιμοποιούν διαφορετικές μηχανές βάσεων δεδομένων, κατάλληλων για χρήση σε κάθε εξατομικευμένη διαδικτυακή εφαρμογή,
- Συνεργάζονται με τα υπόλοιπα μέλη της ομάδας στον σχεδιασμό και στην υλοποίη-ση κατάλληλων βάσεων δεδομένων,
- Διακρίνουν τις τεχνολογίες αποθήκευσης δεδομένων RAID, καθώς και τις εμφωλευ-μένες RAID τεχνολογίες,
- Υιοθετούν θετική στάση ως προς τις νέες τεχνολογίες ανάπτυξης ιστοτόπων.

Βασικές έννοιες και λέξεις-κλειδιά

Δεδομένα και Πληροφορία / Βάση δεδομένων / Σύστημα διαχείρισης βάσεων δεδομένων / Διάγραμμα οντοτήτων-Συσχετίσεων (Entity-Relationship Diagram) / SQL / Query (ερώτημα) / Ευρετήριο (index) / Συστοιχία RAID.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (4), Σύνολο (4).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΙΣ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ
2	ΑΝΑΠΑΡΑΣΤΑΣΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΜΟΝΤΕΛΟ ΟΝΤΟΤΗΤΩΝ ΣΥΣΧΕΤΙΣΕΩΝ
3	ΣΧΕΣΙΑΚΕΣ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΣΧΕΣΙΑΚΗ ΑΛΓΕΒΡΑ
4	ΕΙΣΑΓΩΓΗ ΣΤΗΝ SQL
5	ΑΝΑΠΤΥΞΗ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ΣΤΟΝ MARIADB SERVER
6	ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ SQL
7	ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

Κείμενο αναφοράς

2.2.B.1 | ΕΙΣΑΓΩΓΗ ΣΤΙΣ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Στην υποενότητα αυτή, παρουσιάζονται οι έννοιες των δεδομένων και της πληροφορίας και γίνεται αναφορά στα ακατέργαστα δεδομένα (raw data) και στην εισαγωγή στην επεξεργασία δεδομένων. Επίσης, παρουσιάζεται η έννοια του τύπου δεδομένων (data type) και δίνονται παραδείγματα. Στη συνέχεια, γίνεται αναφορά στους σύνθετους τύπους δεδομένων, στις δομές δεδομένων (data structures) και στις εγγραφές (records), και δίνεται ο ορισμός της έννοιας της βάσης δεδομένων και των πλεονεκτημάτων χρήσης της έναντι των εναλλακτικών τρόπων αποθήκευσης δεδομένων. Τονίζεται η διασφάλιση της ακεραιότητας (integrity), του μη πλεονασμού (non-redundancy) και της συνέπειας (consistency) των δεδομένων. Τέλος, αναφέρονται παραδείγματα εφαρμογών βάσεων δεδομένων σε διάφορους τομείς της οικονομίας. Η υποενότητα περιλαμβάνει επίσης τον ορισμό ενός συστήματος διαχείρισης βάσης δεδομένων (ΣΔΒΔ). Παρουσιάζονται, ακόμη, δημοφιλή ΣΔΒΔ: MS Access, MS SQL, MariaDB, MySQL, Oracle PL/SQL και MongoDB, στα πλεονεκτήματα και στα μειονεκτήματα των οποίων γίνεται αναφορά. Τέλος, παρουσιάζεται η αρχιτεκτονική ενός συστήματος βάσης δεδομένων, η οποία αποτελείται από τρία επίπεδα: το φυσικό επίπεδο (physical layer), το οποίο καθορίζει πώς τα δεδομένα θα αποθηκεύονται στο φυσικό μέσο αποθήκευσης, το εννοιολογικό επίπεδο (conceptual level),

το οποίο καθορίζει το σχήμα μοντελοποίησης των δεδομένων, και το επίπεδο όψης (view level), το οποίο καθορίζει τον τρόπο που τα δεδομένα θα προβάλλονται στον/στη χρήση/-τρια.

2.2.B.2 | ΑΝΑΠΑΡΑΣΤΑΣΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΜΟΝΤΕΛΟ ΟΝΤΟΤΗΤΩΝ ΣΥΣΧΕΤΙΣΕΩΝ

Στην υποενότητα αυτή, αναλύεται η έννοια της μοντελοποίησης δεδομένων, η οποία αφορά τον τρόπο αναπαράστασης των δεδομένων, και παρουσιάζονται τα κυριότερα μοντέλα αναπαράστασης δεδομένων. Συγκεκριμένα, παρουσιάζονται: το επίπεδο μοντέλο (flat model), το ιεραρχικό μοντέλο (hierarchical model), το μοντέλο δικτύου (network model), το σχεσιακό μοντέλο (relational model) και το αντικειμενοστρεφές μοντέλο (object oriented model). Στο πλαίσιο του μαθήματος, θα χρησιμοποιηθεί το μοντέλο οντοτήτων-συσχετίσεων (Entity relationship model, ER), το οποίο ανήκει στα σχεσιακά μοντέλα. Παρουσιάζονται, στη συνέχεια, τα βασικά δομικά στοιχεία του μοντέλου ER, και συγκεκριμένα οι οντότητες (entities), γίνεται η διάκρισή τους σε ασθενείς (weak) και δυνατές (strong), και παρουσιάζονται τα χαρακτηριστικά (attributes) τους, τα οποία μπορεί να είναι απλά (simple) ή σύνθετα (composite), και οι σχέσεις (relationships) μεταξύ των οντοτήτων, οι οποίες έχουν περιορισμούς (constraints). Παρουσιάζεται, επίσης, ο τρόπος απεικόνισης των στοιχείων αυτών στο διάγραμμα ER και ένα παράδειγμα ER διαγράμματος. Τονίζεται ότι τιμές μπορούν να παίρνουν μόνο τα χαρακτηριστικά (attributes) και οι τιμές του κάθε χαρακτηριστικού προσδιορίζονται από το πεδίο τιμών του. Τονίζεται, επίσης, ότι χαρακτηριστικά έχουν οι οντότητες και οι σχέσεις. Στη συνέχεια, γίνεται αναφορά στις εγγραφές (records) μιας οντότητας, στην ανάγκη διάκρισης μεταξύ των εγγραφών, και παρουσιάζεται η έννοια του πρωτεύοντος κλειδιού (primary key) μιας οντότητας. Παρουσιάζονται οι σχέσεις μεταξύ των οντοτήτων, η πολλαπλότητα (cardinality) μιας συσχέτισης και οι τύποι της (N:N), καθώς επίσης και οι σχέσεις ειδίκευσης (specialization), οι οποίες ορίζουν τη συσχέτιση μιας κατηγορίας με τις υποκατηγορίες της. Τέλος, παρουσιάζεται η μεθοδολογία ανάπτυξης ενός μοντέλου οντοτήτων συσχετίσεων, μέσα από παραδείγματα.

2.2.B.3 | ΣΧΕΣΙΑΚΕΣ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΣΧΕΣΙΑΚΗ ΑΛΓΕΒΡΑ

Τα σχεσιακά ΣΔΒΔ αποθηκεύουν τα δεδομένα σε μορφή δισδιάστατων πινάκων. Γίνεται παρουσίαση της δομής του πίνακα για την περίπτωση οντοτήτων και για την περίπτωση σχέσεων, και γίνεται αναφορά στις έννοιες του ξένου κλειδιού (foreign key), του σύνθετου κλειδιού (composite) και στη σημασία τους για την αναπαράσταση σχέσεων. Δίνονται επίσης παραδείγματα πινάκων οντοτήτων και σχέσεων. Το δεύτερο θέμα που πραγματεύεται η υποενότητα είναι η κανονικοποίηση. Γίνεται αναφορά στα προβλήματα που μπορεί να παρουσιάζει μια βάση δεδομένων, π.χ. άσκοπη επανάληψη δεδομένων και ανωμαλίες τροποποίησης (εισαγωγής, διαγραφής ή/και τροποποίησης). Παρουσιάζεται η έννοια της κανονικοποίησης και της σημασίας της για τη βελτιστοποίηση των σχεσιακών βάσεων δεδομένων. Στη συνέχεια, παρουσιάζ-

ζονται οι βαθμοί κανονικοποίησης (1NF, 2NF, 3NF) και οι τεχνικές κανονικοποίησης με διάσπαση και με σύνθεση. Η διαδικασία της κανονικοποίησης προτείνεται να παρουσιαστεί μέσα από ένα πραγματικό παράδειγμα που χρειάζεται κανονικοποίηση και για τους τρεις βαθμούς. Το τελευταίο θέμα της υποενότητας είναι η σχεσιακή άλγεβρα, η οποία παρέχει τελεστές για την εκτέλεση πράξεων μεταξύ των πινάκων. Τονίζεται εδώ ότι κάθε σχεσιακή πράξη που εκτελείται πάνω σε πίνακα, παράγει έναν νέο πίνακα. Γίνεται παρουσίαση των βασικών πράξεων της σχεσιακής άλγεβρας: πράξεις που αναφέρονται σε πίνακες (π.χ. ένωση, τομή, σύζευξη) και πράξεις που αφορούν εγγραφές (π.χ. επιλογή, προβολή). Τέλος, δίνονται παραδείγματα εκτέλεσης βασικών σχεσιακών πράξεων σε πίνακες.

2.2.B.4 | ΕΙΣΑΓΩΓΗ ΣΤΗΝ SQL

Γίνεται παρουσίαση της SQL (structured query language), η οποία είναι μια γλώσσα διαχείρισης σχεσιακών βάσεων δεδομένων, κοινή για όλα τα ΣΔΒΔ, βασισμένη στη σχεσιακή άλγεβρα. Χρησιμοποιείται τόσο για τη διαχείριση πινάκων όσο και για τη διαχείριση ερωτημάτων (queries). Αναλυτικότερα, παρουσιάζονται οι τύποι δεδομένων που παίρνουν τα χαρακτηριστικά (attributes) των πινάκων, οι τιμές NULL, NOT NULL και οι τελεστές της SQL (μαθηματικοί, λογικοί, αλφαριθμητικοί). Γίνεται αναφορά στην ιεραρχία των τελεστών, ποια πράξη δηλαδή εκτελείται πρώτη σε μια ακολουθία πράξεων, και, στη συνέχεια, παρουσιάζονται οι βασικές εντολές και εκφράσεις της SQL και η σύνταξή τους. Συγκεκριμένα, παρουσιάζονται εντολές διαχείρισης πινάκων (CREATE, DROP, ALTER), διαχείρισης δεδομένων (INSERT, DELETE, UPDATE, SET, TRUNCATE), διαχείρισης ερωτήσεων (SELECT, FROM, WHERE, DISTINCT), διαχείρισης πράξεων μεταξύ αποτελεσμάτων ερωτημάτων (UNION, INTERSECT, JOIN), ταξινόμησης εγγραφών (ORDER BY, DESC, ASC, LIMIT), ομαδοποίησης εγγραφών (GROUP BY, HAVING) και συναρτήσεις επεξεργασίας συνόλου δεδομένων (MAX, MIN, SUM, COUNT, AVG). Ιδιαίτερη βαρύτητα δίνεται στην τεχνική ορισμού πρωτευόντων και ξένων κλειδιών, με χρήση της εντολής ALTER TABLE και χρήση του κατάλληλου περιορισμού. Παρουσιάζονται επίσης οι εκφράσεις PRIMARY KEY, FOREIGN KEY και REFERENCE. Η υποενότητα κλείνει με τις κανονικές εκφράσεις (regular expressions), οι οποίες είναι σύμβολα ή συντομογραφίες που συντάσσονται μαζί με συμβολοσειρές και υποδηλώνουν ένα σύνολο συμβολοσειρών. Παρουσιάζονται αναλυτικά τα σύμβολα των κανονικών εκφράσεων και η σύνταξή τους σε ένα ερώτημα.

2.2.B.5 | ΑΝΑΠΤΥΞΗ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ΣΤΟΝ MARIA DB SERVER

Η υποενότητα αυτή περιλαμβάνει την παρουσίαση του MariaDB Server, των λόγων που επιλέχθηκε στο πλαίσιο του μαθήματος και των διαφορών του από τον MySQL Server. Περιλαμβάνει επίσης τα βήματα εγκατάστασης του MariaDB Server. Η εγκατάσταση του MariaDB Server περιλαμβάνει και την εγκατάσταση του HeidiSQL, ενός editor που επιτρέπει τη διαχείριση βάσεων δεδομένων, μέσω ενός γραφικού περιβάλλ-

λοντος. Εναλλακτικά, η διαχείριση των βάσεων δεδομένων γίνεται από το command window (cmd). Παρουσιάζεται, στη συνέχεια, ο HeidiSQL και το πώς συνδεόμαστε στον MariaDB Server μέσω του HeidiSQL, καθώς επίσης και το cmd (command line) του MariaDB Server και το πώς συνδεόμαστε στον MariaDB Server μέσω του cmd. Στη συνέχεια, παρουσιάζονται τα βήματα δημιουργίας μιας βάσης δεδομένων στον MariaDB Server και τα βήματα δημιουργίας πινάκων και εισαγωγής δεδομένων στους πίνακες. Σημειώνεται ότι γίνεται παρουσίαση της εκτέλεσης των παραπάνω ενεργειών μέσω κώδικα SQL στο command window, καθώς και μέσω του γραφικού περιβάλλοντος του HeidiSQL. Ακολουθούν τα βήματα δημιουργίας ερωτημάτων (queries), τόσο μέσω κώδικα SQL στο command window όσο και μέσω του γραφικού περιβάλλοντος του HeidiSQL. Προτείνεται η δημιουργία της βάσης, των πινάκων και των ερωτημάτων να γίνει μέσα από παραδείγματα. Παρουσιάζονται, τέλος, τα βήματα ενσωμάτωσης μιας βάσης δεδομένων σε έναν δυναμικό ιστότοπο. Ο MariaDB προσφέρει ένα σύνολο διακριτών connectors, οι οποίοι έχουν τη δυνατότητα να ενσωματώσουν τη βάση δεδομένων που αναπτύχθηκε σε MariaDB, στον κώδικα υλοποίησης ενός ιστοτόπου που αναπτύχθηκε με διάφορες γλώσσες προγραμματισμού (π.χ. Python, JAVA, PHP κ.ά.). Γίνεται παρουσίαση των αντίστοιχων connectors και της διασύνδεσης μεταξύ MariaDB και του αντίστοιχου προγραμματιστικού περιβάλλοντος.

2.2.B.6 | ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ SQL

Το πρώτο θέμα είναι οι συνδιαλλαγές (transactions), οι οποίες είναι ένα σύνολο εντολών SQL, οι οποίες είτε επικυρώνονται όλες μαζί στο τέλος (commit) είτε σε κάποιο στάδιο αναιρούνται όλες μαζί (rollback). Παρουσιάζονται παραδείγματα κώδικα SQL για την υλοποίηση συνδιαλλαγών και τονίζεται η σημασία τους. Το επόμενο θέμα είναι το έναυσμα (trigger), ένας μηχανισμός που πυροδοτεί ενέργειες όταν συμβαίνουν γεγονότα (όπως INSERT, UPDATE, DELETE) σε έναν πίνακα. Παρουσιάζονται παραδείγματα κώδικα SQL για την υλοποίηση triggers και τονίζεται η σημασία τους, ακολουθούν οι περιορισμοί (constraints) και παρουσιάζονται οι εντολές και οι τεχνικές δημιουργίας περιορισμών για την επιβολή τους στα δεδομένα. Συγκεκριμένα, γίνεται αναφορά στη δημιουργία περιορισμού, στους περιορισμούς πρωτεύοντος και ξένου κλειδιού, στους περιορισμούς μοναδικότητας και στους περιορισμούς ελέγχου. Το επόμενο θέμα είναι η αυτόματη ανανέωση δεδομένων, οι τεχνικές για αυτόματη ανανέωση δεδομένων και τονίζεται ότι χρησιμοποιούνται κυρίως όταν αλλάζει η τιμή των ξένων κλειδιών. Παρουσιάζεται, στη συνέχεια, η έννοια της διαδικασίας (procedure), η οποία είναι μια ρουτίνα που αποθηκεύεται και μπορεί να κληθεί. Ακόμη, παρουσιάζονται: οι εντολές για την υλοποίηση διαδικασιών και η έννοια της ακολουθίας (sequence) και η χρήση της στη δημιουργία πρωτευόντων κλειδιών (primary keys), η έννοια του τεχνητού κλειδιού (surrogate key) και οι εντολές για την υλοποίηση ακολουθιών. Το τελευταίο θέμα είναι τα ευρετήρια (indices). Παρουσιάζεται η έννοια των ευρετηρίων, τα οποία είναι ειδικές δομές δεδομένων που επιταχύνουν τη

διαδικασία της αναζήτησης πληροφοριών από τη βάση δεδομένων. Παρουσιάζεται η χρησιμότητά τους, ιδιαίτερα σε βάσεις με πολλές εγγραφές, και δίνονται παραδείγματα δημιουργίας ευρετηρίων.

2.2.B.7 | ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

Στην υποενότητα αυτή, παρουσιάζεται η έννοια της απόδοσης (performance) της βάσης δεδομένων και οι παράμετροι μέτρησής της, όπως: ταχύτητα, διαθεσιμότητα, ασφάλεια, αξιοπιστία, κλιμάκωση. Στη συνέχεια, γίνεται αναφορά στη βελτιστοποίηση της απόδοσης, η οποία περιλαμβάνει: τη δημιουργία ευρετηρίων (indices), την κανονικοποίηση του σχήματος της βάσης, τη χρήση εργαλείων βελτίωσης απόδοσης που παρέχει το ΣΔΒΔ (παρουσιάζονται εργαλεία του MariaDB Server) και την παρακολούθηση της απόδοσης (monitoring) – παρουσιάζονται και εργαλεία του MariaDB Server (π.χ. MariaDB Monitor). Στη συνέχεια, παρουσιάζεται η έννοια της μηχανής αποθήκευσης (storage engine), η οποία είναι ένα λογισμικό για την αποθήκευση και διαχείριση δεδομένων. Γίνεται αναφορά στα βασικότερα storage engines (InnoDB, SQLite, MyISAM) και εξηγούνται τα κριτήρια επιλογής ενός storage engine. Παρουσιάζεται επίσης η έννοια της διαχείρισης χρηστών/-τριών, η σπουδαιότητά της και οι βασικές της λειτουργίες, όπως: η διαχείριση χρηστών/-τριών (προσθήκη, αφαίρεση), η διαχείριση ρόλων, η παραχώρηση δικαιωμάτων (privileges) και ο έλεγχος ταυτοποίησης και πρόσβασης. Ακολουθούν οι SQL injections, οι κακόβουλες επιθέσεις σε εφαρμογές ή ιστοσελίδες που χρησιμοποιούν βάσεις δεδομένων. Οι επιθέσεις έχουν τη μορφή έγχυσης (injection) SQL κώδικα στον κώδικα της εφαρμογής, με στόχο την εκτέλεση κακόβουλων ενεργειών στα δεδομένα της βάσης. Παρουσιάζονται, επίσης, βασικές μέθοδοι προστασίας, οι οποίες περιλαμβάνουν (ενδεικτικά): δημιουργία παραμετροποιημένων ερωτημάτων, επικύρωση εισαγωγής δεδομένων, καθώς και περιορισμό των δικαιωμάτων των χρηστών/-τριών. Τέλος, παρουσιάζονται οι συστοιχίες RAID (Redundant Array of Independent Disks), οι οποίες χρησιμοποιούνται για την αποθήκευση δεδομένων σε πολλούς φυσικούς σκληρούς δίσκους, δημιουργώντας έναν ενιαίο εικονικό χώρο αποθήκευσης. Παρουσιάζονται τα πλεονεκτήματα των συστοιχιών RAID και παρουσιάζονται οι RAID 0, 1, 10, 2, 3, 4, 5, καθώς και οι εμφωλευμένες RAID τεχνολογίες.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Κεχρής, Ε. (2021). *Σχεσιακές Βάσεις Δεδομένων*. Αθήνα: Εκδόσεις Κριτική.
2. Hoffer, J., Ramesh, V. & Tori, H. (2017). *Βάσεις Δεδομένων: σύγχρονη διαχείριση*. Θεσσαλονίκη: Εκδόσεις Τζιόλα.

Συμπληρωματικές

1. Ο επίσημος ιστότοπος του MariaDB: <https://mariadb.org/documentation/>
2. Ο επίσημος ιστότοπος του HeidiSQL: <https://www.heidisql.com/>

2.2.Γ • ΕΙΣΑΓΩΓΗ ΣΤΗ ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ PYTHON

Περίληψη της μαθησιακής ενότητας

Η «Εισαγωγή στη γλώσσα προγραμματισμού Python» εισάγει τους/τις εκπαιδευόμενους/-ες στη λογική του προγραμματισμού με τη γλώσσα Python. Η αλγοριθμική λογική της γλώσσας προγραμματισμού Python έχει επιρροές από τη γλώσσα προγραμματισμού C11, η οποία διδάσκεται στη μαθησιακή ενότητα «Γλώσσα Προγραμματισμού I (C11)-Αλγοριθμική και δομές δεδομένων» του εξαμήνου Α΄ και από τη μαθησιακή ενότητα «Γλώσσα προγραμματισμού II (C++14)-Αντικειμενοστρεφής προγραμματισμός» του εξαμήνου Β΄ κατάρτισης. Συνεπώς, θα χρησιμοποιηθεί η πρότερη γνώση των δύο παραπάνω μαθησιακών ενότητων. Στην παρούσα μαθησιακή ενότητα, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με προχωρημένα θέματα διαδικαστικού προγραμματισμού, όπως είναι οι δομές επιλογής, επανάληψης, οι λίστες και οι συναρτήσεις, δημιουργώντας διαλογικές και αποκρίσιμες εφαρμογές κονσόλας. Μέσα από τα παραπάνω, ενισχύονται οι βασικές γνώσεις και δεξιότητες για τον προγραμματισμό πολύπλοκων προγραμμάτων σε Python κατά την επόμενη μαθησιακή ενότητα του επόμενου (Γ΄) ακαδημαϊκού εξαμήνου, καθώς και η δυνατότητα υλοποίησης εφαρμογών Python που θα αλληλεπιδρούν με το Διαδίκτυο.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναπτύσσουν εφαρμογές με χρήση της γλώσσας προγραμματισμού Python,
- Συνεισφέρουν με πρωτότυπο κώδικα στην κοινότητα των προγραμματιστών εφαρμογών,
- Τροποποιούν υπάρχοντα κώδικα, ανάλογα με τις προδιαγραφές των εφαρμογών που αναπτύσσουν,
- Αποφασίζουν για τη χρήση ή μη κατάλληλων βιβλιοθηκών, σχετικών με την ανάπτυξη εφαρμογών,

- Συνεργάζονται με άλλους/-ες εκπαιδευόμενους/-ες, συγκροτώντας ομάδες, με σκοπό τη δημιουργία και την επίλυση προβλημάτων των εφαρμογών,
- Υιοθετούν θετική στάση ως προς τις υπάρχουσες τεχνολογίες ανάπτυξης εφαρμογών.

Βασικές έννοιες και λέξεις-κλειδιά

Εγκατάσταση Python / Μεταβλητές και τύποι / Αλγοριθμικές δομές επιλογής και επανάληψης / Λίστες/Σύνολα / Λεξικά/Πλειάδες / Διαχείριση αρχείων κειμένου/ Δυναμικά αρχεία/Αρχεία CSV / Συναρτήσεις/Συναρτήσεις lambda / Αρθρώματα/ Πακέτα / Συμβολοσειρές (strings) / Pygame / Κανονικές εκφράσεις (Regular expressions).

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (3), Σύνολο (3).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ ΜΕ PYTHON
2	ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ
3	ΑΡΧΕΙΑ
4	ΣΥΝΑΡΤΗΣΕΙΣ ΚΑΙ ΒΙΒΛΙΟΘΗΚΕΣ
5	ΕΠΕΞΕΡΓΑΣΙΑ ΣΥΜΒΟΛΟΣΕΙΡΩΝ
6	ΠΡΟΗΓΜΕΝΑ ΘΕΜΑΤΑ ΣΤΗΝ PYTHON

Κείμενο αναφοράς

2.2.Γ.1 | ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ ΜΕ PYTHON

Η μαθησιακή υποενότητα «Εισαγωγή στον προγραμματισμό με Python» έχει σκοπό να εισαγάγει τους μαθητές στη σύνταξη και στη λογική της Python, προσφέροντας ένα σύνολο βασικών γνώσεων και δεξιοτήτων προγραμματισμού. Αρχικά, οι εκπαιδευόμενοι/-ες πρέπει να μάθουν για τη φιλοσοφία της Python και να μπορούν να εγκαταστήσουν και να εκτελέσουν προγράμματα και εντολές από τη βασική γραμμή εντολών ή τον editor του βασικού περιβάλλοντος ανάπτυξης (IDE) της Python. Θα επεξηγηθούν οι δυνατότητες της Python, που την καθιστούν μια δημοφιλή γλώσσα, και θα γίνει αναφορά σε ενδεικτικούς τομείς χρήσης της γλώσσας, για να καταλάβουν οι εκπαιδευόμενοι/-ες τη χρησιμότητά της. Στη συνέχεια, πρέπει να εξοικειωθούν με τη σύνταξη της Python, τους αριθμητικούς, συγκριτικούς και λογικούς τελεστές, τις εντολές εισόδου/εξόδου, τις μεταβλητές, τους τύπους δεδομένων, τις μετατροπές των τύπων και τις αλγοριθμικές δομές της επιλογής (if else...) και της επανάληψης for και while. Θα δοθούν κλασικές εισαγωγικές ασκήσεις, για να καταλάβουν οι εκπαι-

δευόμενοι/-ες τη διαχείριση της δομής των προγραμμάτων της Python με τη χρήση εσοχών. Έμφαση πρέπει να δοθεί στις ειδικές δυνατότητες της Python για πολλαπλή εκχώρηση, την αλυσιδωτή σύγκριση, τις απλές και σύνθετες εντολές if, μιας γραμμής κ.λπ., που διαφοροποιούν την Python από τις άλλες γλώσσες προγραμματισμού. Οι εκπαιδευόμενοι/-ες πρέπει να πειραματιστούν με τους διαφορετικούς τρόπους της επανάληψης for, κατανοώντας την ευελιξία της Python, ώστε να χρησιμοποιήσουν τις continue και break.

2.2.Γ.2 | ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Η μαθησιακή υποεπάρκεια αποτελεί ένα σημαντικό κομμάτι της εκπαίδευσης στον προγραμματισμό με τη γλώσσα Python, καθώς αφορά τον αποτελεσματικό χειρισμό δεδομένων με διάφορες δομές δεδομένων, που διαφέρουν ή δεν υπάρχουν σε γλώσσες προγραμματισμού (π.χ. στη C). Οι εκπαιδευόμενοι/-ες εξερευνούν τη δημιουργία, τη διαχείριση και την αποθήκευση δεδομένων με διάφορες δομές δεδομένων, καθώς και τον τρόπο εργασίας με αρχεία. Κατά τη διάρκεια της εκπαίδευσης, οι εκπαιδευόμενοι/-ες μαθαίνουν να δημιουργούν και να διαχειρίζονται αρχικά τις λίστες και κατανοούν τη δυναμικότητα των συγκεκριμένων δομών σε σχέση με άλλες γλώσσες προγραμματισμού. Μέσα από παραδείγματα, αντιλαμβάνονται ότι είναι σε θέση να δημιουργήσουν λίστες που αλλάζει το μέγεθός τους, λίστες με διαφορετικού τύπου δεδομένα και λίστες λιστών. Μαθαίνουν να προσπελάζουν τις λίστες, να αλλάζουν τα στοιχεία τους, να προσθέτουν ή να αφαιρούν στοιχεία, να εκτελούν επαναληπτικές διαδικασίες, να αντιγράφουν, να ταξινομούν και να ενώνουν λίστες.

Στη συνέχεια, μαθαίνουν για τα σύνολα, τα λεξικά και τις πλειάδες. Κατανοούν ποια είναι η χρησιμότητα κάθε δομής δεδομένων και εστιάζουν την προσοχή στις μεθόδους για δημιουργία, προσπέλαση, αντιγραφή, ταξινόμηση, όπου επιτρέπεται, αντιγραφή και ένωση δομών. Επιπρόσθετα, οι εκπαιδευόμενοι/-ες μαθαίνουν για τις μετατροπές μεταξύ των διαφόρων δομών δεδομένων που υποστηρίζει η Python. Είναι σημαντικό η υποεπάρκεια να διδαχθεί με ασκήσεις και προγραμματιστικά προβλήματα, που μπορούν να επιλυθούν, ατομικά και συνεργατικά, ως μικρά έργα, για πρακτική εξάσκηση σε πραγματικά προβλήματα.

2.2.Γ.3 | ΑΡΧΕΙΑ

Η συγκεκριμένη υποεπάρκεια έρχεται σε συνέχεια της προηγούμενης υποεπάρκειας, διότι αφορά τη διαχείριση δεδομένων, που όμως εδώ αποθηκεύονται στη δευτερεύουσα μνήμη. Αρχικά, οι εκπαιδευόμενοι/-ες μαθαίνουν να διαχειρίζονται αρχεία κειμένου για ανάγνωση, εγγραφή και επεξεργασία κειμένου. Τα δεδομένα μπορούν να φορτωθούν από τα αρχεία σε κατάλληλες δομές δεδομένων και να επεξεργαστούν. Δίνονται παραδείγματα και ασκήσεις, πάντως στις οποίες μπορούν να εργαστούν οι εκπαιδευόμενοι/-ες. Στη συνέχεια, μελετώνται τα δυαδικά αρχεία, μαζί με τη διαδικασία ανάγνωσης, εγγραφής και επεξεργασίας των στοιχείων τους. Στην τρέχουσα μα-

θησιακή υποενότητα, γίνεται αναφορά στη διαχείριση των εξαιρέσεων, μέσω των Try Except. Κατανοούν οι εκπαιδευόμενοι/-ες πώς χειρίζεται η Python απρόσκοπτα τα προβλήματα μη ύπαρξης αρχείων και μαθαίνουν να διαχειρίζονται πολλαπλές εξαιρέσεις. Επιπρόσθετα, οι εκπαιδευόμενοι/-ες μαθαίνουν για τον χειρισμό φακέλων μέσα από προγράμματα Python και μπορούν να διαχειρίζονται αρχεία και φακέλους.

Στην υποενότητα αυτή, είναι σημαντικό να γίνει αναφορά στα αρχεία CSV και στη χρησιμότητά τους, και να δοθούν διάφορα παραδείγματα αρχείων. Στη συνέχεια, στις μεθόδους χειρισμού αρχείων CSV, μαθαίνουν πώς να εισάγουν τη βιβλιοθήκη csv της Python και διδάσκονται τις σημαντικότερες μεθόδους της. Μέσα από επαναληπτικές δομές, επεξεργάζονται τα στοιχεία των αρχείων CSV. Στην υποενότητα αυτή, προτείνεται να δοθούν αρκετές ασκήσεις και προγραμματιστικά προβλήματα στους/στις εκπαιδευόμενους/-ες, που μπορούν να επιλυθούν και συνεργατικά, για πρακτική εξάσκηση σε πραγματικά προβλήματα.

2.2.Γ.4 | ΣΥΝΑΡΤΗΣΕΙΣ ΚΑΙ ΒΙΒΛΙΟΘΗΚΕΣ

Σε αυτή την υποενότητα, οι εκπαιδευόμενοι/-ες εξερευνούν τη δημιουργία, τη χρήση και την επαναχρησιμοποίηση συναρτήσεων, καθώς και την ενσωμάτωση διαφόρων βιβλιοθηκών και πακέτων στα προγράμματά τους. Θα μάθουν να ορίζουν συναρτήσεις, να περνούν παραμέτρους, να επιστρέφουν τιμές και να χρησιμοποιούν συναρτήσεις για την επίλυση επαναλαμβανόμενων λειτουργιών. Μαθαίνουν επίσης τις έννοιες των τοπικών και καθολικών μεταβλητών και πώς αυτές επηρεάζουν την εμβέλεια των μεταβλητών στο πρόγραμμά τους. Θα πρέπει να μάθουν τις ιδιαίτερες λειτουργίες της Python, όπως την επιστροφή δύο ή περισσότερων τιμών στις συναρτήσεις, τις ενσωματωμένες συναρτήσεις, τις συναρτήσεις lambda. Θα γίνει παρουσίαση της τεχνικής της αναδρομής, ώστε να γνωρίζουν οι εκπαιδευόμενοι/-ες και για αυτή την προγραμματιστική τεχνική. Επιπλέον, θα εξερευνήσουν τη χρήση των βιβλιοθηκών (αρθρωμάτων) και πακέτων της Python για την επέκταση των δυνατοτήτων τους. Η χρήση ενσωματωμένων βιβλιοθηκών, όπως είναι η math, πρέπει να αναλυθούν. Θα μάθουν να εγκαθιστούν και να εισάγουν βιβλιοθήκες, όπως το NumPy για τον αριθμητικό υπολογισμό και το Matplotlib για τη δημιουργία γραφικών αναπαραστάσεων δεδομένων. Θα παρουσιαστούν οι πιο γνωστές βιβλιοθήκες για διαφορετικούς σκοπούς. Στην υποενότητα αυτή, πρέπει να γίνει ανάλυση του τρόπου δόμησης των προγραμμάτων σε υποπρογράμματα, δηλαδή να διδαχθούν την έννοια του τμηματικού προγραμματισμού. Θα πρέπει να μάθουν να σχεδιάζουν προγράμματα, χρησιμοποιώντας υποπρογράμματα, ώστε να δημιουργούν πιο αποδοτικά και πολυπλοκότερα προγράμματα.

2.2.Γ.5 | ΕΠΕΞΕΡΓΑΣΙΑ ΣΥΜΒΟΛΟΣΕΙΡΩΝ

Η μαθησιακή υποενότητα «Επεξεργασία Συμβολοσειρών» αποτελεί ένα σημαντικό κομμάτι της εκμάθησης της γλώσσας προγραμματισμού Python και αφορά τον χειρισμό και τον αναλυτικό έλεγχο συμβολοσειρών (strings). Οι εκπαιδευόμενοι/-ες εξοικει-

ώνονται με τον τρόπο αποθήκευσης και αναπαράστασης συμβολοσειρών στην Python, με τους διαφορετικούς τύπους που προσφέρει, καθώς και με τις βασικές λειτουργίες επεξεργασίας κειμένου, όπως την αντικατάσταση, τη συγχώνευση και τη διαίρεση συμβολοσειρών. Επίσης, μαθαίνουν να χειρίζονται τα διάφορα είδη χαρακτήρων και κωδικοποιήσεων, καθώς και τα όρια της συμβολοσειράς. Οι ειδικοί χαρακτήρες πρέπει να παρουσιαστούν στο πλαίσιο του μαθήματος και θα πρέπει να διδαχθούν τα συστήματα κωδικοποίησης χαρακτήρων και την κωδικοποίηση UTF-8. Επιπλέον, εξετάζουν τις διάφορες συναρτήσεις που προσφέρει η Python για την επεξεργασία συμβολοσειρών, όπως τη μετατροπή σε κεφαλαία/πεζά, τη διαίρεση σε λέξεις και την αντιστροφή της σειράς χαρακτήρων. Η μορφοποίηση εξόδου μέσω της μεθόδου `format`, ώστε να βελτιωθεί η μορφή της εκτύπωσης, πρέπει να παρουσιαστεί αναλυτικά μέσα από παραδείγματα. Η υποενότητα αυτή πρέπει να περιλαμβάνει ασκήσεις, που απαιτούν από τους/τις εκπαιδευόμενους/-ες να εφαρμόσουν τις νέες τους δεξιότητες σε πραγματικά προβλήματα, όπως είναι η ανάλυση και η επεξεργασία κειμένου από αρχεία και η εξαγωγή πληροφοριών από δομημένα δεδομένα. Αυτή η υποενότητα είναι καίρια για τον προγραμματισμό εφαρμογών που απαιτούν την επεξεργασία και τον έλεγχο κειμένου, όπως των εφαρμογών αναζήτησης, ανάλυσης δεδομένων και επεξεργασίας γλωσσικού υλικού.

2.2.Γ.6 | ΠΡΟΗΓΜΕΝΑ ΘΕΜΑΤΑ ΣΤΗΝ PYTHON

Στην υποενότητα «Προηγμένα θέματα Python» προτείνεται να διδαχθούν κάποια επιπλέον θέματα, που θα βοηθήσουν τους/τις εκπαιδευόμενους/-ες να δουν αρκετές από τις δυνατότητες της Python. Αρχικά, μπορούν να αναλυθούν οι κανονικές εκφράσεις (Regular Expressions ή RegEx) στην Python, που χρησιμοποιούνται για να γίνει έλεγχος εάν μια συμβολοσειρά ή ένα σύνολο συμβολοσειρών περιέχει ένα συγκεκριμένο μοτίβο. Προτείνεται να χρησιμοποιηθούν οι συναρτήσεις που βοηθούν στην αντιστοίχιση προτύπων, όπως η `findall()`, η `search()`, η `split()` κ.λπ. Δυνατότητες όπως το `multithreading`, που χρησιμοποιείται κυρίως για την επιτάχυνση του υπολογισμού, μπορούν να αναλυθούν στο πλαίσιο του μαθήματος. Η χρήση αρχείων Excel και η εμβάθυνση στη διαχείριση των εξαιρέσεων θα αποτελέσουν επιπλέον θέματα της υποενότητας. Προτείνεται οι εκπαιδευόμενοι/-ες να αξιοποιούν βιβλιοθήκες, όπως οι `decimals` και `fractions`. Οι βασικοί μέθοδοι της γλώσσας για την ταξινόμηση και την αναζήτηση πρέπει να χρησιμοποιηθούν μέσα από κατάλληλα επιλεγμένα παραδείγματα. Η δημιουργία γραφικών διεπαφών μέσω του πακέτου `tkinter` και η επικοινωνία με βάσεις δεδομένων πρέπει να παρουσιαστούν και να δοθούν σύντομα παραδείγματα, ώστε να αποκτήσουν οι εκπαιδευόμενοι/-ες γνώσεις στα συγκεκριμένα θέματα. Η εισαγωγή στη βιβλιοθήκη `pygame` είναι σημαντικό να διδαχθεί στην υποενότητα μέσω παραδειγμάτων, ώστε να κατανοήσουν οι εκπαιδευόμενοι/-ες ποιος είναι ο τρόπος ανάπτυξης διαδραστικών εφαρμογών. Η δημιουργία κάποιων απλών παιχνιδιών με την `pygame` θα μπορούσε να γίνει συνεργατικά από τους/τις εκπαιδευόμενους/-ες.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Μαγκούτης, Κ. & Νικολάου, Χ. (2015). *Εισαγωγή στον αντικειμενοστρεφή προγραμματισμό με Python*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/1708>
2. Μανής, Γ. (2015). *Εισαγωγή στον Προγραμματισμό με αρωγό τη γλώσσα Python*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/2745>

Συμπληρωματικές

1. *Εισαγωγή στην Python*, Mathesis (χ.χ.). Ηράκλειο: Πανεπιστημιακές Εκδόσεις Κρήτης, ΙΤΕ: <https://mathesis.cup.gr/courses/course-v1:ComputerScience+CS1.1+20C/course/>
2. *Προχωρημένος προγραμματισμός με Python*, Mathesis, Ηράκλειο: Πανεπιστημιακές Εκδόσεις Κρήτης, ΙΤΕ: <https://mathesis.cup.gr/courses/course-v1:ComputerScience+CS1.2+20C/course/>

2.2.Δ • ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ II (C++14)-ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Γλώσσα προγραμματισμού II (C++14)-Αντικειμενοστρεφής προγραμματισμός» εισάγει τους/τις εκπαιδευόμενους/-ες στη λογική του αντικειμενοστρεφούς προγραμματισμού. Η σύνταξη της γλώσσας C++14 είναι επηρεασμένη από τη γλώσσα προγραμματισμού C11, η οποία διδάσκεται στη μαθησιακή ενότητα «Γλώσσα προγραμματισμού I (C11)-Αλγοριθμική και δομές δεδομένων» του εξαμήνου Α' και άρα θα πρέπει να θεωρηθούν ήδη γνωστά όλα όσα διδάχθηκαν εκεί. Στην παρούσα μαθησιακή ενότητα, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με τα χαρακτηριστικά που διέπουν τον αντικειμενοστρεφή προγραμματισμό, θα εξοικειωθεί με τις ροές εισόδου/εξόδου της C++, τις μεθόδους δημιουργού/καταστροφέα, θα διδαχθεί τη λογική των εξαιρέσεων και της υπερφόρτωσης μεθόδων, συναρτήσεων και τελεστών, θα δημιουργήσει κλάσεις και αντικείμενα, θα χρησιμοποιήσει τρόπους κληρονομικότητας και πολυμορφισμού και θα εκμεταλλευτεί την ασφάλεια της ενθυλάκωσης των πεδίων κάθε αντικείμενου, δημιουργώντας έτσι πιο ανθεκτικές και πιο ασφαλείς εφαρμογές κονσόλας.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναγνωρίζουν τις βασικές προγραμματιστικές αρχές του αντικειμενοστρεφούς προγραμματισμού, βάσει των οποίων έχει υλοποιηθεί μια εφαρμογή,
- Επιλέγουν τα κατάλληλα μέλη κλάσεων (πεδία, μεθόδους κ.ά.) που θα χρησιμοποιήσουν στις εφαρμογές που θα αναπτύσσουν,
- Αναγνωρίζουν τη σημασία του πολυμορφισμού και της κληρονομικότητας,
- Αναπτύσσουν εφαρμογές με χρήση της γλώσσας προγραμματισμού C++14,
- Συνεισφέρουν με πρωτότυπο κώδικα στην κοινότητα των προγραμματιστών εφαρμογών,
- Τροποποιούν υπάρχοντα κώδικα, ανάλογα με τις προδιαγραφές των εφαρμογών που αναπτύσσουν,
- Αποφασίζουν για τη χρήση ή μη κατάλληλων βιβλιοθηκών, σχετικών με την ανάπτυξη εφαρμογών,
- Συνεργάζονται με άλλους/-ες εκπαιδευόμενους/-ες, συγκροτώντας ομάδες, με σκοπό τη δημιουργία και την επίλυση προβλημάτων των εφαρμογών,
- Υιοθετούν θετική στάση ως προς τις υπάρχουσες τεχνολογίες ανάπτυξης εφαρμογών.

Βασικές έννοιες και λέξεις-κλειδιά

Αντικειμενοστρέφεια / Κλάση / Μέθοδος / Ιδιότητα/Πεδίο/Γνώρισμα / Κληρονομικότητα / Πολυμορφισμός / Ενθυλάκωση / Υπερφόρτωση / Εξαιρέσεις / OpenGL GLUT / Δομές δεδομένων (Vector, Array, Stack, Queue).

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (6), Σύνολο (6).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΤΕΛΕΣΤΕΣ ΕΙΣΟΔΟΥ/ΕΞΟΔΟΥ ΣΤΗ C++14-ΧΩΡΟΙ ΟΝΟΜΑΤΩΝ-ΒΙΒΛΙΟΘΗΚΕΣ IOSTREAM, IOMANIP, TYPEINFO, ALGORITHM-ΑΝΤΙΣΤΟΙΧΗΣΗ ΕΝΝΟΙΩΝ ΤΗΣ C11 ΣΤΗ C++14, ΟΙ ΜΕΤΑΒΛΗΤΕΣ SIZE_T, WCHAR_T, CHAR16_T, CHAR32_T, ΝΕΟΙ ΤΡΟΠΟΙ ΑΡΧΙΚΟΠΟΙΗΣΗΣ
2	ΤΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΤΟΥ ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΟΥΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ-Η ΕΝΝΟΙΑ ΤΗΣ ΚΛΑΣΗΣ/ΜΕΘΟΔΩΝ/ΠΕΔΙΩΝ ΚΑΙ ΤΩΝ ΑΝΤΙΚΕΙΜΕΝΩΝ ΤΗΣ-ΔΗΜΙΟΥΡΓΙΑ ΕΡΓΟΥ ΣΕ ΠΟΛΛΑΠΛΑ ΑΡΧΕΙΑ ΚΩΔΙΚΑ, Η ΕΝΤΟΛΗ CONSTEXPR/CONST/MUTABLE, ΜΑΚΡΟΕΝΤΟΛΕΣ ΠΡΟΕΠΕΞΕΡΓΑΣΤΗ
3	ΣΥΝΑΡΤΗΣΕΙΣ ΠΟΥ ΕΠΙΣΤΡΕΦΟΥΝ ΑΝΑΦΟΡΑ/VOID*, ΣΥΝΑΡΤΗΣΕΙΣ INLINE, FRIEND-ΥΠΕΡΦΟΡΤΩΣΗ ΣΥΝΑΡΤΗΣΕΩΝ-ΠΡΟΤΥΠΑ-ΜΕΘΟΔΟΙ ΔΗΜΙΟΥΡΓΙΑΣ (DEFAULT, EXPLICIT, CUSTOM, COPY)/ΚΑΤΑΣΤΡΟΦΗΣ
4	ΥΛΟΠΟΙΗΣΗ ΣΤΟΙΒΑΣ/ΟΥΡΑΣ ΣΤΗ C++14
5	ΕΜΒΕΛΕΙΑ ΣΤΗ C++14: ΣΤΑΤΙΚΑ, ΤΟΠΙΚΑ ΚΑΙ ΚΑΘΟΛΙΚΑ ΑΝΤΙΚΕΙΜΕΝΑ-Η ΔΗΛΩΣΗ AUTO, REGISTER, VOLATILE-ΔΙΑΡΚΕΙΑ ΕΜΒΕΛΕΙΑΣ ΑΝΤΙΚΕΙΜΕΝΟΥ, Η ΕΝΤΟΛΗ EXTERN
6	ΥΠΕΡΦΟΡΤΩΣΗ ΤΕΛΕΣΤΩΝ-ΔΥΝΑΜΙΚΗ ΔΕΣΜΕΥΣΗ/ΑΠΟΔΕΣΜΕΥΣΗ ΜΝΗΜΗΣ-ΔΕΙΚΤΕΣ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗ ΜΝΗΜΗΣ/Ο ΤΕΛΕΣΤΗΣ THIS
7	Η ΕΝΤΟΛΗ ASM, ΕΙΔΗ CASTING, ΓΝΩΣΤΕΣ ΕΤΟΙΜΕΣ ΚΛΑΣΕΙΣ ΤΗΣ STL C++14: STRING, ARRAY, VECTOR, LIST-REGEX,
8	ΚΛΗΡΟΝΟΜΙΚΟΤΗΤΑ ΣΤΗ C++14: 5 ΕΙΔΗ-VIRTUAL/ABSTRACT/INTERFACE ΚΛΑΣΕΙΣ-COMPOSITION-AGGREGATION-ΣΧΕΣΕΙΣ IS-A/HAS-A/USES-A, Η ΕΝΤΟΛΗ OVERRIDE/FINAL
9	ΡΟΕΣ ΑΡΧΕΙΩΝ ΚΕΙΜΕΝΟΥ (BINARY, ANSI/ASCII)-ΕΞΑΙΡΕΣΕΙΣ
10	ΣΥΝΑΡΤΗΣΕΙΣ ΛΑΜΒΔΑ, ΓΡΑΦΙΚΑ ΣΤΗ C++14

Κείμενο αναφοράς

2.2.Δ.1 | ΤΕΛΕΣΤΕΣ ΕΙΣΟΔΟΥ/ΕΞΟΔΟΥ ΣΤΗ C++14-ΧΩΡΟΙ ΟΝΟΜΑΤΩΝ-ΒΙΒΛΙΟΘΗΚΕΣ IOSTREAM, IOMANIP, TYPEINFO, ALGORITHM-ΑΝΤΙΣΤΟΙΧΗΣΗ ΕΝΝΟΙΩΝ ΤΗΣ C11 ΣΤΗ C++14, ΟΙ ΜΕΤΑΒΛΗΤΕΣ SIZE_T, WCHAR_T, CHAR16_T, CHAR32_T, ΝΕΟΙ ΤΡΟΠΟΙ ΑΡΧΙΚΟΠΟΙΗΣΗΣ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τις έννοιες των χώρων ονομάτων (namespaces) και τα χαρακτηριστικά τους, με ιδιαίτερη έμφαση να δίνεται στον standard χώρο ονομάτων (std). Στη συνέχεια, θα αναλυθούν οι εντολές εισόδου/εξόδου ροών δεδομένων της βιβλιοθήκης iostream (cout/cerr, cin) και οι εντολές μορφοποίησης της βιβλιοθήκης iomanip (bool-(no)boolalpha-precision-setprecision-fixed-setw-setbase-(no)showpos-uppercase-internal-(no)showbase, setfill, fill). Επιπλέον, θα αναλυθεί ο τρόπος ενσωμάτωσης βιβλιοθηκών της C11, όταν αυτό είναι απαραίτητο (cstdlib/cstring/cstdio κ.ά.). Στη συνέχεια, θα πραγματοποιηθεί παραλληλισμός λειτουργιών της C11, που ισχύουν και στη C++14, και θα αναφερθούν καινούργια χαρακτηριστικά που επιτρέπει η γλώσσα C++14 (π.χ. τη χρήση του είδους μεταβλητής string και την εν γένει χρήση της βιβλιοθήκης string, τη χρήση μεταβλητών bool, με τιμές true/false, χωρίς τη χρήση της βιβλιοθήκης stdbool, τη χρήση for, με δήλωση μετρητών τοπικής εμβέλειας κ.ά.). Επιπλέον, θα αναφερθεί και ο νέος

τελεστής αρχικοποίησης “{}” (π.χ. `int x{0}`; αντί για το κλασικό `int x=0`;), ενώ θα αναφερθούν και νέα είδη μεταβλητών, όπως: `size_t`, `wchar_t`, `char16_t`/`char32_t`. Αναφορικά με το χρησιμοποιούμενο IDE (Integrated Development Environment), θα πρέπει να υποστηρίζει τις αλλαγές που περιέχει το πρότυπο C++14 του 2014 και προτείνεται ένα από τα Dev C++ 5.11/CodeBlocks και όχι τα DevC++ 6/Visual Studio. Τέλος, οι εκπαιδευόμενοι/-ες θα χρησιμοποιήσουν τις βιβλιοθήκες `typeinfo`/`algorithm`, για να ελέγχουν το είδος μεταβλητής και να πραγματοποιούν αλγορίθμους, π.χ. ταξινόμηση (`sort`), χωρίς επιπλέον υλοποίηση κώδικα. Έτσι, θα διαπιστωθεί ότι συνεχίζουν να ισχύουν στη C++14 όλα τα εργαλεία που διδάχτηκαν στη C11.

2.2.Δ.2 | ΤΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΤΟΥ ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΟΥΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ -Η ΕΝΝΟΙΑ ΤΗΣ ΚΛΑΣΗΣ/ΜΕΘΟΔΩΝ/ΠΕΔΙΩΝ ΚΑΙ ΤΩΝ ΑΝΤΙΚΕΙΜΕΝΩΝ ΤΗΣ- ΔΗΜΙΟΥΡΓΙΑ ΕΡΓΟΥ ΣΕ ΠΟΛΛΑΠΛΑ ΑΡΧΕΙΑ ΚΩΔΙΚΑ, Η ΕΝΤΟΛΗ `CONSTEXPR/CONST/MUTABLE`, ΜΑΚΡΟΕΝΤΟΛΕΣ ΠΡΟΕΠΕΞΕΡΓΑΣΤΗ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τα χαρακτηριστικά γνωρίσματα της αντικειμενοστρέφειας και του αντικειμενοστρεφούς προγραμματισμού, ξεκινώντας με την ενθυλάκωση/απόκρυψη δεδομένων (`data encapsulation/hiding`). Με αναφορά στη δομή δεδομένων `struct` ως βάση, οι σπουδαστές/-τριες θα παρατηρήσουν τον τρόπο που δομείται μια `class` και τις προεπιλογές εμβέλειας που αυτή έχει σε σύγκριση με τη `struct`. Στη συνέχεια, θα πραγματοποιηθεί εκτενής αναφορά στα συστατικά στοιχεία που συναποτελούν μια `class`, όπως είναι τα `private` πεδία και οι `public` μέθοδοι. Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα διδαχθούν τις μεθόδους `mutators`, οι οποίες αλλάζουν τις τιμές των `private` πεδίων (γνωστές και ως `setters`), και τις μεθόδους `accessors`, οι οποίες επιστρέφουν τις τιμές των `private` πεδίων (γνωστές και ως `getters`). Επιπλέον, θα διδαχθούν την υλοποίηση και άλλων δύο βοηθητικών μεθόδων, όπως είναι οι `setAll` και `print`, ώστε να αρχικοποιούν και να εμφανίζουν τις τιμές των αντικειμένων μιας `class` σύντομα. Επίσης, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη συσχέτιση των εντολών `const/mutable` μεταξύ μεθόδων και `private` πεδίων, ενώ θα παρατηρήσουν και τη διαφορά μεταξύ των διακριτικών `constexpr` και `const` για τη δήλωση σταθερών στοιχείων σε έναν κώδικα C++14. Τέλος, χρησιμοποιώντας μακροεντολές προεπεξεργαστή (`#define/undef/ifndef/ifdef/if defined/endif/if/elif/else κ.ά.`), οι εκπαιδευόμενοι/-ες θα διδαχθούν τον τρόπο κατακερματισμού κάθε κλάσης στο `.h` αρχείο με τη δήλωση των στοιχείων της κλάσης, στο `.cpp` αρχείο με την υλοποίηση των μεθόδων της κλάσης, και στο `driver` αρχείο (`main.cpp`), το οποίο θα ελέγχει την ορθή λειτουργία της κλάσης. Τα αρχεία `.h/.cpp` της κλάσης θα πρέπει να έχουν το ίδιο όνομα, ενώ το `driver` αρχείο όχι.

2.2.Δ.3 | ΣΥΝΑΡΤΗΣΕΙΣ ΠΟΥ ΕΠΙΣΤΡΕΦΟΥΝ ΑΝΑΦΟΡΑ/VOID *, ΣΥΝΑΡΤΗΣΕΙΣ INLINE, FRIEND-ΥΠΕΡΦΟΡΤΩΣΗ ΣΥΝΑΡΤΗΣΕΩΝ-ΠΡΟΤΥΠΑ-ΜΕΘΟΔΟΙ ΔΗΜΙΟΥΡΓΙΑΣ (DEFAULT, EXPLICIT, CUSTOM, COPY)/ΚΑΤΑΣΤΡΟΦΗΣ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τα είδη των συναρτήσεων που παρουσιάζουν ιδιαιτερότητες στη C++14. Πιο αναλυτικά, θα παρουσιαστούν συναρτήσεις που επιστρέφουν αναφορά ως επιστρεφόμενο τύπο, καθώς και οι συναρτήσεις που επιστρέφουν δείκτη ενώ είναι void. Επιπλέον, θα γίνει αναφορά στην προτροπή inline στις συναρτήσεις και στον τρόπο που ο compiler αποφαινεται για το αν θα υλοποιήσει ή όχι την προτροπή inline, στο πλαίσιο της βελτιστοποίησης του παραγόμενου αποτελέσματος. Επιπροσθέτως, θα πραγματοποιηθεί αναφορά στις φιλικές συναρτήσεις και στους λόγους που μια συνάρτηση επιλέγεται να έχει πρόσβαση στα private πεδία μιας κλάσης, χωρίς να είναι μέλος της κλάσης. Στη συνέχεια, πραγματοποιείται εκτενής αναφορά στο δεύτερο χαρακτηριστικό της αντικειμενοστρέφειας (πολυμορφισμού), μέσω της υπερφόρτωσης συναρτήσεων ως εργαλείο γενικής χρήσης ενός ονόματος συνάρτησης, το οποίο να λειτουργεί για πολλά είδη οντοτήτων (primitive μεταβλητών/αντικειμένων), ενώ οι εκπαιδευόμενοι/-ες διδάσκονται και τον λόγο ύπαρξης των templates ως το βέλτιστο είδος υπερφόρτωσης συναρτήσεων. Στη συνέχεια, γίνεται αναφορά στην ύπαρξη μεθόδου δημιουργού, διδάσκονται αναλυτικά όλα τα είδη μεθόδων δημιουργίας (default constructor, explicit constructor, custom constructor και copy constructor), ενώ γίνεται αναφορά στην ανάγκη ύπαρξης μεθόδου καταστροφείας (destructor) και στον λόγο ύπαρξης αυτής της ανάγκης στο πλαίσιο διαχείρισης της διαμοιραζόμενης μνήμης από την εφαρμογή, την οποία οι εκπαιδευόμενοι/-ες προγραμματίζουν. Επιπλέον, γίνεται σύνδεση της μεθόδου setAll κατά την υλοποίηση μιας μεθόδου constructor, ενώ αντίστοιχα πραγματοποιείται η σύνδεση μεταξύ της υλοποίησης του καταστροφείας και της βοηθητικής μεθόδου print. Τέλος, γίνεται αναφορά των συνθηκών στις οποίες λειτουργεί ο explicit constructor σε μια εφαρμογή C++14.

2.2.Δ.4 | ΥΛΟΠΟΙΗΣΗ ΣΤΟΙΒΑΣ/ΟΥΡΑΣ ΣΤΗ C++14

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν το τρόπο λειτουργίας των δομών δεδομένων Στοίβας (Stack) και Ουράς (Queue), και τον αλγόριθμο προτεραιοποίησης, τον οποίο υλοποιεί και ακολουθεί η κάθε δομή. Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη λογική LIFO (Last In First Out), η οποία ακολουθείται από τη δομή δεδομένων της Στοίβας (Stack), θα τους αναπαρασταθεί διαγραμματικά ο τρόπος ώθησης (push) ενός στοιχείου εντός της Στοίβας, ο τρόπος απώθησης (pop) ενός στοιχείου εκτός της Στοίβας, θα γίνει αναφορά στο φαινόμενο της υποχείλισης (προσπάθεια απώθησης ενός στοιχείου από μια κενή Στοίβα) και της υπερχείλισης (προσπάθεια ώθησης ενός στοιχείου μέσα σε μια γεμάτη Στοίβα). Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα υλοποιήσουν όλα τα παραπάνω σε Project τριών αρχείων (Stack.h και Stack.cpp για την κλάση και main.cpp για τον έλεγχο ορθής λει-

τουργίας της). Αντίστοιχα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη λογική FIFO (First In First Out), η οποία ακολουθείται από τη δομή δεδομένων της Ουράς (Queue), θα τους αναπαρασταθεί διαγραμματικά ο τρόπος εισαγωγής (enQueue) ενός στοιχείου εντός της Ουράς, ο τρόπος εξαγωγής (deQueue) ενός στοιχείου εκτός της Ουράς, θα γίνει αναφορά στο φαινόμενο της υποχείλισης (προσπάθειας εξαγωγής ενός στοιχείου από μια κενή Ουρά) και της υπερχείλισης (προσπάθειας εισαγωγής ενός στοιχείου μέσα σε μια γεμάτη Ουρά), ενώ στη συνέχεια οι εκπαιδευόμενοι/-ες θα υλοποιήσουν όλα τα παραπάνω σε Project τριών αρχείων (Queue.h και Queue.cpp για την κλάση και main.cpp για τον έλεγχο ορθής λειτουργίας της).

2.2.Δ.5 | ΕΜΒΕΛΕΙΑ ΣΤΗ C++14: ΣΤΑΤΙΚΑ, ΤΟΠΙΚΑ ΚΑΙ ΚΑΘΟΛΙΚΑ ΑΝΤΙΚΕΙΜΕΝΑ- Η ΔΗΛΩΣΗ AUTO, REGISTER, VOLATILE-ΔΙΑΡΚΕΙΑ ΕΜΒΕΛΕΙΑΣ ΑΝΤΙΚΕΙΜΕΝΟΥ, Η ΕΝΤΟΛΗ EXTERN

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν την εμβέλεια των μεταβλητών και των αντικειμένων. Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα υλοποιήσουν ασκήσεις, που θα χρησιμοποιούν καθολικές μεταβλητές και αντικείμενα κλάσεων, και τοπικές μεταβλητές και αντικείμενα κλάσεων, κάποια από τα οποία θα είναι στατικά, ενώ άλλα όχι. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τη δημιουργία αντικειμένων στη μνήμη με τη μορφή της Στοιβάς και την καταστροφή τους με βάση τη λογική LIFO που διδάχτηκε παραπάνω. Με τη βοήθεια της βιβλιοθήκης ctime, θα υλοποιηθούν αντικείμενα start και end με χρήση της clock_t, τα οποία θα προσμετρούν τη διάρκεια ζωής κάθε αντικειμένου μέσα στο πρόγραμμα σε δευτερόλεπτα, υποβοηθούμενα από τη χρήση της σταθεράς CLOCKS_PER_SEC. Έτσι, θα γίνει κατά νοητό στους/στις εκπαιδευόμενους/-ες το πότε δημιουργείται κάθε αντικείμενο, πότε καταστρέφεται και ποιο είναι το σύνολο των δευτερολέπτων που διήρκεσε μέσα στην εφαρμογή. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν τον αυτόματο τύπο δεδομένων (auto) και τους βαθμούς ελευθερίας που αυτός προσδίδει κατά τη συγγραφή κώδικα C++14, την προτροπή αποθήκευσης μεταβλητών στους καταχωρητές εντός της CPU (register), κάτι που, σε περίπτωση που επιτραπεί από την επικοινωνία του compiler με το Hardware, θα κάνει την εφαρμογή να είναι βέλτιστη σε απόδοση, ενώ, τέλος, οι εκπαιδευόμενοι/-ες θα διδαχθούν και το προσδιοριστικό βελτιστοποίησης volatile, το οποίο επιτρέπει να εκτελεστούν με διαφορετικό τρόπο κάποιες δηλώσεις και έλεγχοι, με αποτέλεσμα να εκτελείται γρηγορότερα η εφαρμογή. Τέλος, οι εκπαιδευόμενοι/-ες θα διδαχθούν την εντολή extern, τόσο αναφορικά με τη χρήση της σε συναρτήσεις όσο και με τη χρήση της σε μεταβλητές μιας εφαρμογής σε C++14.

2.2.Δ.6 | ΥΠΕΡΦΟΡΤΩΣΗ ΤΕΛΕΣΤΩΝ-ΔΥΝΑΜΙΚΗ ΔΕΣΜΕΥΣΗ/ΑΠΟΔΕΣΜΕΥΣΗ ΜΝΗΜΗΣ-ΔΕΙΚΤΕΣ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗ ΜΝΗΜΗΣ/Ο ΤΕΛΕΣΤΗΣ THIS

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν μια ακόμα έκφανση του πολυμορφισμού, μέσα από τον τρόπο υπερφόρτωσης τελεστών. Πιο

αναλυτικά, με τη δεσμευμένη λέξη `operator`, οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους υπερφόρτωσης τελεστών, όπως είναι οι τελεστές εισόδου δεδομένων (`>>`), εξόδου δεδομένων (`<<`), οι μοναδιαίοι τελεστές αύξησης και μείωσης (`++` και `--`) κ.λπ. Προτείνεται η χρήση των υπερφορτωμένων τελεστών σε παραδείγματα και ασκήσεις ημερολογίου, όπου θα τροποποιείται η μεταβολή (αύξηση/μείωση) της ημέρας ενός έτους, ενός μήνα, μιας ώρας κ.λπ., μέσα από υπερφόρτωση των αντίστοιχων τελεστών. Επιπλέον, οι εκπαιδευόμενοι/-ες θα μάθουν τρόπους για να δημιουργούν δυναμικές δομές δεδομένων και πίνακες, οι οποίοι θα δεσμεύουν δυναμικό μέγεθος στοιχείων (δηλαδή όχι εκ των προτέρων γνωστό), που στο τέλος του προγράμματος θα πρέπει να αποδεσμεύεται. Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα χρησιμοποιήσουν την εντολή `new` για τη δημιουργία δυναμικού πλήθους πινάκων και την εντολή `delete` για την εκκαθάριση των δεδομένων των πινάκων από τη μνήμη. Με δεδομένη τη γνώση των μεθόδων `constructor` και `destructor`, οι εκπαιδευόμενοι/-ες είναι σημαντικό να αντιστοιχήσουν την εντολή `new` ως μέρος της λειτουργικότητας ενός `constructor` και την εντολή `delete` ως μέρος της λειτουργικότητας ενός `destructor`. Τέλος, οι εκπαιδευόμενοι/-ες θα διδαχθούν τον τελεστή `this` και τον συνδυασμό χρήσης του με τους τελεστές `*` και `->`, για τη σύνταξη προγραμμάτων με χρήση δεικτών αντικειμένων. Επιπλέον, θα γίνει αναφορά σε `mutators` που «επιστρέφουν» αντικείμενο (`this`) και στη νέα χρήση του `ampersand` ως `call by reference` εργαλείο, χωρίς να είναι απαραίτητη η χρήση `*` κατά την υλοποίηση συνάρτησης.

2.2.Δ.7 | Η ΕΝΤΟΛΗ ASM, ΕΙΔΗ CASTING, ΓΝΩΣΤΕΣ ΕΤΟΙΜΕΣ ΚΛΑΣΕΙΣ ΤΗΣ STL C++14: STRING, ARRAY, VECTOR, LIST-RESEX

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν την εντολή `asm/_asm/asm_`, η οποία επιτρέπει σε έναν κώδικα C/C++ να ενσωματώσει εντολές `assembly`. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα αλληλεπιδράσουν με τα είδη μετατροπής τύπου (`casting`) σε όλες τις λειτουργίες και τα είδη τους: δυναμική μετατροπή τύπου (`dynamic_cast`), στατική μετατροπή τύπου (`static_cast`), σταθερή μετατροπή τύπου (`const_cast`) και επανερμηνεία μετατροπής τύπου (`reinterpret_cast`). Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα ενσωματώσουν τις βιβλιοθήκες `string`, `array`, `vector`, `list` και θα παρατηρήσουν τον τρόπο με τον οποίο οι εφαρμογές τους θα μπορούν να υποστηρίξουν την ίδια διαλειτουργικότητα, έχοντας λιγότερες γραμμές υλοποιημένου κώδικα. Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα διδαχθούν τις μεθόδους `length`, `size`, `getline`, `substr`, `replace`, `erase`, `c_str`, `find`, `rfind`, `append`, `insert`, `compare` της κλάσης `string`, τις μεθόδους `at`, `get`, `operator`, `front`, `back`, `size`, `max_size`, `swap`, `empty`, `fill` της κλάσης `array`, τις μεθόδους `begin`, `end`, `rbegin`, `rend`, `cbegin`, `cend`, `crbegin`, `crend`, `size`, `max_size`, `capacity`, `resize`, `empty`, `shrink_to_fit`, `reserve`, `reference_operator`, `at`, `front`, `back`, `data`, `assign`, `push_back`, `pop_back`, `insert`, `erase`, `swap`, `clear`, `emplace`, `emplace_back` της κλάσης `vector`, καθώς και τις μεθόδους `front`, `back`, `push_front`, `push_back`, `pop_front`, `pop_back`, `insert`, `size`, `begin`, `end`, `front`, `back`, `list::begin`, `list::end`, `list`

rbegin(), rend(), list cbegin(), cend(), list crbegin(), crend(), empty(), insert(), erase(), assign(), remove(), reverse(), size(), list resize(), sort(), list max_size(), list unique(), list::emplace_front(), list::emplace_back(), list::clear(), list::operator, list::swap(), list splice(), list merge(), list emplace() της κλάσης list. Τέλος, οι εκπαιδευόμενοι/-ες θα κάνουν μια πρώτη γνωριμία με τις κανονικοποιημένες εκφράσεις (regular expressions-regex), με απλά παραδείγματα για τη χρήση της λειτουργίας τους.

2.2.Δ.8 | ΚΛΗΡΟΝΟΜΙΚΟΤΗΤΑ ΣΤΗ C++14: 5 ΕΙΔΗ-VIRTUAL/ABSTRACT/INTERFACE ΚΛΑΣΕΙΣ-COMPOSITION-AGGREGATION-ΣΧΕΣΕΙΣ IS-A/HAS-A/USES-A, Η ΕΝΤΟΛΗ OVERRIDE/FINAL

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν την κληρονομικότητα, το τρίτο και τελευταίο χαρακτηριστικό του αντικειμενοστρεφούς προγραμματισμού. Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα διδαχθούν την ύπαρξη του χαρακτηρισμού protected στη θέση του private και θα πρέπει να υλοποιήσουν τα πέντε είδη κληρονομικότητας (Απλή, Πολλαπλή, Πολυεπίπεδη, Ιεραρχική και Υβριδική), εργαζόμενοι/-ες πάνω σε προγράμματα-παραδείγματα για την κάθε περίπτωση. Επιπλέον, θα γίνει αναφορά στις εικονικές (virtual) κλάσεις, στις αφηρημένες (abstract) κλάσεις και στις κλάσεις-διεπαφές (interfaces). Στο σημείο αυτό, οι εκπαιδευόμενοι/-ες θα διδαχθούν το πρόβλημα του “dreaded diamond” και τον τρόπο επίλυσής του. Στη συνέχεια, θα γίνει παρουσίαση τη λειτουργίας της σύνθεσης κλάσεων (class composition) μέσα από απλές υλοποιήσεις σε ενιαίο project, και κατόπιν θα πραγματοποιηθεί εκμάθηση της συνάθροισης κλάσεων (class aggregation). Μόλις αποσαφηνιστούν οι λεπτές διαφορές ανάμεσα στην κληρονομικότητα κλάσεων, στη συνάθροιση κλάσεων και στη σύνθεση κλάσεων, θα διδαχθεί η αντιπαράθεση των τριών ειδών σχέσεων του μοντέλου Οντοτήτων συσχετίσεων με τις αντίστοιχες τρεις λειτουργίες. Πιο συγκεκριμένα, θα εξηγηθούν οι λόγοι που αποδίδεται στις κληρονομούμενες κλάσεις η σχέση ΕΙΝΑΙ-ΕΝΑ (IS-A), στις συντεθειμένες κλάσεις η σχέση ΧΡΗΣΙΜΟΠΟΙΕΙ-ΕΝΑ (USES-A) και στις συναθροιζόμενες κλάσεις η σχέση ΕΧΕΙ-ΕΝΑ (HAS-A). Τέλος, θα γίνει αναφορά στα προσδιοριστικά override και final, τα οποία δηλώνουν την ιδιότητα της επέκτασης ή μη της λειτουργία μιας virtual μεθόδου.

2.2.Δ.9 | ΡΟΕΣ ΑΡΧΕΙΩΝ ΚΕΙΜΕΝΟΥ (BINARY, ANSI/ASCII)-ΕΞΑΙΡΕΣΕΙΣ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους αλληλεπίδρασης μιας προγραμματιστικής εφαρμογής με αρχεία κειμένου. Πιο αναλυτικά, θα διδαχθούν το είδος ροών αρχείου fstream, ifstream και ofstream και των συναρτήσεων open και close, οι οποίες ανοίγουν και κλείνουν την πρόσβαση σε ένα αρχείο κειμένου στον σκληρό δίσκο. Στη συνέχεια, θα πραγματοποιηθεί αναφορά στις εντολές εισόδου και εξόδου δεδομένων σε αρχεία κειμένου, όπως είναι οι getline, is, out κ.ά., ενώ οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους χειρισμού αρχείων ,

μέσω των εντολών `seekg`, `seekr` κ.λπ. Επιπλέον, θα διδαχθούν τους τρόπους δημιουργίας αμυντικών μηχανισμών, μέσα από τη χρήση εξαιρέσεων (`exceptions`). Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη δομή `try-catch`, μαζί με τις ιδιαιτερότητες στη χρήση αγκίστρων κατά τη σύνταξη του κάθε μπλοκ κώδικα (π.χ. σε ένα `catch block` είναι υποχρεωτική η ύπαρξη αγκίστρων, ακόμη κι όταν υπάρχει μόνο μια εντολή μέσα στο `catch block`). Ενθαρρύνεται η χρήση παραδειγμάτων υλοποίησης αμυντικών μηχανισμών που να βοηθούν την αντιπαραβολή με `try-catch` και χωρίς αυτά, ώστε να φαίνεται η διαφορά κατά την υλοποίηση. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη χρήση των πολλαπλών `catch blocks` και του τερματικού `catch block` (...). Επιπρόσθετα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη χρήση της βιβλιοθήκης `exception` και τα πιο συχνά `exceptions` που χρησιμοποιούνται. Θα γίνει χρήση της εντολής `throw`, της εντολής `what`, και, τέλος, θα πραγματοποιηθεί αναφορά στις εντολές `finally`, `except` και `leave` και στον τρόπο που αυτές καθορίζουν τον χειρισμό εξαιρέσεων στη C++14.

2.2.Δ.10 | ΣΥΝΑΡΤΗΣΕΙΣ ΛΑΜΒΔΑ-ΓΡΑΦΙΚΑ ΣΤΗ C++14

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη χρήση των ανώνυμων συναρτήσεων (`anonymous functions` ή `lambda functions`) και τους λόγους που η χρήση τους βολεύει πολύ κατά τη διαδικασία υλοποίησης ενός προγράμματος σε γλώσσα προγραμματισμού C++14. Στη συνέχεια, θα γνωρίσουν τρόπους δημιουργίας γραφικών σχημάτων και αναπαραστάσεων δισδιάστατων και τρισδιάστατων σχημάτων, τα οποία θα υποστηρίζουν εφέ κίνησης, ώστε να μπορούν να δημιουργήσουν απλές εφαρμογές κονσόλας που θα υποστηρίζουν χρώμα και κίνηση εκτός από λειτουργικότητα. Στο πλαίσιο αυτό, θα πραγματοποιηθεί αναφορά στη βιβλιοθήκη `OpenGL Utility Toolkit (GLUT)`, η οποία είναι ενσωματωμένη στα αρχεία βιβλιοθήκης `glut.h` και `glut32.dll`, τα οποία περιλαμβάνουν τα πρωτότυπα των συναρτήσεων γραφικών που θα χρησιμοποιηθούν. Τα αρχεία αυτά δεν περιλαμβάνονται ήδη σε κάποιο IDE και θα πρέπει να γίνει λήψη τους και αποθήκευσή τους στον αντίστοιχο κατάλογο που υπάρχουν και τα υπόλοιπα αρχεία βιβλιοθήκης συστήματος, μαζί με το αντίστοιχο αρχείο `libglut32.a`. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν εντολές, όπως: `glClearColor`, `glClear(GL_COLOR_BUFFER_BIT)`, `glColor3f`, `glRectf`, `glBegin(GL_LINE_LOOP)`, `glBegin(GL_POLYGON)`, `glVertex2f`, `glPolygonMode(GL_FRONT, GL_FILL)`, `glutPostRedisplay()` κ.λπ. Οι παραπάνω εντολές θα υπάρχουν ανάμεσα στις εξής εντολές: `glutInit`, `glutInitWindowPosition`, `glutInitWindowSize`, `glutInitDisplayMode(GLUT_SINGLE|GLUT_RGBA)`, `glutCreateWindow`, `glMatrixMode(GL_PROJECTION)`, `gluOrtho2D`, `glutDisplayFunc`, `glutMainLoop` κ.λπ. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν και τη χρήση `mouse-keyboard listeners`, ώστε να δημιουργούν εφαρμογές που θα υποστηρίζουν την είσοδο δεδομένων από πληκτρολόγιο και ποντίκι αντίστοιχα. Με τις παραπάνω λειτουργίες, οι εκπαιδευόμενοι/-ες θα σχεδιάζουν

παιχνίδια, όπως π.χ. μια 3x3 τρίλιζα με χρώματα, και θα είναι σε θέση να σχεδιάσουν και εφαρμογές 3D, όπως η επίλυση ενός 3x3 κύβου ρούμπικ κ.ά.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Behrouz, A. F. & Richard, F. G. (2022). *Προγραμματισμός με C++*. *Αντικειμενοστρεφής Προσέγγιση*. Αθήνα: Εκδόσεις Επίκεντρο.
2. Stroustrup, B. (2018). *Προγραμματισμός με τη C++*. Αθήνα: Εκδόσεις Παπασωτηρίου.

Συμπληρωματικές

1. Χατζηγιαννάκης, Ν. Μ. (2014). *Η γλώσσα C++ σε βάθος*. Αθήνα: Εκδόσεις Κλειδάριθμος.
2. Harvey, M. & Deitel, P. (2015). *C++ προγραμματισμός*. Αθήνα: Εκδόσεις Μ. Γκιούρδα.

2.2.E • ΑΝΟΙΚΤΑ ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ ΒΑΣΙΣΜΕΝΑ ΣΤΟ LINUX

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Ανοικτά λειτουργικά συστήματα βασισμένα στο Linux» εισάγει τους/τις εκπαιδευόμενους/-ες στο ΛΣ Linux και στη βασική ιδέα του δημιουργού του Linus Torvalds. Η ενότητα αυτή είναι συμπληρωματική της μαθησιακής ενότητας «Σύγχρονα λειτουργικά συστήματα» του εξαμήνου Α΄ σπουδών, δίνοντας έμφαση στο ελεύθερο λογισμικό/λογισμικό ανοικτού κώδικα (ΕΛ/ΛΑΚ). Στην παρούσα μαθησιακή ενότητα, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με την αρχιτεκτονική του ΛΣ Linux, τις διανομές του Linux και την εγκατάσταση διανομών Linux σε εικονικές μηχανές. Επίσης, ενισχύεται η εξοικείωση με τη γραμμή εντολών, τα διάφορα είδη συστημάτων αρχείων, τη βασική πλοήγηση και τα χαρακτηριστικά καταλόγων και αρχείων του Linux (manual pages, διαχείριση αρχείων μέσω εντολών, wildcards, δικαιώματα καταλόγων και αρχείων [permissions], φίλτρα [filters], grep και regular expressions, pipelining και redirection, process management κ.λπ.). Επιπλέον, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με κειμενογράφους στη γραμμή εντολών, με τα είδη φλοιού και με τη δημιουργία απλών shell scripts. Τέλος, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με το λειτουργικό σύστημα Android, τη σχέση του με το Linux και με τη δημιουργία μιας απλής Android εφαρμογής, μέσω του λογισμικού Android Studio.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Υιοθετούν θετική στάση ως προς το λειτουργικό σύστημα (ΛΣ) Linux,
- Περιγράφουν τη φιλοσοφία του λειτουργικού συστήματος (ΛΣ) Linux,
- Εγκαθιστούν διανομές του Linux (μέσω εικονικής μηχανής),
- Εξοικειώνονται με τα διαμερίσματα σκληρού δίσκου που απαιτούνται σε ένα ανοικτό λειτουργικό σύστημα του Linux, το σύστημα αρχείων του, τη γραμμή εντολών του (κέλυφο/φλοιό-shell) και τις βασικές εντολές του,
- Αναλύουν τους τρόπους λειτουργίας του κειμενογράφου σε διανομές του Linux και τους τρόπους συγγραφής shell scripting,
- Δημιουργούν μια απλή Android εφαρμογή, έχοντας έρθει σε επαφή με το ΛΣ Android και μέσω του Android Studio,
- Αναγνωρίζουν τα πλεονεκτήματα των ανοικτών λειτουργικών συστημάτων με αναφορά τόσο σε ΛΣ που απαιτούν άδεια χρήσης όσο και σε αυτά που βασίζονται στο ΕΛ/ΛΑΚ,
- Επιλέγουν την πιο κατάλληλη διανομή του λειτουργικού συστήματος Linux που θα πρέπει να εγκαταστήσουν, ανάλογα με την περίπτωση χρήσης του,
- Αναγνωρίζουν τη διαφορά μεταξύ των οικογενειών ανοικτών λειτουργικών συστημάτων και την αντιστοίχιση της κάθε μιας, ανάλογα με τις ανάγκες της επιχείρησης/οργανισμού,
- Αναπτύσσουν τρόπους βέλτιστης παραμετροποίησης του ανοικτού λειτουργικού συστήματός τους, με σκοπό να εξυπηρετούν τον σκοπό χρήσης του,
- Αναλύουν τις πιθανές αιτίες βλάβης του ανοικτού λειτουργικού συστήματος ενός ηλεκτρονικού υπολογιστή,
- Αποφασίζουν για τη χρήση ή μη κατάλληλων εργαλείων διαχείρισης υλικού/εφαρμογών/διεργασιών, εμφάνισης κ.λπ., που προσφέρει ένα ανοικτό λειτουργικό σύστημα,
- Διακρίνουν τις διαφορές μεταξύ των οικογενειών λειτουργικών συστημάτων Linux, συγκρίνοντας επιτραπέζιες εκδόσεις, καθώς και εκδόσεις για φορητές συσκευές (tablet/smartphone κ.λπ.).

Βασικές έννοιες και λέξεις-κλειδιά

Unix / Linux / Debian, Redhat, Slackware / Εικονική μηχανή (virtual device) / Πυρήνας (kernel) / Φλοιός (shell) / Shell scripts / Διεργασία (Process).

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (3), Σύνολο (3).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

Α/Α	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΑ ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ LINUX
2	ΕΙΣΑΓΩΓΗ ΣΤΗ ΧΡΗΣΗ ΦΛΟΙΟΥ ΚΑΙ ΤΗ ΣΥΝΤΑΞΗ ΕΝΤΟΛΩΝ
3	ΕΙΣΑΓΩΓΗ ΣΤΟ ΣΥΣΤΗΜΑ ΑΡΧΕΙΩΝ
4	ΔΙΚΑΙΩΜΑΤΑ ΠΡΟΣΒΑΣΗΣ ΣΕ ΑΡΧΕΙΑ ΚΑΙ ΚΑΤΑΛΟΓΟΥΣ
5	ΔΙΕΡΓΑΣΙΕΣ
6	ΕΙΣΑΓΩΓΗ ΣΤΑ ΣΕΝΑΡΙΑ ΦΛΟΙΟΥ (SHELL SCRIPTS)
7	ΤΟ ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ ANDROID

Κείμενο αναφοράς

2.2.E.1 | ΕΙΣΑΓΩΓΗ ΣΤΑ ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ LINUX

Στην εν προκειμένω ενότητα, γίνεται εισαγωγή στα λειτουργικά συστήματα UNIX και LINUX και τονίζεται η βασική διαφορά τους, η οποία είναι ότι το LINUX είναι μια παραλλαγή του UNIX, ειδικά σχεδιασμένη για την αρχιτεκτονική των PC. Στη συνέχεια, γίνεται παρουσίαση της αρχιτεκτονικής του συστήματος Linux και των δομικών συστατικών του: ο πυρήνας (kernel), το βασικότερο κομμάτι του λειτουργικού συστήματος που αναλαμβάνει τη διαχείριση του υλικού, τον προγραμματισμό των συσκευών και τη διαχείριση των διεργασιών, οι εφαρμογές (applications), οι οποίες επικοινωνούν με τον χρήστη και εκτελούν λειτουργίες, ο φλοιός (shell), ο οποίος είναι μια εφαρμογή που προωθεί τις εντολές των χρηστών/-τριών στον πυρήνα, και τα γραφικά περιβάλλοντα (graphical environments), τα οποία παρέχουν γραφικό περιβάλλον διεπαφής με τον/τη χρήστη/-τρια (GNOME, KDE). Στη συνέχεια, γίνεται αναφορά στα πλεονεκτήματα του Linux έναντι άλλων λειτουργικών συστημάτων και τονίζονται ιδιαίτερα: ο απόλυτος έλεγχος του διαχειριστή πάνω στον υπολογιστή και στο λογισμικό, η παραμετροποίηση όλων των στοιχείων του υπολογιστή και τα μικρότερα κενά ασφάλειας. Παρουσιάζονται, επίσης, οι κυριότερες διανομές των Linux: debian, redhat και slackware. Γίνεται αναφορά στις διαφορές τους, οι οποίες αφορούν κυρίως τον τρόπο με τον οποίο γίνεται η ρύθμιση των παραμέτρων (configuration) του λειτουργικού συστήματος, καθώς και τον τρόπο εγκατάστασης και διαχείρισης προγραμμάτων και λειτουργιών. Τέλος, γίνεται σύντομη παρουσίαση της διανομής Ubuntu (βασίζεται στη debian), που προτείνεται να χρησιμοποιηθεί στο πλαίσιο του μαθήματος. Γίνεται επίσης εγκατάσταση της τελευταίας έκδοσης του Ubuntu, το οποίο κατεβάζουμε από τον επίσημο δικτυακό τόπο. Για την εγκατάστασή του σε Windows, απαιτείται η εγκατάσταση μιας εικονικής μηχανής (virtual machine), η οποία θα τρέχει το Ubuntu σε υπολογιστή με Windows. Προτείνεται η χρήση του VMWare (<http://www.vmware.com/>) ή εναλλακτικά του Virtual Box (<https://www.virtualbox.org/>).

2.2.E.2 | ΕΙΣΑΓΩΓΗ ΣΤΗ ΧΡΗΣΗ ΦΛΟΙΟΥ ΚΑΙ ΤΗ ΣΥΝΤΑΞΗ ΕΝΤΟΛΩΝ

Στην υποενότητα αυτή, παρουσιάζεται η έννοια του φλοιού (shell), μιας εφαρμογής που προωθεί τις εντολές των χρηστών/-τριών στον πυρήνα και παρουσιάζεται ο φλοιός του λειτουργικού Linux που εγκαταστάθηκε. Τονίζεται ότι υπάρχουν πολλοί τύποι φλοιών και ότι ο τρόπος σύνταξης και ερμηνείας των εντολών εξαρτάται από τον φλοιό που χρησιμοποιείται. Στο πλαίσιο του μαθήματος, προτείνεται να χρησιμοποιηθεί ο φλοιός bash. Για να χρησιμοποιηθεί ο φλοιός, πρέπει να ανοίξει το bash prompt, δηλαδή η γραμμή εντολών, στην οποία καταχωρούνται οι εντολές. Στη συνέχεια, γίνεται εισαγωγή στη σύνταξη εντολών, στα ορίσματα εντολών και στα flags. Δίνονται παραδείγματα εκτέλεσης απλών εντολών στο κέλυφος, όπως των echo, touch, cat, tar και ls. Ιδιαίτερα τονίζεται η εντολή man, η οποία χρησιμοποιείται για να εμφανιστεί το εγχειρίδιο (manual page) μιας εντολής. Κάθε εντολή του Linux συνήθως συνοδεύεται από ένα manual page, που περιέχει πληροφορίες, σχετικές με τη χρήση της συγκεκριμένης εντολής, τις διαθέσιμες επιλογές και τα παραδείγματα χρήσης. Γίνεται επίσης αναφορά στην έννοια των φίλτρων (filters), τα οποία μπορούν να χρησιμοποιηθούν για την επεξεργασία της εξόδου των εντολών, και παρουσιάζονται δημοφιλή φίλτρα, όπως το sort, uniq, wc. Τονίζεται ιδιαίτερα το φίλτρο grep, που χρησιμοποιείται για την αναζήτηση κειμένου με βάση κείμενο ή κανονικές εκφράσεις (regular expressions). Αναλύεται η έννοια των κανονικών εκφράσεων και γίνεται αναφορά στις πιο ευρέως χρησιμοποιούμενες κανονικές εκφράσεις (ενδεικτικά *, .?, [],).

2.2.E.3 | ΕΙΣΑΓΩΓΗ ΣΤΟ ΣΥΣΤΗΜΑ ΑΡΧΕΙΩΝ

Εδώ, παρουσιάζεται το σύστημα αρχείων (file system), το οποίο νοείται ως το σύνολο των αρχείων, καταλόγων και αποθηκευτικών μέσων, και ο τύπος συστήματος αρχείων (file system type), όρος που περιγράφει τη δομή και την αντιστοίχιση των αρχείων σε φυσικά τμήματα του δίσκου. Γίνεται αναφορά στους γνωστότερους τύπους συστημάτων αρχείων (FAT, NTFS, Ext) και αναλύονται τα χαρακτηριστικά και οι διαφορές τους. Στη συνέχεια, παρουσιάζεται το σύστημα αρχείων στο Linux και οι διαφορές του από το σύστημα αρχείων των Windows. Γίνεται αναφορά στο root directory και στο home directory. Επίσης, αναλύεται η έννοια της διαδρομής (path), και δίνονται παραδείγματα διαδρομών σε Linux και Windows. Γίνεται επίσης αναφορά στη σχετική διαδρομή (relative path) και στον τρέχοντα κατάλογο εργασίας (current working directory). Ακολουθεί πλοήγηση στα αρχεία του Linux με χρήση του shell και της εντολής cd. Το επόμενο θέμα της υποενότητας είναι η διαχείριση καταλόγων και αρχείων μέσω εντολών. Παρουσιάζονται οι βασικές εντολές διαχείρισης καταλόγων (mkdir, rmdir) και οι εντολές διαχείρισης αρχείων (cp, mv, rm). Επιπρόσθετα, παρουσιάζονται οι wildcards (χαρακτήρες μπαλαντέρ), ειδικοί χαρακτήρες που χρησιμοποιούνται για να υποδηλώσουν ένα σύνολο αρχείων. Γίνεται επίδειξη της χρήσης των βασικότερων wildcards (*, ?, [],), εξηγείται η διαφορά τους από τις κανονικές εκφράσεις και τονίζεται η σημασία τους για τη μαζική επεξεργασία αρχείων.

2.2.E.4 | ΔΙΚΑΙΩΜΑΤΑ ΠΡΟΣΒΑΣΗΣ ΣΕ ΑΡΧΕΙΑ ΚΑΙ ΚΑΤΑΛΟΓΟΥΣ

Το πρώτο θέμα της υποενότητας είναι η ιεραρχία των χρηστών/-τριών στα συστήματα Linux. Γίνεται αναφορά στην έννοια του/της χρήστη/-τριας (user), του/της χρήστη/-τριας με τα υψηλότερα δικαιώματα πρόσβασης (root user) και της ομάδας χρηστών/-τριών (user group). Γίνεται αναφορά στα δικαιώματα του χρήστη root και παρουσιάζεται η εντολή sudo. Το επόμενο θέμα είναι οι άδειες πρόσβασης (permissions) σε αρχεία και καταλόγους. Παρουσιάζονται δικαιώματα ανάγνωσης (read), εγγραφής (write) και εκτέλεσης (execute), που αφορούν αρχεία και καταλόγους. Επίσης, τονίζεται ότι οι άδειες πρόσβασης ορίζονται πάντα για τρεις κατηγορίες χρηστών/-τριών: για τον/την ιδιοκτήτη/-τρια του αρχείου/καταλόγου (owner), για την ομάδα ιδιοκτησίας του αρχείου/καταλόγου (group) και για όλες τις υπόλοιπες περιπτώσεις (others). Γίνεται, επίσης, αναφορά στην εντολή "ls-l", η οποία παρουσιάζει τις άδειες πρόσβασης σε αρχεία/καταλόγους ανά κατηγορία χρηστών/-τριών. Ακολουθεί η διαχείριση των δικαιωμάτων πρόσβασης (δηλαδή η μεταβολή δικαιωμάτων), η οποία μπορεί να γίνει μόνο από τον/την ιδιοκτήτη/-τρια του αρχείου και από τον/τη χρήστη/-τρια root. Γίνεται αναφορά στην εντολή chmod, η οποία χρησιμοποιείται για τη μεταβολή των δικαιωμάτων των αρχείων, και παρουσιάζονται τα ορίσματα και ο τρόπος χρήσης της. Επιπροσθέτως, παρουσιάζονται οι εντολές chown, chgrp και umask, και δίνονται παραδείγματα χρήσης τους.

2.2.E.5 | ΔΙΕΡΓΑΣΙΕΣ

Παρουσιάζεται η έννοια της διεργασίας (process), η οποία είναι ένα στιγμιότυπο ενός προγράμματος που βρίσκεται σε εξέλιξη. Γίνεται αναφορά στον τρόπο που δημιουργούνται οι διεργασίες (είτε από τον/τη χρήστη/-τρια κατά την εκτέλεση μιας εντολής είτε αυτόματα από το σύστημα) και τονίζεται ότι το λειτουργικό σύστημα είναι υπεύθυνο για τη διαχείριση των διεργασιών. Παρουσιάζονται οι καταστάσεις, στις οποίες μπορούν να βρίσκονται οι διεργασίες και συγκεκριμένα: δημιουργήθηκε (created), σε αναμονή (waiting), σε εκτέλεση (running), σε αναμονή πόρου (blocked) και στη μνήμη swap (swapped). Τονίζεται ότι το λειτουργικό σύστημα διατηρεί μια λίστα με όλες τις ενεργές διεργασίες. Γίνεται αναφορά στη διαδικασία εκκίνησης του συστήματος (boot) και στη βασική διεργασία init, η οποία ενεργοποιεί το λειτουργικό σύστημα. Παρουσιάζονται οι βασικές εντολές διαχείρισης των διεργασιών, όπως είναι η ps και η pstree (για την προβολή των διεργασιών), η fork και η exec (για την εκτέλεση διεργασιών) και η kill (τερματισμός διεργασιών). Στη συνέχεια, παρουσιάζεται η έννοια της ανακατεύθυνσης (redirection), μια τεχνική που χρησιμοποιείται για την αλλαγή της κατεύθυνσης της εισόδου ή της εξόδου μιας εντολής, και δίνονται παραδείγματα. Τέλος, παρουσιάζεται η έννοια της διασωλήνωσης (pipelining), μια τεχνική που χρησιμοποιείται για τη σύνδεση της εξόδου μιας εντολής με την είσοδο μιας άλλης εντολής, επιτρέποντας σε δύο εντολές

να επικοινωνούν μεταξύ τους και να δημιουργούν πιο σύνθετες εργασίες. Δίνονται επίσης παραδείγματα διασώληνωσης εντολών.

2.2.E.6 | ΕΙΣΑΓΩΓΗ ΣΤΑ ΣΕΝΑΡΙΑ ΦΛΟΙΟΥ (SHELL SCRIPTS)

Παρόλο που η έννοια του φλοιού έχει αναφερθεί σε προηγούμενη υποενότητα, προτείνεται να γίνει μια σύντομη εισαγωγή. Αναφέρεται ότι, εκτός από απλές εντολές που εισάγονται απευθείας στη γραμμή εντολών, ο/η χρήστης/-τρια μπορεί να γράψει μια ακολουθία εντολών σε ένα αρχείο, το οποίο είναι και το shell script (σενάριο φλοιού), και στη συνέχεια ο φλοιός, με την κατάλληλη εντολή, θα εκτελέσει τις εντολές στο αρχείο. Προτείνεται να χρησιμοποιηθεί ένας κειμενογράφος για τη σύνταξη των shell scripts και η χρήση του Visual Studio Code, που είναι δωρεάν και διαθέτει διάφορες επεκτάσεις για την υποστήριξη των shell scripts. Επιδεικνύεται η εκτέλεση ενός shell script μέσα από τη γραμμή εντολών. Γίνεται, επίσης, αναφορά στις άδειες χρήσης για την εκτέλεση των scripts, και στις τεχνικές αποσφαλμάτωσης (με χρήση των ορισμάτων `n` και `x`). Στη συνέχεια, παρουσιάζονται οι μεταβλητές κελύφους και τονίζεται ότι όλες είναι αλφαριθμητικού τύπου. Παρουσιάζεται ο τελεστής εκχώρησης τιμής σε μεταβλητή και ο τρόπος εκχώρησης μιας τιμής που δίνει ο/η χρήστης/-τρια, ως είσοδο από το πληκτρολόγιο (εντολή `read`). Τονίζεται, επίσης, η χρήση εισαγωγικών, η ειδική σημασία τους και ο τρόπος προσπέλασης της τιμής μιας μεταβλητής. Τέλος, παρουσιάζεται ο τρόπος εκτέλεσης αριθμητικών πράξεων και οι επιτρεπόμενοι αριθμητικοί τελεστές.

2.2.E.7 | ΤΟ ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ ANDROID

Εδώ, γίνεται παρουσίαση του λειτουργικού συστήματος Android, το οποίο χρησιμοποιεί μια προσαρμοσμένη έκδοση του πυρήνα του Linux και της αρχιτεκτονικής του. Επίσης, γίνεται αναφορά στο ολοκληρωμένο περιβάλλον ανάπτυξης (Android studio) και στις γλώσσες προγραμματισμού (Kotlin, Java), που χρησιμοποιούνται για την ανάπτυξη Android εφαρμογών. Στη συνέχεια, γίνεται η εγκατάσταση του Android studio και παρουσιάζονται οι βασικές λειτουργίες του. Επίσης, δημιουργείται μια εικονική συσκευή (virtual device), στην οποία δοκιμάζονται οι εφαρμογές. Το επόμενο θέμα της υποενότητας είναι η παρουσίαση των βασικών βημάτων ανάπτυξης μιας Android εφαρμογής. Το πρώτο βήμα προτείνεται να είναι η υλοποίηση της γραφικής διεπαφής της εφαρμογής. Πρώτα προστίθενται οι πόροι της εφαρμογής (εικονίδια, μενού, χρώματα) και στη συνέχεια σχεδιάζεται η γραφική διεπαφή με τον layout editor (κατά προτίμηση). Το δεύτερο βήμα είναι η υλοποίηση της εφαρμογής. Εδώ, το πρώτο βήμα είναι η σύνδεση της διεπαφής με τον κώδικα, ορίζονται δηλαδή οι ενέργειες του/της χρήστη/-τριας στα διάφορα στοιχεία της διεπαφής. Στη συνέχεια, υλοποιούνται οι λειτουργίες που υλοποιούνται μετά από κάποια ενέργεια του/της χρήστη/-τριας. Το τρίτο βήμα είναι η εκσφαλμάτωση (debugging) και ο έλεγχος της εφαρμογής στην εικονική συσκευή που έχει δημιουργηθεί. Τέλος,

προτείνεται να υλοποιήσουν οι σπουδαστές/-τριες μια απλή διαδραστική εφαρμογή, για παράδειγμα ο υπολογισμός του δείκτη μάζας σώματος, βάσει ύψους και βάρους που θα δίνονται από τον/τη χρήστη/-τρια.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Κανίδης, Ε., Κωνσταντοπούλου, Μ. Δ., Λέων, Π. & Μόρμορης, Ε. *Εισαγωγή στα Λειτουργικά Συστήματα*. Ανοιχτά Μαθήματα ΕΕΛΛΑΚ: <https://elearn.ellak.gr/course/view.php?id=195>
2. Ομάδα του Εγχειριδίου του Ubuntu. (2010). *Ξεκινώντας με το Ubuntu 10.04* [Ηλεκτρονική έκδοση]: <https://ubuntu-manual.org/?lang=el>
3. Σιδηρόπουλος, Α. (2015). *Εισαγωγή στα λειτουργικά συστήματα με χρήση και προγραμματισμό του Κελύφους*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοιχτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/2474>

Συμπληρωματικές

1. Ο επίσημος ιστότοπος του Ubuntu: <https://ubuntu.com/>
2. Ο επίσημος ιστότοπος του Android studio: <https://developer.android.com/studio>

2.3 • ΕΞΑΜΗΝΟ Γ΄

2.3.A • ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ (PL/SQL)

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Προγραμματισμός βάσεων δεδομένων (PL/SQL)» εισάγει τους/τις εκπαιδευόμενους/-ες στη λογική του υψηλού επιπέδου διαδικαστικού προγραμματισμού βάσεων δεδομένων, ενσωματώνοντας εντολές SQL. Η αλγοριθμική λογική της γλώσσας προγραμματισμού PL/SQL έχει επιρροές από τη γλώσσα προγραμματισμού C11, η οποία διδάσκεται στη μαθησιακή ενότητα «Γλώσσα προγραμματισμού Ι (C11)-Αλγοριθμική και δομές δεδομένων» του εξαμήνου Α΄, και από τη γλώσσα προγραμματισμού SQL, η οποία διδάσκεται στη μαθησιακή ενότητα «Συστήματα διαχείρισης βάσεων δεδομένων» του εξαμήνου Β΄ κατάρτισης. Συνεπώς, θα χρησιμοποιηθεί η πρότερη γνώση των δύο παραπάνω μαθησιακών ενότητων. Στην παρούσα μαθησιακή ενότητα, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με εντολές εισόδου και εξόδου, δομές βρόγχων και ελέγχων, εντολές SQL *Plus για τη διαχείριση των πινάκων της βάσης δεδομένων, εντολές διαχείρισης σφαλμάτων (exceptions), μεθόδους εξόδου/επικοινωνίας με φόρμες της Oracle, τύπους δεδομένων, τελεστές και ιεραρχίας τους, εντολές περιβάλλοντος και ενσωματωμένες συναρτήσεις βιβλιοθήκης, δομές επιλογής και επανάληψης, δρομείς (cursors) κ.ά. Επίσης, παρουσιάζονται οι βασικές εντολές της PL/SQL, όπως EXCEPTIONS, DECLARE, LOOP, FETCH, TRIGGER, PACKAGE, OPEN, CLOSE, BULK COLLECT, CREATE, SELECT (INTO), DROP, INSERT, UPDATE, DELETE, TRUNCATE, WHERE, LIKE, ORDER BY, GROUP BY, LIMIT, DISTINCT, ALTER, FROM, FUNCTIONS (MAX, MIN, SUM, COUNT, AVG κ.λπ.), regular expressions, INDEXES, JOINS, UNION και INTERSECT.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Εφαρμόζουν κατάλληλα ερωτήματα για την ανάκτηση ή/και τροποποίηση ή/και διαγραφή μεγάλου όγκου δεδομένων,
- Συσχετίζουν πίνακες βάσεων δεδομένων,
- Δημιουργούν κατάλληλες βάσεις δεδομένων με σκοπό τη χρήση τους σε συστήματα διαχείρισης βάσεων δεδομένων ή/και εφαρμογών,
- Συνεργάζονται με τα υπόλοιπα μέλη της ομάδας στον σχεδιασμό και στην υλοποίηση κατάλληλων βάσεων δεδομένων,
- Συνδυάζουν αλγοριθμική λογική, μαζί με εντολές διαχείρισης βάσεων δεδομένων,
- Υιοθετούν θετική στάση ως προς τις νέες τεχνολογίες ανάπτυξης βάσεων δεδομένων.

Βασικές έννοιες και λέξεις-κλειδιά

PL/SQL (Procedural Language/SQL) / SQL*Plus / Block / Cursor / Run time error / Exceptions / Procedures / Sequences / Trigger / Packages.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (4), Σύνολο (4).

Διάρθρωση των μαθησιακών ενότητων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΗΝ PL/SQL
2	ΔΙΑΧΕΙΡΙΣΗ ΠΙΝΑΚΩΝ
3	ΔΙΑΧΕΙΡΙΣΗ ΕΡΩΤΗΜΑΤΩΝ
4	ΔΙΑΔΙΚΑΣΤΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ
5	ΘΕΜΑΤΑ ΣΥΝΔΕΣΗΣ ΜΕ ΤΟ ΣΔΒΔ ΤΗΣ ORACLE
6	ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

Κείμενο αναφοράς

2.3.A.1 | ΕΙΣΑΓΩΓΗ ΣΤΗΝ PL/SQL

Αρχικά, παρουσιάζεται η PL/SQL (Procedural Language/SQL), η οποία είναι μια γλώσσα προγραμματισμού για την ανάπτυξη εφαρμογών που βασίζονται σε βάσεις δεδομένων. Γίνεται αναφορά στα πλεονεκτήματά της έναντι της SQL και τονίζεται ιδιαίτερα ότι επιτρέπει την ανάπτυξη διαδικαστικών προγραμμάτων. Ακολουθεί η παρουσίαση του SQL*Plus, ένα command line εργαλείο, το οποίο αλληλεπιδρά με βάσεις δεδομένων της Oracle και προτείνεται να χρησιμοποιηθεί στο πλαίσιο του μαθήματος.

Στη συνέχεια, παρουσιάζεται η βασική μονάδα της PL/SQL, το block και γίνεται αναφορά στις βασικές εντολές του: BEGIN, END. Παρουσιάζεται, επίσης, η εντολή RUN, η οποία εκτελεί ένα block, και ο τρόπος σύνταξής της. Γίνεται παρουσίαση του τρόπου ορισμού των μεταβλητών. Παρουσιάζεται η εντολή DECLARE, που προηγείται του ορισμού κάθε block, την οποία ακολουθούν οι ορισμοί των μεταβλητών. Οι βασικοί τύποι μεταβλητών της PL/SQL είναι: NUMBER, VARCHAR2, CHAR, DATE, BOOLEAN. Επίσης γίνεται αναφορά στις τιμές NULL, NOT NULL και τονίζεται ιδιαίτερα η έκφραση %TYPE. Παρουσιάζονται, στη συνέχεια, οι τελεστές της PL/SQL και συγκεκριμένα: οι μαθηματικοί (ενδεικτικά: +, -, *, /), σύγκρισης (ενδεικτικά: <=, >, >=, =, <>, BETWEEN, NOT BETWEEN, IN, IS NULL), λογικοί (ενδεικτικά: AND, OR, NOT), ανάθεσης (:=) και αλφαριθμητικοί (ενδεικτικά: ||, LIKE, NOT LIKE) τελεστές. Παρουσιάζονται, ακόμη, η ιεραρχία των τελεστών, δηλαδή ποια πράξη εκτελείται πρώτη σε μια ακολουθία πράξεων, και οι κανονικές εκφράσεις (*, ., ^, \$), η εντολή REGEXP, και η σύνταξη της REGEXP_LIKE μαζί με κανονικές εκφράσεις. Τέλος, γίνεται ανα-

φορά στις καθολικές μεταβλητές της Oracle (%FOUND, %NOTFOUND, %ISOPEN, %ROWCOUNT) και στον τρόπος χρήσης τους:

2.3.A.2 | ΔΙΑΧΕΙΡΙΣΗ ΠΙΝΑΚΩΝ

Η υποενότητα αυτή περιλαμβάνει εντολές και τεχνικές διαχείρισης των πινάκων και των δεδομένων τους. Σημειώνεται ότι στην PL/SQL δεν επιτρέπονται εντολές ορισμού δεδομένων (data definition statements), όπως οι CREATE TABLE, DROP TABLE και ALTER TABLE. Στις περισσότερες περιπτώσεις, η δημιουργία πινάκων γίνεται είτε χρησιμοποιώντας εντολές SQL απευθείας στο ΣΔΒΔ είτε χρησιμοποιώντας το γραφικό περιβάλλον διαχείρισης του ΣΔΒΔ. Στο πλαίσιο του μαθήματος, χρησιμοποιείται το εργαλείο SQL*Plus, το οποίο επιτρέπει τη διαχείριση πινάκων μέσω των εντολών: CREATE TABLE για τη δημιουργία πινάκων, ALTER TABLE για την αλλαγή της δομής του πίνακα και DROP TABLE για τη διαγραφή πίνακα. Ο κώδικας PL/SQL μπορεί να αλληλεπιδράσει με τους προϋπάρχοντες πίνακες. Γίνεται, επίσης, παρουσίαση του τρόπου με τον οποίο γίνεται να εκτελεστούν οι παραπάνω εντολές διαχείρισης πινάκων στην PL/SQL. Συγκεκριμένα, παρουσιάζεται η εντολή EXECUTE IMMEDIATE και ο τρόπος χρήσης της για την εκτέλεση των παραπάνω εντολών. Τονίζεται ότι η δημιουργία πινάκων από το PL/SQL χρησιμοποιείται συνήθως σε πιο προηγμένα σενάρια, όπου χρειάζεται να δημιουργηθούν δυναμικά πίνακες κατά τον χρόνο εκτέλεσης του προγράμματος, με βάση συγκεκριμένες συνθήκες ή απαιτήσεις. Στη συνέχεια, παρουσιάζονται οι παρακάτω εντολές διαχείρισης πινάκων και ο τρόπος σύνταξής τους: INSERT INTO για εισαγωγή εγγραφών σε πίνακα, DELETE FROM για διαγραφή εγγραφών από πίνακα (η οποία συνοδεύεται από τη WHERE ακολουθούμενη από μία συνθήκη), UPDATE για ενημέρωση εγγραφών πινάκων (συνοδεύεται από την SET, η οποία θέτει τιμές στα χαρακτηριστικά των εγγραφών και προαιρετικά από τη WHERE, ακολουθούμενη από μια συνθήκη) και την TRUNCATE για την απαλοιφή όλων των δεδομένων από έναν πίνακα. Σημειώνεται ότι οι παραπάνω εντολές συνοδεύονται από την εντολή COMMIT, η οποία εξασφαλίζει ότι οι αλλαγές θα αποθηκευτούν μόνιμα στη βάση δεδομένων.

2.3.A.3 | ΔΙΑΧΕΙΡΙΣΗ ΕΡΩΤΗΜΑΤΩΝ

Παρουσιάζεται η εντολή SELECT, η οποία χρησιμοποιείται για την ανάκτηση δεδομένων από έναν πίνακα. Τονίζεται ότι στην PL\SQL η SELECT επιστρέφει τα δεδομένα από μια γραμμή και όχι από πολλές. Παρουσιάζεται η σύνταξη της SELECT και οι ενδείξεις INTO, WHERE, DISTINCT. Στη συνέχεια, γίνεται παρουσίαση της έννοιας του δρομέα (cursor), μιας δομής που χρησιμοποιείται για τη διαχείριση των αποτελεσμάτων μιας ερώτησης. Τονίζεται ότι ο cursor δείχνει στα αποτελέσματα της ερώτησης και επιτρέπει σε ένα πρόγραμμα να επεξεργαστεί τα αποτελέσματα της ερώτησης. Τονίζεται επίσης το πλεονέκτημά του σε σχέση με τη SELECT, ότι δηλαδή επιτρέπει την πρόσβαση σε πολλαπλές γραμμές ενός πίνακα και όχι σε μια γραμμή

μόνο. Παρουσιάζονται, επίσης, οι εντολές διαχείρισης ενός cursor (CURSOR IS για τη δημιουργία cursor, OPEN για το άνοιγμα cursor, FETCH για τη μεταφορά των τιμών των αποτελεσμάτων του ερωτήματος σε μεταβλητές και η CLOSE για το κλείσιμο του cursor). Ακολουθούν οι εντολές για την παρουσίαση των αποτελεσμάτων ενός ερωτήματος. Αναφέρεται η ταξινόμηση των εγγραφών, η οποία γίνεται με χρήση της εντολής ORDER BY, ακολουθούμενη από τις ενδείξεις ASC (αύξουσα) ή DESC (φθίνουσα), και η ομαδοποίηση εγγραφών, η οποία γίνεται με χρήση της εντολής GROUP BY, ακολουθούμενη από την ένδειξη HAVING (ακολουθείται από μια συνθήκη). Σημειώνεται εδώ ότι οι παραπάνω εντολές χρησιμοποιούνται από τους cursors, οι οποίοι επιστρέφουν ένα σύνολο εγγραφών. Παρουσιάζονται, επίσης, οι συναρτήσεις επεξεργασίας ενός συνόλου δεδομένων και ενδεικτικά αναφέρονται οι: MAX (μέγιστο), MIN (ελάχιστο), SUM (άθροισμα), COUNT (μετρητής), AVG (μέση τιμή). Τέλος, παρουσιάζονται οι εντολές διαχείρισης αποτελεσμάτων ερωτημάτων, και συγκεκριμένα: η UNION για την ένωση αποτελεσμάτων χωρίς διπλοεγγραφές, η INTERSECT για την τομή αποτελεσμάτων (κοινά αποτελέσματα μόνο) και η JOIN για την ένωση αποτελεσμάτων βάσει κοινού πεδίου.

2.3.A.4 | ΔΙΑΔΙΚΑΣΤΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

Παρουσιάζεται η έννοια των δομών ελέγχου για τον έλεγχο της ροής του προγράμματος και οι εντολές για την υλοποίηση δομών ελέγχου IF, THEN, ELSE, ELSIF, END IF, CASE, και η εντολή EXIT. Παρουσιάζεται επίσης, μέσα από παραδείγματα, η δυνατότητα που δίνει η PL/SQL να συνδεθούν πολλές συνθήκες μεταξύ τους, κάνοντας χρήση λογικών τελεστών. Επιπροσθέτως, παρουσιάζεται η έννοια των δομών επανάληψης, του βρόγχου επανάληψης και οι εντολές της υλοποίησης δομών επανάληψης. Συγκεκριμένα, παρουσιάζονται οι LOOP, END LOOP για τον ορισμό ενός βρόγχου επανάληψης, οι EXIT, EXIT WHEN για την έξοδο από τον βρόγχο, η WHILE για την εκτέλεση ενός βρόγχου, όσο ισχύει μια συνθήκη, και η FOR για την εκτέλεση ενός βρόγχου, όσο μια μεταβλητή βρίσκεται σε ένα προκαθορισμένο εύρος τιμών (IN). Στη συνέχεια, παρουσιάζονται οι διαδικασίες (procedures). Η διαδικασία είναι μια ρουτίνα, η οποία αποθηκεύεται και στη συνέχεια μπορεί να κληθεί. Παρουσιάζονται οι εντολές για την υλοποίηση procedures (CREATE PROCEDURE). Επίσης, γίνεται αναφορά στα ορίσματα της διαδικασίας και εξηγούνται οι τύποι των ορισμάτων (modes): IN, OUT, INOUT. Γίνεται αναφορά στον τρόπο κλήσης της διαδικασίας και παρουσιάζεται η έννοια της ακολουθίας και η χρήση της στη δημιουργία πρωτεύοντων κλειδιών (primary keys). Παρουσιάζονται, ακόμη, η έννοια του τεχνητού κλειδιού (surrogate key) και οι εντολές για τη δημιουργία μιας ακολουθίας, όπως: η CREATE SEQUENCE για τη δημιουργία ακολουθίας, η START WITH για τον ορισμό της αρχικής τιμής της ακολουθίας και η INCREMENT BY για τον ορισμό του βήματος της ακολουθίας.

2.3.A.5 | ΘΕΜΑΤΑ ΣΥΝΔΕΣΗΣ ΜΕ ΤΟ ΣΔΒΔ ΤΗΣ ORACLE

Η PL/SQL διαθέτει ενσωματωμένες συναρτήσεις από έτοιμες βιβλιοθήκες, οι οποίες μπορούν να χρησιμοποιηθούν κατά την ανάπτυξη προγραμμάτων. Παρουσιάζονται οι σημαντικότερες από αυτές: μαθηματικές συναρτήσεις (abs, sqrt, mod, power, cos, sin, tan, log, ln), συναρτήσεις διαχείρισης αλφαριθμητικών (COMPOSE, LENGTH, UPPER, TO_CHAR, TO_NUMBER) και συναρτήσεις ημερομηνίας (CURRENT_DATE, CURRENT_TIME, MONTHS_BETWEEN. Ακολουθούν οι εντολές εισόδου/εξόδου. Παρουσιάζεται η βασική εντολή εξόδου στην PL/SQL, η DBMS_OUTPUT.PUT_LINE και γίνεται αναφορά στον τρόπο που εμφανίζονται συμβολοσειρές, αριθμοί και συνενωμένα δεδομένα. Στη συνέχεια, τονίζεται ότι η PL/SQL δεν διαθέτει εντολές ή συναρτήσεις για είσοδο δεδομένων από το πληκτρολόγιο. Αυτό γίνεται γιατί στην Oracle η είσοδος δεδομένων γίνεται συνήθως από φόρμες. Παρουσιάζεται εδώ η τεχνική εισαγωγής δεδομένων με υποκατάστατες μεταβλητές και τονίζεται ότι οι μεταβλητές αυτές είναι χαρακτηριστικό της SQL*Plus. Παρουσιάζονται, επίσης, η επικοινωνία της PL/SQL με φόρμες της Oracle, η οποία γίνεται μέσω του Oracle Forms PL/SQL API, και οι βασικές μέθοδοι επικοινωνίας που παρέχει το API: CALL_FORM/EXIT_FORM, OPEN_FORM/CLOSE_FORM, GET_ITEM_PROPERTY/SET_ITEM_PROPERTY, NEXT_ITEM/PREVIOUS_ITEM. Τέλος, εξηγείται ότι στην PL/SQL, ως εντολές και μεταβλητές περιβάλλοντος, εννοούνται οι ειδικές εντολές και μεταβλητές του ΣΔΒΔ που χρησιμοποιείται (Oracle), π.χ. η DBMS_OUTPUT.PUT_LINE() και οι μεταβλητές SQLCODE και SQLERRM.

2.3.A.6 | ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

Σε αυτή την υποενότητα, παρουσιάζεται η έννοια των run time errors (σφάλματα χρόνου εκτέλεσης), η έννοια της εξαίρεσης (exception), μια ρουτίνα διαχείρισης ενός σφάλματος, και η σημασία της για την εκτέλεση ενός προγράμματος. Γίνεται αναφορά στις κύριες εξαιρέσεις της Oracle (no_data_found, too_many_rows, zero_divide) και στις μεταβλητές SQLCODE και SQLERRM. Στη συνέχεια, παρουσιάζεται το τμήμα EXCEPTION του βασικού block, όπου γράφεται ο κώδικας διαχείρισης των εξαιρέσεων, και δίνεται παράδειγμα κώδικα για τον χειρισμό μιας εξαίρεσης. Εκτός αυτών, παρουσιάζεται ο τρόπος υλοποίησης της δήλωσης μιας εξαίρεσης από τον/τη χρήστη/-τρια. Συγκεκριμένα, παρουσιάζεται η δήλωση της εξαίρεσης στην περιοχή DECLARE, η εντολή RAISE που ενεργοποιεί την εξαίρεση και ο χειρισμός της εξαίρεσης στην περιοχή EXCEPTION. Το επόμενο θέμα της υποενότητας είναι τα εναύσματα (triggers). Το trigger είναι ένας μηχανισμός που πυροδοτεί ενέργειες, όταν συμβαίνουν γεγονότα (όπως INSERT, UPDATE, DELETE) σε έναν πίνακα. Παρουσιάζονται παραδείγματα κώδικα SQL για την υλοποίηση triggers (CREATE TRIGGER) και τονίζεται η σημασία τους. Ακολουθεί η παρουσίαση της έννοιας των ευρετηρίων (indicies) και τονίζεται η χρησιμότητά τους, ιδιαίτερα σε βάσεις με πολλές εγγραφές. Τονίζεται ότι η PL/SQL δεν επιτρέπει τη δημιουργία ευρετηρίων, γίνεται όμως

παρουσίαση του τρόπου με τον οποίον μπορεί να δημιουργήσει κανείς στην PL/SQL. Συγκεκριμένα, παρουσιάζεται η εντολή EXECUTE IMMEDIATE και ο τρόπος χρήσης της για την εκτέλεση των εντολών δημιουργίας ευρετηρίων (CREATE INDEX). Επιπροσθέτως, παρουσιάζεται η έννοια του πακέτου (package), μια δομή δεδομένων που ομαδοποιεί τα αντικείμενα (π.χ. διαδικασίες, μεταβλητές, δρομείς, εξαιρέσεις) και τα πλεονεκτήματα χρήσης του. Παρουσιάζονται οι εντολές CREATE PACKAGE και CREATE PACKAGE BODY, καθώς και ο τρόπος με τον οποίο καλούνται τα αντικείμενα του package σε ένα πρόγραμμα. Τέλος, παρουσιάζεται η έννοια της μαζικής εισαγωγής δεδομένων και τα πλεονεκτήματά της και η εντολή BULK COLLECT, για την οποία δίνεται παράδειγμα χρήσης της σε συνδυασμό με την εντολή SELECT INTO.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Κεχρής, Ε. (2021). *Σχεσιακές Βάσεις Δεδομένων*. Αθήνα: Εκδόσεις Κριτική.
2. Hoffer, J., Ramesh, V. & Topi, H. (2017). *Βάσεις Δεδομένων: σύγχρονη διαχείριση*. Θεσσαλονίκη: Εκδόσεις Τζιόλα.

Συμπληρωματικές

1. Ο επίσημος ιστότοπος της Oracle για την PL\SQLQ:
<https://www.oracle.com/database/technologies/appdev/plsql.html>
2. Ο επίσημος ιστότοπος της Oracle για την SQL*Plus:
<https://www.oracle.com/database/technologies/appdev/plsql.html>
3. Το επίσημο κανάλι της Oracle στο YouTube:
<https://www.youtube.com/@Oracle>

2.3.B • ΓΛΩΣΣΑ SERVER-SIDE ΔΙΑΔΙΚΤΥΑΚΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ (PHP)

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Γλώσσα server-side διαδικτυακού προγραμματισμού (PHP)» εισάγει τους/τις εκπαιδευόμενους/-ες στη γλώσσα διαδικτυακού προγραμματισμού PHP, ώστε να αποκτήσουν τις βάσεις για τη δημιουργία server-side δυναμικού περιεχομένου, κατάλληλου για χρήση στον Παγκόσμιο Ιστό. Η σύνταξη της γλώσσας αυτής είναι επηρεασμένη από τη γλώσσα προγραμματισμού C11, η οποία διδάσκεται στη μαθησιακή ενότητα «Γλώσσα προγραμματισμού Ι (C11)-Αλγοριθμική και δομές δεδομένων» του εξαμήνου Α΄. Επίσης, θα χρησιμοποιηθεί η πρότερη γνώση των μαθησιακών ενοτήτων «Εργαλεία ανάπτυξης εφαρμογών Διαδικτύου», «Γλώσσα client-side διαδικτυακού προγραμματισμού (JavaScript)» (εξαμήνου Α΄) και «Συστήματα διαχείρισης βάσεων δεδομένων» (εξαμήνου Β΄). Ειδικότερα, παρουσιάζεται το

συντακτικό της γλώσσας PHP και βασικές έννοιες, όπως: σύνδεση με βάσεις δεδομένων, βιβλιοθήκη PHP, εγκατάσταση διαδικτυακού διακομιστή, υπηρεσίες φιλοξενίας, FTP και έννοιες αντικειμενοστρεφούς προγραμματισμού (κλάσεις, αντικείμενα, μέθοδος δημιουργός/καταστροφάς, κληρονομικότητα και διεπαφές). Επίσης, αναφέρονται προχωρημένα θέματα διαδικτυακού προγραμματισμού (διαχείριση αρχείων, sessions, cookies, filters, exceptions, PHP mail, JSON, AJAX, XML, δημοφιλή PHP frameworks), θέματα ασφάλειας διαδικτυακών εφαρμογών (OWASP top ten) και τρόποι προστασίας ενάντια σε απειλές. Τέλος, σχεδιάζονται και αναπτύσσονται δυναμικοί και ασφαλείς ιστότοποι με σύνδεση με βάσεις δεδομένων.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναγνωρίζουν τα πλεονεκτήματα των server-side γλωσσών προγραμματισμού,
- Επιλέγουν τα κατάλληλα frameworks που θα χρησιμοποιήσουν για την ανάπτυξη των ιστοτόπων,
- Αναγνωρίζουν τη σημασία του δομημένου και τεκμηριωμένου κώδικα,
- Αναπτύσσουν δυναμικούς ιστότοπους, με τη χρήση των γλωσσών PHP, HTML5 και βάσεων δεδομένων,
- Συνεισφέρουν με πρωτότυπο κώδικα στην κοινότητα των προγραμματιστών ιστοτόπων,
- Προσαρμόζουν τον κώδικα και τη δομή των βάσεων δεδομένων, ανάλογα με τις προδιαγραφές των ιστοτόπων που αναπτύσσουν,
- Ενσωματώνουν βάσεις δεδομένων σε ιστότοπους,
- Συνεργάζονται με άλλους/-ες εκπαιδευόμενους/-ες, συγκροτώντας ομάδες, με σκοπό τη δημιουργία και την επίλυση προβλημάτων των ιστοτόπων,
- Υιοθετούν θετική στάση ως προς τις νέες τεχνολογίες ανάπτυξης ιστοτόπων.

Βασικές έννοιες και λέξεις-κλειδιά

Backend Development / Server-side Development / Open Source / Web Development / Database Management / Form Handling / File Handling / Cookies/Sessions.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (4), Σύνολο (4).

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΗΝ PHP-ΣΥΝΤΑΚΤΙΚΟ ΚΑΙ ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΩΝ
2	ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΤΕΛΕΣΤΕΣ ΣΤΗΝ PHP-ΔΟΜΕΣ ΕΛΕΓΧΟΥ ΡΟΗΣ ΠΡΟΓΡΑΜΜΑΤΟΣ-ΣΥΝΑΡΤΗΣΕΙΣ-ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΠΙΝΑΚΕΣ ΣΤΗΝ PHP
3	PHP FORMS-ΔΙΑΧΕΙΡΙΣΗ ΑΡΧΕΙΩΝ ΣΤΗΝ PHP-ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΕΙΑ ΣΤΗΝ PHP
4	ΕΠΙΚΟΙΝΩΝΙΑ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ΣΤΗΝ PHP
5	PHP ΚΑΙ COOKIES/SESSIONS
6	ΠΡΟΗΓΜΕΝΕΣ ΤΕΧΝΙΚΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΣΤΗΝ PHP

Κείμενο αναφοράς

2.3.B.1 | ΕΙΣΑΓΩΓΗ ΣΤΗΝ PHP-ΣΥΝΤΑΚΤΙΚΟ ΚΑΙ ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΩΝ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν την PHP ως μια γλώσσα προγραμματισμού που χρησιμοποιείται κυρίως για την ανάπτυξη δυναμικών ιστοσελίδων και εφαρμογών. Αρχικά, θα πραγματοποιηθεί αναφορά στο όνομά της (PHP: “PHP Hypertext Preprocessor”), το οποίο σημαίνει ότι ο κώδικας PHP εκτελείται πριν από την αποστολή της ιστοσελίδας στον/στην πελάτη/-ισσα. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν ότι η PHP είναι μια ανοιχτού κώδικα, διασυνδεδεμένη, αντικειμενοστρεφής γλώσσα προγραμματισμού, η οποία εκτελείται στον διακομιστή (server) και υποστηρίζει διάφορες πλατφόρμες βάσεων δεδομένων και τεχνολογιών. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τα βασικά στοιχεία και τις δυνατότητες της PHP και πώς να δημιουργήσουν απλές και σύνθετες ιστοσελίδες και εφαρμογές με τη χρήση της. Γι’ αυτό, θα χρησιμοποιηθούν εργαλεία, όπως τα ΧΑΜΡΡ/ΜΑΜΡ/ΛΑΜΡ/ΨΑΜΡ και όλα τα services που ενυπάρχουν σε αυτά τα εργαλεία. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν τις ομοιότητες του συντακτικού της PHP με αυτό των γλωσσών C11, Java9 και Perl (με κάποιες διαφορές). Επιπλέον, θα διδαχθούν τον τρόπο ενσωμάτωσης κώδικα PHP σε ένα αρχείο HTML (πρέπει να ξεκινάει και να τελειώνει με τα ειδικά σύμβολα <?php και ?>) και η κατάληξη του αρχείου κώδικα συνήθως να είναι .php. Αναφορικά με τις μεταβλητές στην PHP, οι εκπαιδευόμενοι/-ες θα διδαχθούν ότι ξεκινούν με το σύμβολο \$ και ακολουθούνται από ένα όνομα που μπορεί να περιέχει γράμματα, αριθμούς και κάτω παύλες, όπως και σε όλες τις υπόλοιπες γλώσσες προγραμματισμού. Οι μεταβλητές θα διδαχθούν στους/στις εκπαιδευόμενους/-ες ως loosely typed, το οποίο λειτουργικά ερμηνεύεται στο ότι η δήλωση του τύπου τους εκ των προτέρων είναι προαιρετική.

2.3.B.2 | ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΤΕΛΕΣΤΕΣ ΣΤΗΝ PHP-ΔΟΜΕΣ ΕΛΕΓΧΟΥ ΡΟΗΣ ΠΡΟΓΡΑΜΜΑΤΟΣ-ΣΥΝΑΡΤΗΣΕΙΣ-ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΠΙΝΑΚΕΣ ΣΤΗΝ PHP

Σε αυτή τη μαθησιακή υποεπένδυση, οι εκπαιδευόμενοι/-ες θα διδαχθούν ότι η PHP υποστηρίζει διάφορους τύπους δεδομένων, όπως αριθμούς (ακέραιους, δεκαδικούς, εκθετικούς), συμβολοσειρές (ακολουθίες χαρακτήρων που περικλείονται σε μονά ή διπλά εισαγωγικά), λογικές τιμές (true/false), πίνακες (αριθμητικούς, συσχετιστικούς ή πολυδιάστατους), αντικείμενα (στιγμιότυπα κλάσεων με ιδιότητες και μεθόδους), πόρους (ειδικούς τύπους δεδομένων που αντιπροσωπεύουν εξωτερικές συνδέσεις, όπως βάσεις δεδομένων, αρχεία ή εικόνες) και null (ειδική τιμή που δηλώνει την έλλειψη ή τη μη ορισμένη τιμή). Επιπλέον, η PHP υποστηρίζει αριθμητικούς, συγκριτικούς, λογικούς, συνενωτικούς και εκχωρητικούς τελεστές. Οι εκπαιδευόμενοι/-ες θα μάθουν τα προαναφερθέντα στην πράξη, καθώς και την αναγνώριση και την επίλυση συνηθισμένων σφαλμάτων που μπορεί να προκύψουν κατά τη χρήση τους. Οι εκπαιδευόμενοι/-ες θα διδαχθούν επίσης την ανάγνωση και την κατανόηση των σφαλμάτων που παράγονται από τον διερμηνέα της PHP, και την εφαρμογή των καλύτερων πρακτικών για την αποφυγή των σφαλμάτων αυτών. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν ότι η PHP υποστηρίζει δομές ελέγχου ροής προγράμματος, όπως if, else, elseif, switch, while, do-while, for, foreach, continue και break, καθώς επίσης και συναρτήσεις (χρήση της function) και ενσωματωμένες συναρτήσεις της PHP (συναρτήσεις για συμβολοσειρές, πίνακες, μαθηματικά, ημερομηνίες, αρχεία κ.λπ.). Επίσης, οι εκπαιδευόμενοι/-ες θα διδαχθούν ότι η PHP παρέχει πολλές ενσωματωμένες συναρτήσεις για τη διαχείριση, την αναζήτηση, την ταξινόμηση και την επεξεργασία πινάκων. Τέλος, οι εκπαιδευόμενοι/-ες θα διδαχθούν τις δύο βασικές δομές δεδομένων της PHP: τους πίνακες και τα αντικείμενα, με έμφαση στους συσχετιστικούς πίνακες, οι οποίοι έχουν ονομαστικά κλειδιά που μπορούν να είναι συμβολοσειρές ή αριθμοί.

2.3.B.3 | PHP FORMS-ΔΙΑΧΕΙΡΙΣΗ ΑΡΧΕΙΩΝ-ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΕΙΑ ΣΤΗΝ PHP

Οι εκπαιδευόμενοι/-ες θα διδαχθούν PHP Forms ως μέσο επικοινωνίας της PHP με τον/την τελικό/-ή χρήστη/-τρια, λαμβάνοντας δεδομένα από τον/την πελάτη/-ισσα και κάνοντας την επεξεργασία τους στον διακομιστή. Τα PHP Forms χρησιμοποιούν HTML Forms, στοιχεία της HTML που επιτρέπουν στον/στη χρήστη/-τρια να εισαγάγει δεδομένα (κείμενο, κουμπιά, κουτιά επιλογής, λίστες κ.λπ.), χρησιμοποιώντας δύο μεθόδους για να στείλουν τα δεδομένα από τον/την πελάτη/-ισσα στον διακομιστή: την GET και την POST (ανάλογα με τα προς αποστολή δεδομένα). Τέλος, η PHP παρέχει τις δηλώσεις \$_GET και \$_POST για να προσπελάσει τα δεδομένα που στέλνονται από τις PHP Forms. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν ότι η PHP υποστηρίζει τη διαχείριση αρχείων, δηλαδή τη δυνατότητα να διαβάζουμε, να γράφουμε, να δημιουργούμε, να διαγράφουμε και να τροποποιούμε αρχεία στον διακομιστή. Οι εκπαιδευόμενοι/-ες θα αλληλεπιδράσουν με τις ενσωματωμένες συναρτήσεις για τη διαχείριση αρχείων της PHP, όπως: fopen, fclose, fread, fwrite, file_exists, copy, rename,

unlink κ.ά. Εκτός από τη διαδικασία ανοίγματος ενός αρχείου (μέσω της εντολής fopen), είναι απαραίτητος και ο καθορισμός του τρόπου με τον οποίο θα ανοιχτεί το αρχείο, όπως r για ανάγνωση, w για εγγραφή, a για επέκταση κ.λπ. (ακριβώς όπως και στη γλώσσα προγραμματισμού C11). Επίσης, οι εκπαιδευόμενοι/-ες θα διδαχθούν ότι, πριν από την κλήση της fclose, θα πρέπει να πραγματοποιείται έλεγχος της επιστρεφόμενης τιμής της fopen, ούτως ώστε να γνωρίζουμε αν πραγματοποιήθηκε η ζητούμενη διαδικασία. Τέλος, οι εκπαιδευόμενοι/-ες θα διδαχθούν την αντικειμενοστρέφεια που υποστηρίζει η PHP, έχοντας γνωστές λειτουργίες, όπως κλάσεις, αντικείμενα, μεθόδους, πεδία, κληρονομικότητα, ενθυλάκωση, πολυμορφισμό και αφαίρεση (απλοποίηση ενός αντικειμένου, αγνοώντας περιττές/πολύπλοκες λεπτομέρειες).

2.3.B.4 | ΕΠΙΚΟΙΝΩΝΙΑ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ΣΤΗΝ PHP

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν ότι η επικοινωνία βάσης δεδομένων στην PHP είναι η δυνατότητα να συνδεόμαστε, να θέτουμε ερωτήματα, να ανακτούμε και να ενημερώνουμε δεδομένα από μια βάση δεδομένων, που είναι ένα σύστημα αποθήκευσης και διαχείρισης δεδομένων, όπως η MySQL, η Oracle, η PostgreSQL και άλλες τεχνολογίες συστημάτων διαχείρισης βάσεων δεδομένων. Πιο συγκεκριμένα, οι εκπαιδευόμενοι/-ες θα διδαχθούν ότι η PHP χρησιμοποιεί διάφορες επεκτάσεις και αφαιρετικά επίπεδα για να επικοινωνεί με τις βάσεις δεδομένων, όπως την εντολή mysqli, την PDO (PHP Data Objects), την ODBC (Open Data Base Connectivity) κ.λπ., καθώς επίσης και τα στοιχεία σύνδεσης που πρέπει να καθοριστούν για να εγκαθιδρυθεί σύνδεση με μια βάση δεδομένων, όπως το όνομα του διακομιστή, το όνομα της βάσης δεδομένων, το όνομα χρήστη/-τριας και ο κωδικός πρόσβασής του. Για να εκτελέσουν ερωτήματα σε μια βάση δεδομένων, οι εκπαιδευόμενοι/-ες θα πρέπει να χρησιμοποιήσουν μια γλώσσα ερωτημάτων, όπως η SQL, η οποία επιτρέπει την επιλογή, την εισαγωγή, την ενημέρωση και τη διαγραφή δεδομένων από τους πίνακες της βάσης δεδομένων. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν τον τρόπο ανάκτησης των αποτελεσμάτων ενός ερωτήματος, μέσω μιας μεταβλητής που αποθηκεύει το σύνολο των εγγραφών που επιστρέφονται από το ερώτημα, και η οποία θα διατρέχεται από ένα βρόχο, όπως while ή foreach. Μέσω της PHP, οι εκπαιδευόμενοι/-ες θα μπορούν να εκτελέσουν κάθε CRUD (Create, Read, Update, Delete) λειτουργία σε μια βάση δεδομένων.

2.3.B.5 | PHP ΚΑΙ COOKIES/SESSIONS

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με την έννοια των cookies και των sessions, δύο βασικών εργαλείων που χρησιμοποιεί η PHP για τη διαχείριση της κατάστασης και της ταυτότητας του/της χρήστη/-τριας. Τα cookies είναι μικρά αρχεία κειμένου, που αποθηκεύονται στον/στην πελάτη/-ισσα και περιέχουν πληροφορίες για τον/τη χρήστη/-τρια, όπως το όνομα, το e-mail, οι προτιμήσεις πλοήγησης και άλλα δεδομένα. Η PHP μπορεί να δημιουργήσει, να δια-

βάσει, να τροποποιήσει και να διαγράψει cookies, χρησιμοποιώντας τις συναρτήσεις `setcookie`, `$_COOKIE`, `setrawcookie` και `unset`. Από την άλλη πλευρά, οι εκπαιδευόμενοι/-ες θα διδαχθούν την ύπαρξη των sessions, ως μηχανισμών που επιτρέπουν στην PHP να αποθηκεύει προσωρινά δεδομένα στον διακομιστή για κάθε χρήστη/-τρια που επισκέπτεται μια ιστοσελίδα. Τα sessions χρησιμοποιούν ένα μοναδικό αναγνωριστικό, γνωστό ως session ID, για να συσχετίζουν τα δεδομένα του/της χρήστη/-τρι-ας με τον/την πελάτη/-ισσα. Η PHP μπορεί να ξεκινήσει, να διαβάσει, να τροποποιήσει και να τερματίσει sessions, χρησιμοποιώντας τις συναρτήσεις `session_start`, `$_SESSION`, `session_regenerate_id` και `session_destroy`. Συνοψίζοντας, η εκπαιδευτική αυτή ενότητα έχει στόχο να παράσχει στους/στις εκπαιδευόμενους/-ες μια ολοκληρωμένη εικόνα για τον τρόπο με τον οποίο η PHP διαχειρίζεται την κατάσταση και την ταυτότητα του χρήστη, μέσω των cookies και των sessions. Με αυτόν τον τρόπο, οι εκπαιδευόμενοι/-ες θα είναι σε θέση να κατανοήσουν καλύτερα τη λειτουργία των ιστοσελίδων και να αναπτύξουν πιο αποτελεσματικές και ασφαλείς εφαρμογές.

2.3.B.6 | ΠΡΟΗΓΜΕΝΕΣ ΤΕΧΝΙΚΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΣΤΗΝ PHP

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν ότι η PHP είναι μια πλούσια και ευέλικτη γλώσσα προγραμματισμού, η οποία παρέχει πολλές προηγμένες τεχνικές και δυνατότητες (αναφορά, αναδρομή, υπερφόρτωση, ανακατεύθυνση, δυνατότητα εφαρμογής φίλτρων, εξαιρέσεις, μαγικές μέθοδοι και χώροι ονομάτων). Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα διδαχθούν την αναφορά ως μια δυνατότητα να δημιουργήσουν μια μεταβλητή, που δείχνει στη διεύθυνση μιας άλλης μεταβλητής, χρησιμοποιώντας το σύμβολο `&`, ακριβώς όπως και στη γλώσσα προγραμματισμού C11. Επίσης, όπως και στις υπόλοιπες γλώσσες προγραμματισμού, οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους χρήσης αναδρομικών συναρτήσεων, χρησιμοποιώντας το όνομα σε επανακλήσεις ή χρησιμοποιώντας τη δεσμευμένη λέξη-κλειδί `self`. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν την υπερφόρτωση μεθόδων ως λειτουργία της κληρονομικότητας, τις εξαιρέσεις, χρησιμοποιώντας τις λέξεις-κλειδιά `try`, `catch` και `throw`, και την ανακατεύθυνση, δηλαδή τη δυνατότητα να μεταβαίνει ο/η τελικός/-ή χρήστης/-τρια σε μια άλλη ιστοσελίδα, χρησιμοποιώντας τη συνάρτηση `header`. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τη λειτουργία φίλτρων στην PHP ως τη δυνατότητα να ελέγξουν, να επικυρώσουν και να εκκαθαρίσουν δεδομένα που λαμβάνουν, χρησιμοποιώντας τις συναρτήσεις `filter_var`, `filter_input` και `filter_input_array`. Επιπρόσθετα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τις μαγικές μεθόδους, οι οποίες έχουν ειδικά ονόματα που ξεκινούν με δύο κάτω παύλες και εκτελούνται αυτόματα όταν συμβεί κάποιο γεγονός, όπως η δημιουργία, η κλήση, η πρόσβαση ή η καταστροφή ενός αντικειμένου. Τέλος, οι εκπαιδευόμενοι/-ες θα χρησιμοποιήσουν χώρους ονομάτων, στους οποίους η PHP οργανώνει και ομαδοποιεί τα ονόματα των κλάσεων, των συναρτήσεων και των σταθερών, χρησιμοποιώντας τη λέξη-κλειδί `namespace`, με αντίστοιχη λειτουργία όπως στη C++14.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Sarabia, R. & Hansen, B. (2023). *Test-Driven Development with PHP 8: Build extensible, reliable, and maintainable enterprise-level applications using TDD and BDD with PHP*. Birmingham Packt Publishing.
2. Ullman, L. (2009). *Εισαγωγή στις PHP 6 & MYSQL 5 με εικόνες*. Αθήνα: Εκδόσεις Κλειδάριθμος.

Συμπληρωματικές

1. Duckett, J. (2023). *Front-End Back-End Development with HTML, CSS, JavaScript, jQuery, PHP, and MySQL*. New Jersey: Wiley & Sons.
2. Mann, A. E. (2023). *PHP Cookbook*. California: O'Reilly.

2.3.Γ • ΕΞΕΙΔΙΚΕΥΜΕΝΑ ΖΗΤΗΜΑΤΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΚΑΙ ΕΦΑΡΜΟΓΩΝ ΔΙΑΔΙΚΤΥΟΥ ΜΕ PYTHON

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Εξειδικευμένα ζητήματα προγραμματισμού και εφαρμογών Διαδικτύου με Python» εισάγει τους/τις εκπαιδευόμενους/-ες στη λογική του αντικειμενοστρεφούς προγραμματισμού και με την επικοινωνία ενός ιστοτόπου με μια τοπική εφαρμογή με τη γλώσσα Python. Η παρούσα μαθησιακή ενότητα έχει συνάφειες με τη μαθησιακή ενότητα «Εισαγωγή στη γλώσσα προγραμματισμού Python» του εξαμήνου Β΄ σπουδών και, ουσιαστικά, αποτελεί συνέχειά της. Εδώ, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με προχωρημένα θέματα διαδικαστικού και αντικειμενοστρεφούς προγραμματισμού, όπως αυτά των εξαιρέσεων της υπερφόρτωσης, της δημιουργίας κλάσεων και αντικειμένων, της χρήσης μεθόδων κληρονομικότητας και πολυμορφισμού, μέσω των ασφαλών τεχνικών ενθυλάκωσης των πεδίων κάθε αντικειμένου, δημιουργώντας, έτσι, πιο ανθεκτικές και πιο ασφαλείς εφαρμογές κονσόλας. Μέσα από τα παραπάνω, ενισχύονται οι βασικές γνώσεις και δεξιότητες για τον προγραμματισμό πολύπλοκων προγραμμάτων σε Python, ώστε να υλοποιηθούν εφαρμογές γραφικού περιβάλλοντος σε Python και εφαρμογές που θα αλληλεπιδρούν με το Διαδίκτυο.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναπτύσσουν αντικειμενοστρεφείς εφαρμογές με χρήση της γλώσσας προγραμματισμού Python,

- Συνεισφέρουν με πρωτότυπο κώδικα στην κοινότητα των προγραμματιστών εφαρμογών,
- Τροποποιούν υπάρχοντα κώδικα, ανάλογα με τις προδιαγραφές των εφαρμογών που αναπτύσσουν,
- Αποφασίζουν για τη χρήση ή μη κατάλληλων βιβλιοθηκών, σχετικών με την ανάπτυξη εφαρμογών,
- Συνεργάζονται με άλλους/-ες εκπαιδευόμενους/-ες, συγκροτώντας ομάδες, με σκοπό τη δημιουργία και την επίλυση προβλημάτων των εφαρμογών,
- Υιοθετούν θετική στάση ως προς τις υπάρχουσες τεχνολογίες ανάπτυξης εφαρμογών,
- Αναγνωρίζουν τις βασικές προγραμματιστικές αρχές του αντικειμενοστρεφούς προγραμματισμού, βάσει των οποίων έχει υλοποιηθεί μια εφαρμογή Python,
- Επιλέγουν τα κατάλληλα μέλη κλάσεων (πεδία, μεθόδους κ.ά.) που θα χρησιμοποιήσουν στις εφαρμογές που θα αναπτύσσουν,
- Αναγνωρίζουν τη σημασία του πολυμορφισμού και της κληρονομικότητας.

Βασικές έννοιες και λέξεις-κλειδιά

Αντικειμενοστρεφής σχεδίαση / Κλάση / Αντικείμενο / Ιδιότητες / Μέθοδοι / Κληρονομικότητα / Πολυμορφισμός / Σύνδεση Python με βάσεις δεδομένων / HTTP αιτήματα / Προγραμματιστικά πλαίσια / Flask / Django.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (4), Σύνολο (4).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ ΜΕ PYTHON
2	ΠΡΟΧΩΡΗΜΕΝΕΣ ΕΝΝΟΙΕΣ ΣΤΟΝ ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ
3	ΧΡΗΣΗ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ ΣΤΗΝ PYTHON
4	ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΩΝ ΕΦΑΡΜΟΓΩΝ ΜΕ PYTHON
5	ΧΡΗΣΗ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΩΝ ΠΛΑΙΣΙΩΝ ΓΙΑ ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΩΝ ΕΦΑΡΜΟΓΩΝ
6	ΧΡΗΣΗ ΕΞΕΙΔΙΚΕΥΜΕΝΩΝ ΒΙΒΛΙΟΘΗΚΩΝ

Κείμενο αναφοράς

2.3.Γ.1 | ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ ΜΕ PYTHON

Ο «Αντικειμενοστρεφής προγραμματισμός (Object Oriented Programming)» αποτελεί μια από τις πιο βασικές και ισχυρές έννοιες στον σύγχρονο προγραμματισμό. Η εν προκειμένω ενότητα, λοιπόν, αναπτύσσει την ικανότητα των καταρτιζομένων να

οργανώνουν και να διαχειρίζονται τον κώδικά τους με έναν δομημένο και αποδοτικό τρόπο, χρησιμοποιώντας αντικείμενα και κλάσεις. Στην υποενότητα αυτή, οι εκπαιδευόμενοι/-ες κατανοούν τις βασικές αρχές της αντικειμενοστρεφούς προσέγγισης, προσπαθώντας να αντιληφθούν τον κόσμο ως ένα σύνολο αντικειμένων που αλληλεπιδρούν μεταξύ τους. Αρχικά, οι εκπαιδευόμενοι/-ες εισάγονται στην έννοια της αντικειμενοστρεφούς σχεδίασης. Κατανοούν πώς γίνεται η αντικειμενοστρεφής μοντελοποίηση και διαχωρίζουν τις έννοιες: κλάση, αντικείμενο, ιδιότητα και μέθοδος. Χρησιμοποιούν τα βασικά διαγράμματα της UML, για να αναπαραστήσουν κλάσεις και αντικείμενα και, ξεκινώντας από ένα πρόβλημα, δημιουργούν αρχικά τις κλάσεις ανάλυσης, που δεν περιέχουν ιδιότητες, και φθάνουν στις κλάσεις σχεδίασης, που περιέχουν ιδιότητες και μεθόδους.

Στη συνέχεια, διδάσκεται η σύνταξη των κλάσεων στην Python και οι εκπαιδευόμενοι/-ες μαθαίνουν να δημιουργούν κλάσεις στην Python και να κατανοούν έννοιες όπως την αρχικοποίηση μέσω μεθόδων constructors (`__init__`), τη δημιουργία και επικοινωνία αντικειμένων και τον προσδιοριστή `self`. Δημιουργούνται κατάλληλα προγράμματα, στα οποία οι εκπαιδευόμενοι/-ες αρχικοποιούν και καταστρέφουν αντικείμενα, καλούν τις μεθόδους και γενικά πειραματίζονται με τον νέο τρόπο προγραμματισμού. Η υποενότητα αποτελεί εισαγωγή στη σχεδίαση και τον προγραμματισμό με βάση την αντικειμενοστρεφή λογική.

2.3.Γ.2 | ΠΡΟΧΩΡΗΜΕΝΕΣ ΕΝΝΟΙΕΣ ΣΤΟΝ ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ

Στη δεύτερη υποενότητα του μαθήματος, οι εκπαιδευόμενοι/-ες έρχονται σε επαφή με προχωρημένες έννοιες του αντικειμενοστρεφούς προγραμματισμού. Μαθαίνουν για τις έννοιες της ενθυλάκωσης και τις μορφές ιδιωτικότητας (δημόσια/μη δημόσια-Public/Non-Public), για ιδιότητες και μεθόδους, μέσω της κάτω παύλας (`underscore`), μπροστά από το όνομα μιας μεθόδου ή μιας ιδιότητας. Μαθαίνουν για τις μεθόδους καταστροφής ενός αντικειμένου (`destructor`) και για τον μηχανισμό συλλογής «σκουπιδιών», δηλαδή αντικειμένων που δεν είναι ενεργά (Python).

Ένα από τα κύρια στοιχεία της αντικειμενοστρεφούς προσέγγισης είναι η κληρονομικότητα, βελτιώνοντας την επαναχρησιμοποίηση του κώδικα και την οργάνωσή του σε μια ιεραρχία κλάσεων. Η χρησιμότητά της θα καταστεί σαφής μέσα από κατάλληλα παραδείγματα, που αναδεικνύουν τα είδη κληρονομικότητας που υποστηρίζει η Python (απλή, πολλαπλή, πολυεπίπεδη, ιεραρχική και υβριδική κληρονομικότητα). Στη συνέχεια, οι εκπαιδευόμενοι/-ες εισάγονται στη χρήση βασικών αρχών της αντικειμενοστρεφούς προσέγγισης, όπως της αφαίρεσης και του πολυμορφισμού. Μαθαίνουν για τον πολυμορφισμό με μεθόδους κλάσεων και για τον πολυμορφισμό μέσω κληρονομικότητας. Μέσα από παραδείγματα, εξασκούνται στον πολυμορφισμό, χρησιμοποιώντας κληρονομικότητα και υπερφόρτωση μεθόδων.

Οι στατικές μέθοδοι και οι στατικές μεταβλητές, οι ειδικές μέθοδοι και οι μέθοδοι κλάσης είναι σημαντικό να διδαχθούν μέσα από αρκετά παραδείγματα και μικρής

έκτασης project που ανατίθενται στους/στις εκπαιδευόμενους/-ες, ώστε να γίνει κατανόητη η χρήση τους, καθώς αποτελούν δυσνόητες έννοιες.

2.3.Γ.3 | ΧΡΗΣΗ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ ΣΤΗΝ PYTHON

Η Python μπορεί να χρησιμοποιηθεί σε εφαρμογές βάσεων δεδομένων και να συνδυαστεί με εφαρμογές Διαδικτύου. Αρχικά, η ενότητα εστιάζεται σε κλασικά σχεσιακά συστήματα βάσεων δεδομένων, όπως είναι η MySQL και η MariaDB. Οι εκπαιδευόμενοι/-ες κατανοούν πώς γίνεται η σύνδεση με τα συστήματα βάσεων δεδομένων, πώς δημιουργούνται οι βάσεις δεδομένων και οι πίνακες και πώς εκτελούνται ερωτήματα SQL. Έτσι, θα είναι σε θέση να δημιουργούν εφαρμογές, που επεξεργάζονται στο πρόγραμμα τα δεδομένα που αντλούν από τις βάσεις δεδομένων.

Στη συνέχεια, ασχολούνται με NoSQL βάσεις δεδομένων. Μια από τις πιο δημοφιλείς βάσεις δεδομένων NoSQL είναι η MongoDB, που ήδη έχει διδαχθεί σε προηγούμενη ενότητα. Η MongoDB αποθηκεύει δεδομένα σε έγγραφα τύπου JSON, κάτι που την καθιστά πολύ ευέλικτη και επεκτάσιμη και μπορεί είτε να εγκατασταθεί τοπικά είτε μέσω μιας υπηρεσίας Cloud για χρήση βάσεων δεδομένων (<https://www.mongodb.com/cloud/atlas>). Οι εκπαιδευόμενοι/-ες μαθαίνουν για τον τρόπο σύνδεσης με τη βάση μέσω κάποιου driver, όπως του PyMongo. Στη συνέχεια, μαθαίνουν για τη δημιουργία βάσεων δεδομένων και τις εντολές δημιουργίας συλλογών (δηλαδή πινάκων στις τυπικές βάσεις δεδομένων), εισαγωγής δεδομένων, εκτέλεσης ερωτήσεων, διαγραφής δεδομένων κ.λπ. Η έμφαση της εκπαίδευσης δίνεται στην εκτέλεση ερωτήσεων και στο φιλτράρισμα των αποτελεσμάτων μέσω τροποποιητών (modifiers) ή κανονικών εκφράσεων. Επιπρόσθετα, μαθαίνουν για τον χειρισμό των δεδομένων σε μορφή JSON στην Python και τη μετατροπή τους από/σε δομές δεδομένων τύπου λεξικού.

2.3.Γ.4 | ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΩΝ ΕΦΑΡΜΟΓΩΝ ΜΕ PYTHON

Στην εισαγωγική ενότητα στην ανάπτυξη διαδικτυακών εφαρμογών με χρήση της γλώσσας προγραμματισμού Python, καλύπτονται ορισμένα βασικά θέματα που βοηθούν τους/τις εκπαιδευόμενους/-ες να εξοικειωθούν με τις βασικές αρχές και τεχνικές του πεδίου. Ένα σημαντικό κομμάτι της εκπαίδευσης επικεντρώνεται στην κατανόηση του πρωτοκόλλου HTTP και του μοντέλου αλληλεπίδρασης «πελάτη-εξυπηρετητή». Οι εκπαιδευόμενοι/-ες μαθαίνουν πώς λειτουργεί η επικοινωνία μεταξύ του προγράμματος «πελάτη και εξυπηρετητή», μέσω αιτημάτων και απαντήσεων HTTP. Στη συνέχεια, μαθαίνουν τις βασικές βιβλιοθήκες και εντολές για τη δημιουργία εφαρμογών, και για τη βιβλιοθήκη requests, που αποτελεί μια απλή βιβλιοθήκη HTTP για την υποβολή αιτημάτων HTTP. Το Beautiful Soup είναι ένα πακέτο Python για την ανάλυση εγγράφων HTML και XML, η html5lib μια βιβλιοθήκη για εργασία με έγγραφα HTML και, μέσω της βιβλιοθήκης webbrowser, μπορεί να γίνει χρήση φυλλομετρητών σελίδων από ένα πρόγραμμα. Οι εκπαιδευόμενοι/-ες, μέσα από εφαρ-

μογές, μπορούν να πειραματιστούν με αυτές τις βιβλιοθήκες και να επεξεργάζονται HTML αρχεία που κατεβάζουν μέσα από την Python.

Στην ενότητα, προτείνεται να διδαχθεί και η δημιουργία ενός απλού http server, χρησιμοποιώντας εντολές όπως `python3-m http.server`, αλλά και βιβλιοθήκες όπως η `http.server`. Ο server μπορεί να ακούει τα αιτήματα που στέλνονται μέσω ενός προγράμματος client και επιστρέφει κάποια αποτελέσματα. Η σύνδεση με μια βάση δεδομένων, μέσω του server, αποτελεί ένα παράδειγμα εφαρμογής που μπορεί να χρησιμοποιηθεί στην υποενότητα.

2.3.Γ.5 | ΧΡΗΣΗ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΩΝ ΠΛΑΙΣΙΩΝ ΓΙΑ ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΩΝ ΕΦΑΡΜΟΓΩΝ

Στην τρέχουσα ενότητα, οι εκπαιδευόμενοι/-ες μαθαίνουν για τα προγραμματιστικά πλαίσια για την ανάπτυξη εφαρμογών Διαδικτύου. Αρχικά, εξηγείται ότι ένα πλαίσιο είναι μια συλλογή από πακέτα και βιβλιοθήκες που αποτελούνται από έτοιμο, τυποποιημένο κώδικα, που υποστηρίζει την ανάπτυξη διαδικτυακών εφαρμογών, καθιστώντας την ανάπτυξη ταχύτερη και ευκολότερη και τις εφαρμογές πιο αξιόπιστες και επεκτάσιμες. Τα προγραμματιστικά πλαίσια για το Διαδίκτυο της Python χρησιμοποιούνται μόνο στο backend για τεχνολογία διακομιστή, βοηθώντας στη δρομολόγηση URL, στα αιτήματα και στις απαντήσεις HTTP, στην πρόσβαση σε βάσεις δεδομένων και σε ζητήματα ασφάλειας.

Στη συνέχεια, οι εκπαιδευόμενοι/-ες, προχωρούν στη μελέτη των βασικών πλαισίων ανάπτυξης διαδικτυακών εφαρμογών που χρησιμοποιούνται με Python, όπως το Flask και το Django. Το Django είναι ένα από τα πιο χρησιμοποιημένα πλαίσια για τη δημιουργία ιστοτόπων με Python. Το Flask θεωρείται ένα μικροπλαίσιο, που σημαίνει ότι διαθέτει λιγότερες λειτουργίες από πλαίσια όπως το Django, αλλά μπορεί να χρησιμοποιηθεί για πιο γρήγορη ανάπτυξη ιστοτόπων με αρκετά χαρακτηριστικά. Για τη δημιουργία εφαρμογών χρησιμοποιώντας αυτά τα πλαίσια, προτείνεται η χρήση κάποιου ολοκληρωμένου περιβάλλοντος ανάπτυξης (IDE) για διευκόλυνση της διαδικασίας ανάπτυξης – π.χ. PyCharm Community Edition (CE). Οι εκπαιδευόμενοι/-ες μαθαίνουν τη χρησιμότητα κάθε πλαισίου, τον τρόπο εγκατάστασης και χρήσης τους και αναπτύσσουν σχετικές εφαρμογές. Για παράδειγμα, με τη χρήση του Django μπορούν να αναπτύξουν εφαρμογές με χρήση δεδομένων από βάσεις δεδομένων.

2.3.Γ.6 | ΧΡΗΣΗ ΕΞΕΙΔΙΚΕΥΜΕΝΩΝ ΒΙΒΛΙΟΘΗΚΩΝ

Η Python είναι μια γλώσσα που διαθέτει πληθώρα βιβλιοθηκών και πακέτων, που επιτρέπουν την ανάπτυξη εφαρμογών διαφορετικής στόχευσης και τύπων. Συνεπώς, είναι σημαντικό οι εκπαιδευόμενοι/-ες να γνωρίσουν και να αναπτύξουν εφαρμογές με διαφορετικές βιβλιοθήκες. Αρχικά, προτείνεται να χρησιμοποιήσουν την `pygame`, με την οποία είχαν έρθει σε μια αρχική επαφή στην προηγούμενη μαθησιακή ενότητα που αφορούσε την Python. Οι εκπαιδευόμενοι/-ες είναι σε θέση να αναπτύξουν

μικρές εφαρμογές, παιχνίδια με παράθυρα, σύντομη κίνηση, μορφοποίηση κειμένου, ανίχνευση συμβάντων ποντικού ή πληκτρολογίου. Επιπρόσθετα, οι εκπαιδευόμενοι/-ες μπορούν να πειραματιστούν περισσότερο με τη βιβλιοθήκη Tkinter και να δημιουργήσουν πιο ολοκληρωμένες εφαρμογές. Στη συνέχεια, προτείνεται να γίνει εισαγωγή στην ανάλυση δεδομένων μέσω βιβλιοθηκών, όπως είναι οι TensorFlow, NumPy, SciPy, Pandas και Matplotlib. Μέσα από αυτές τις βιβλιοθήκες, μπορούν να επεξεργαστούν και να οπτικοποιήσουν δεδομένα. Μπορούν να παρουσιαστούν και να μελετηθούν έτοιμες εφαρμογές, ώστε να είναι πιο εύκολη η κατανόηση των δυνατοτήτων των βιβλιοθηκών. Στην υποενότητα αυτή, είναι σημαντικό να χρησιμοποιηθούν βιβλιοθήκες, σχετικές με την επεξεργασία και τη διαχείριση εικόνων (OpenCV, Scikit-image, Pillow κ.ά.), αλλά και τη διαχείριση αρχείων ήχου (pyAudioAnalysis και pydub). Μέσα από αυτές τις βιβλιοθήκες, οι εκπαιδευόμενοι/-ες μπορούν να πραγματοποιούν επεξεργασίες που έμαθαν σε άλλες ενότητες. Η μαθησιακή υποενότητα μπορεί να διδαχθεί μέσω ανάπτυξης σύντομων εφαρμογών, που αναδεικνύουν κάποιες από τις δυνατότητες των βιβλιοθηκών. Επίσης, μπορεί να γίνει συνδυασμός με τις προηγούμενες ενότητες, ώστε να αναπτυχθούν σχετικές διαδικτυακές δυνατότητες που κάνουν χρήση των βιβλιοθηκών σε επίπεδο server.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Αβούρης, Ν., Κουκιάς, Μ., Παλιουράς, Β., Σγάρμπας, Κ. (2023). *Python-Εισαγωγή στους υπολογιστές*. Ηράκλειο: Πανεπιστημιακές Εκδόσεις Κρήτης.
2. Μαγκούτης, Κ. & Νικολάου, Χ. (2015). *Εισαγωγή στον αντικειμενοστραφή προγραμματισμό με Python*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/1708>

Συμπληρωματικές

1. *Εισαγωγή στην Python*, Mathesis (χ.χ.). Ηράκλειο: Πανεπιστημιακές Εκδόσεις Κρήτης, ΙΤΕ: <https://mathesis.cup.gr/courses/course-v1:ComputerScience+CS1.1+20C/course/>
2. *Προχωρημένος προγραμματισμός με Python*, Mathesis. Ηράκλειο: Πανεπιστημιακές Εκδόσεις Κρήτης, ΙΤΕ: <https://mathesis.cup.gr/courses/course-v1:ComputerScience+CS1.2+20C/course/>

2.3.Δ • ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΙΙΙ (JAVA9/JAVAFX)- ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΕΦΑΡΜΟΓΩΝ ΓΡΑΦΙΚΗΣ ΔΙΕΠΑΦΗΣ ΧΡΗΣΤΗ (GUI) Ι

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Γλώσσα Προγραμματισμού ΙΙΙ (JAVA9/JAVAFX)-Προγραμματισμός εφαρμογών γραφικής διεπαφής χρήστη (GUI) Ι» εισάγει τους/τις εκπαιδευόμενους/-ες στη λογική του Προγραμματισμού με τη γλώσσα Python. Η αλγοριθμική λογική της γλώσσας προγραμματισμού JAVA έχει επιρροές από τη γλώσσα προγραμματισμού C11, η οποία διδάσκεται στη μαθησιακή ενότητα «Γλώσσα προγραμματισμού Ι (C11)-Αλγοριθμική και δομές δεδομένων» του εξαμήνου Α' και από τη μαθησιακή ενότητα «Γλώσσα προγραμματισμού ΙΙ (C++14)-Αντικειμενοστρεφής προγραμματισμός» του εξαμήνου Β' κατάρτισης. Συνεπώς, θα χρησιμοποιηθεί η πρότερη γνώση των δύο παραπάνω μαθησιακών εννοιτήτων. Στην παρούσα μαθησιακή ενότητα, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με προχωρημένα θέματα αντικειμενοστρεφούς προγραμματισμού, όπως είναι οι δομές επιλογής, επανάληψης, οι συναρτήσεις, οι εξαιρέσεις και η υπερφόρτωση, η δημιουργία κλάσεων και αντικειμένων, η χρήση μεθόδων κληρονομικότητας και πολυμορφισμού μέσω των ασφαλών τεχνικών ενθυλάκωσης των πεδίων κάθε αντικειμένου, δημιουργώντας έτσι πιο ανθεκτικές και ασφαλείς εφαρμογές κονσόλας. Επιπροσθέτως, ο/η εκπαιδευόμενος/-η θα διδαχθεί τεχνικές δημιουργίας εφαρμογών γραφικού περιβάλλοντος (GUI), τόσο μέσω GUI APIs, όπως τα awt και swing, όσο και μέσω JAVAFX. Έτσι, οι εκπαιδευόμενοι/-ες θα αποκτήσουν τις βασικές γνώσεις και δεξιότητες για τον προγραμματισμό πολύπλοκων εφαρμογών γραφικού περιβάλλοντος σε JAVA.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναπτύσσουν εφαρμογές κονσόλας ή/και γραφικού περιβάλλοντος με χρήση JAVA,
- Συνεισφέρουν με πρωτότυπο κώδικα στην κοινότητα των προγραμματιστών εφαρμογών,
- Τροποποιούν υπάρχοντα κώδικα, ανάλογα με τις προδιαγραφές των εφαρμογών που αναπτύσσουν,
- Αποφασίζουν για τη χρήση ή μη κατάλληλων βιβλιοθηκών, σχετικών με την ανάπτυξη εφαρμογών,
- Συνεργάζονται με άλλους/-ες εκπαιδευόμενους/-ες, συγκροτώντας ομάδες, με σκοπό τη δημιουργία και την επίλυση προβλημάτων των εφαρμογών,
- Υιοθετούν θετική στάση ως προς τις υπάρχουσες τεχνολογίες ανάπτυξης εφαρμογών,
- Αναγνωρίζουν τις βασικές προγραμματιστικές αρχές του Αντικειμενοστρεφούς Προγραμματισμού, βάσει των οποίων έχει υλοποιηθεί μια εφαρμογή JAVA,

- Επιλέγουν τα κατάλληλα μέλη κλάσεων (πεδία, μεθόδους κ.ά.), τα οποία θα χρησιμοποιήσουν στις εφαρμογές που θα αναπτύσσουν,
- Διακρίνουν μια διεπαφή (interface) από μια κλάση (class).

Βασικές έννοιες και λέξεις-κλειδιά

Αντικειμενοστρεφής προγραμματισμός / Cross-platform / Πολυνηματικότητα / Ασφάλεια / Just-in-Time / JDK / JRE / Αρχιτεκτονική JVM (JAVA Virtual Machine) / Garbage Collection / Κληρονομικότητα / Εξαιρέσεις / Κλάσεις, Διεπαφές.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (6), Σύνολο (6).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΗΝ JAVA9
2	ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ ΤΗΣ JAVA9
3	ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ (OOP) ΣΤΗΝ JAVA9
4	ΕΡΓΑΣΙΑ ΜΕ ΑΡΧΕΙΑ, STREAMS ΚΑΙ ΔΙΚΤΥΑΚΕΣ ΕΦΑΡΜΟΓΕΣ
5	ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΩΝ ΕΦΑΡΜΟΓΩΝ ΜΕ SWING ΚΑΙ JAVAFX
6	ΒΙΒΛΙΟΘΗΚΕΣ ΤΗΣ JAVA-ΕΙΣΑΓΩΓΗ ΣΤΟ SPRING FRAMEWORK

Κείμενο αναφοράς

2.3.Δ.1 | ΕΙΣΑΓΩΓΗ ΣΤΗΝ JAVA9

Σε συνέχεια της μαθησιακής ενότητας «Γλώσσα προγραμματισμού II (C++14)-Αντικειμενοστρεφής Προγραμματισμός», οι εκπαιδευόμενοι/-ες συνεχίζουν την εκμάθηση αντικειμενοστρεφών γλωσσών προγραμματισμού, μέσω της αλληλεπίδρασής τους με τη γλώσσα προγραμματισμού JAVA9. Οι εκπαιδευόμενοι/-ες εισέρχονται στον αχανή χώρο μιας γλώσσας προγραμματισμού, η οποία αποτελεί βασικό εργαλείο για την υλοποίηση ευρύτατου φάσματος εφαρμογών. Η JAVA9, μια γλώσσα προγραμματισμού ανεξάρτητης πλατφόρμας, επιτρέπει την εύκολη μεταφορά εφαρμογών μεταξύ διαφόρων συστημάτων. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν τον τρόπο δημιουργίας ενός bytecode αρχείου και τους τρόπους εκτέλεσής του. Εντός του πλαισίου αυτού, οι εκπαιδευόμενοι/-ες καλούνται να εξοικειωθούν με τη συντακτική δομή της JAVA9, αναλύοντας τη δομή ενός προγράμματος και τις διαδικασίες εκτέλεσής του. Ως εκ τούτου, αποκτούν γνώσεις επί του compiler και του debugger, δύο εργαλείων ουσιαστικής σημασίας για την ανάπτυξη και διόρθωση του λογισμικού, καθώς επίσης και για την κατηγοριοποίηση της JAVA9 ως Just-in-Time γλώσσας προγραμματισμού. Επιπροσθέτως, αναγκαία είναι η κατανόηση της αρχιτεκτονικής της JAVA Virtual Machine (JVM), καθώς η

λειτουργία της επηρεάζει δραστικά την εκτέλεση των προγραμμάτων, εξασφαλίζοντας την ομαλή λειτουργία, ανεξαρτήτως της υποκείμενης υλικοτεχνικής υποδομής. Η εισαγωγή στην JAVA9 βάζει τα θεμέλια για μια εμπειριστατωμένη κατανόηση των εννοιών και τεχνικών του αντικειμενοστρεφούς προγραμματισμού, θέτοντας τις βάσεις για την εξερεύνηση πιο σύνθετων δομών και για την υλοποίηση προηγμένων λύσεων λογισμικού. Αναφορικά με το περιβάλλον συγγραφής που θα χρησιμοποιηθεί, προτείνεται ένα IDE (BlueJ, NetBeans, IntelliJ, Eclipse κ.ά.), με την εγκατάσταση του απαιτούμενου Software (JDK) και την παραμετροποίηση που απαιτείται στις Μεταβλητές Συστήματος.

2.3.Δ.2 | ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ ΤΗΣ JAVA9

Στο πλαίσιο της εκπαίδευσης στην JAVA9, οι εκπαιδευόμενοι/-ες καλούνται να απομολώσουν βασικές έννοιες της γλώσσας. Η εκμάθηση, αρχικά, αφορά σε console επίπεδο. Αρχικά, η κατανόηση των μεταβλητών και των τύπων δεδομένων είναι ουσιώδης, καθώς αυτά τα στοιχεία επιτρέπουν την αντιστοίχιση τιμών με ονόματα (μεταβλητές, σταθερές, κυριολεκτικά) και τον καθορισμό της φύσης των τιμών που μπορούν να λάβουν οι μεταβλητές. Επιπλέον, οι ελεγκτές ροής διαδραματίζουν κρίσιμο ρόλο στη διαμόρφωση της σειράς εκτέλεσης των εντολών μιας JAVA9 εφαρμογής, επιτρέποντας την υλοποίηση δομών επιλογής (με αναφορά στη συμβατότητα της switch με strings σε κάθε case) και επανάληψης (βρόχους), με ιδιαίτερη αναφορά να πραγματοποιείται στο labeled break και τη ranged for (γνωστή και ως for-each). Η χρήση συναρτήσεων/μεθόδων και lambda functions ενθαρρύνει την επαναχρησιμοποίηση κώδικα και τη δόμησή του σε επαναλαμβανόμενες λειτουργικές διαδικασίες. Η εμβέλεια/διάρκεια ζωής των μεταβλητών αποτελούν σημαντικές έννοιες, διότι καθορίζουν το πεδίο ορατότητας και διαθεσιμότητας των δεδομένων στο πρόγραμμα. Επιπλέον, γίνεται αναφορά στην κλάση Scanner (είσοδο δεδομένων), στους τρόπους εμφάνισης (print, println, printf), στην υπερφόρτωση της μεθόδου toString() και στον υπολογισμό ψευδοτυχαίων αριθμών (Random/Secure Random και στις διαφορές τους). Τέλος, η κατανόηση δομών δεδομένων, όπως των πινάκων, των λιστών, των ArrayLists, των Collections κ.λπ., είναι ουσιώδης για την αποτελεσματική διαχείριση συλλογών δεδομένων. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τα είδη διαπέρασης πίνακα και τους αλγορίθμους ταξινόμησης πινάκων, χρησιμοποιώντας μεθόδους από έτοιμες κλάσεις κ.λπ., ενώ θα διδαχθούν και τη χρήση πακέτων ή/και τη δημιουργία τους. Αυτή η προσέγγιση επιτρέπει στους/στις εκπαιδευόμενους/-ες να αναπτύξουν μια σταθερή βάση πριν από την περαιτέρω εξειδίκευση στη γλώσσα και στις εφαρμογές της.

2.3.Δ.3 | ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ (OOP) ΣΤΗΝ JAVA9

Στο πλαίσιο της εκμάθησης της JAVA, απαραίτητο είναι για τους/τις εκπαιδευόμενους/-ες να εμβαθύνουν στις βασικές αρχές του αντικειμενοστρεφούς προγραμματισμού (Object Oriented Programming-OOP). Οι αρχές αυτές περιλαμβάνουν σημαντικές έννοιες, όπως την κληρονομικότητα, όπου μια κλάση αποκτά τα χαρακτηριστικά

μιας άλλης, την αφηρημένη κλάση και τη διεπαφή, οι οποίες επιτρέπουν τον ορισμό συμπεριφορών για τις υποκλάσεις, την εξαίρεση για τη διαχείριση σφαλμάτων, τη συλλογή για αποθήκευση και διαχείριση δεδομένων, καθώς επίσης και τον πολυνηματικό προγραμματισμό, ο οποίος επιτρέπει την πολλαπλή χρήση αντικειμένων σε παράλληλα εκτελούμενες διεργασίες. Πιο αναλυτικά, σε ό,τι αφορά στην κληρονομικότητα μεταξύ κλάσεων, οι εκπαιδευόμενοι/-ες θα διδαχθούν μια εκτενή αναφορά στα τρία βασικά είδη κληρονομικότητας (extends): απλή, ιεραρχική, πολυεπίπεδη, καθώς επίσης και μια αναφορά στην πολλαπλή κληρονομικότητα μέσω των interfaces (implements), ενώ, τέλος, θα πραγματοποιηθεί αναφορά και στην υβριδική κληρονομικότητα, ως έναν τρόπο σύνθεσης όλων των υπολοίπων ειδών κληρονομικότητας. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη δήλωση final και @Override και τις διάφορες περιπτώσεις χρήσης τους. Κατανοώντας την εφαρμογή αυτών των αρχών, οι εκπαιδευόμενοι/-ες μαθαίνουν να δημιουργούν κλάσεις και αντικείμενα, να χρησιμοποιούν μεθόδους και συναρτήσεις, και να διαχειρίζονται την κατάσταση και τη συμπεριφορά των αντικειμένων. Η συνολική κατανόηση και η ικανότητα εφαρμογής αυτών των αρχών είναι ζωτικής σημασίας για την ανάπτυξη αποδοτικού και ευέλικτου λογισμικού, ανοίγοντας τον δρόμο για την προσέγγιση προκλήσεων προγραμματισμού με στρατηγική και δημιουργικότητα.

2.3.Δ.4 | ΕΡΓΑΣΙΑ ΜΕ ΑΡΧΕΙΑ, STREAMS ΚΑΙ ΔΙΚΤΥΑΚΕΣ ΕΦΑΡΜΟΓΕΣ

Στη συνέχεια, οι εκπαιδευόμενοι/-ες αποκτούν σημαντικές δεξιότητες σχετικά με τον χειρισμό αρχείων, την επεξεργασία δεδομένων μέσω streams και την ανάπτυξη εφαρμογών Διαδικτύου. Η εμπέδωση στη διαχείριση αρχείων προσφέρει τη δυνατότητα αποθήκευσης και ανάκτησης δεδομένων, κρίσιμη για την ανάπτυξη εφαρμογών που απαιτούν μόνιμη αποθήκευση πληροφοριών. Από την άλλη πλευρά, η εργασία με streams επιτρέπει την αποτελεσματική επεξεργασία δεδομένων σε πραγματικό χρόνο, ενισχύοντας την απόδοση των εφαρμογών. Η κατανόηση και η ανάπτυξη εφαρμογών Διαδικτύου ανοίγει νέους δρόμους για την αντίληψη του τρόπου λειτουργίας των υποστηρωμάτων, τα οποία συναποτελούν το Διαδίκτυο. Η επιτυχής ενσωμάτωση των βιβλιοθηκών JAVA σε ένα έργο (Project) με αρχεία, streams και επικοινωνία με το Διαδίκτυο είναι καθοριστική για την ανάπτυξη σύγχρονων, αποδοτικών και ασφαλών εφαρμογών. Οι εκπαιδευόμενοι/-ες ενθαρρύνονται να εξοικειωθούν με τα αντίστοιχα προγραμματιστικά παραδείγματα και τις ασκήσεις, προκειμένου να κατανοήσουν βαθύτερα τη λειτουργία και την εφαρμογή των εν λόγω τεχνολογιών. Η πρακτική εφαρμογή των δεξιοτήτων αυτών, στο πλαίσιο πραγματικών σεναρίων, αποτελεί ουσιαστικό μέρος της διαδικασίας μάθησης, διευκολύνοντας τους/τις εκπαιδευόμενους/-ες να αντιληφθούν τη σημασία και την εφαρμογή της τεχνολογίας στην καθημερινότητα. Η διερεύνηση των βιβλιοθηκών JAVA προσφέρει τα εργαλεία και τις απαραίτητες γνώσεις για την ανάπτυξη ευέλικτων και δυναμικών λύσεων στο πεδίο της Πληροφορικής, ενισχύοντας το επίπεδο κατάρτισης και την επαγγελματική ετοιμότητα των εκπαιδευομένων.

2.3.Δ.5 | ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΩΝ ΕΦΑΡΜΟΓΩΝ ΜΕ SWING ΚΑΙ JAVA FX

Σε αυτή τη μαθησιακή υποενοότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν πώς να χρησιμοποιούν τη βιβλιοθήκη Swing της JAVA για τη δημιουργία γραφικών εφαρμογών. Θα διδαχθούν πώς να δημιουργούν παράθυρα, πώς να προσθέτουν στοιχεία σε ένα παράθυρο, πώς να διαχειρίζονται τα συμβάντα και πώς να χρησιμοποιούν τα layouts. Θα πρέπει να κατανοήσουν πώς να χρησιμοποιούν τις βιβλιοθήκες της Swing για τη δημιουργία γραφικών εφαρμογών, πώς να δημιουργούν παράθυρα και πώς να προσθέτουν στοιχεία σε αυτά. Επιπλέον, θα πρέπει να κατανοήσουν πώς να διαχειρίζονται τα συμβάντα και πώς να χρησιμοποιούν τα layouts για τη δημιουργία πιο πολύπλοκων γραφικών εφαρμογών. Στη συνέχεια, και με δεδομένο ότι η χρήση της Swing θεωρείται ξεπερασμένη, οι εκπαιδευόμενοι/-ες θα διδαχθούν πώς να χρησιμοποιούν την τεχνολογία της JavaFX για τη δημιουργία γραφικών εφαρμογών. Θα διδαχθούν πώς να δημιουργούν παράθυρα, πώς να προσθέτουν στοιχεία σε ένα παράθυρο, πώς να διαχειρίζονται τα συμβάντα και πώς να χρησιμοποιούν τα layouts. Η JavaFX παρέχει πιο προηγμένες δυνατότητες για τη δημιουργία γραφικών, όπως τα 3D graphics και τα multimedia. Οι εκπαιδευόμενοι/-ες θα πρέπει να κατανοήσουν πώς να χρησιμοποιούν τις βιβλιοθήκες της JavaFX για τη δημιουργία γραφικών εφαρμογών, πώς να δημιουργούν παράθυρα και πώς να προσθέτουν στοιχεία σε αυτά. Επιπλέον, θα πρέπει να κατανοήσουν πώς να διαχειρίζονται τα συμβάντα και πώς να χρησιμοποιούν τα layouts για τη δημιουργία πιο πολύπλοκων γραφικών εφαρμογών.

2.3.Δ.6 | ΒΙΒΛΙΟΘΗΚΕΣ ΤΗΣ JAVA-ΕΙΣΑΓΩΓΗ ΣΤΟ SPRING FRAMEWORK

Σε αυτή τη μαθησιακή υποενοότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τις διάφορες βιβλιοθήκες της JAVA, τις οποίες να μπορούν να χρησιμοποιήσουν για να επεκτείνουν τις δυνατότητες των εφαρμογών τους. Θα μάθουν για την τεχνολογία JDBC (JAVA DataBase Connection), η οποία αφορά την πρόσβαση σε βάσεις δεδομένων, την JPA (JAVA Persistence API), για τη διαχείριση των αντικειμένων της βάσης δεδομένων, και την JAX-RS (JAVA API for XML Web Services) για τη δημιουργία RESTful Web Services. Η JDBC επιτρέπει στους/στις εκπαιδευόμενους/-ες να εκτελούν SQL ερωτήματα από την JAVA, ενώ η JPA επιτρέπει στους/στις εκπαιδευόμενους/-ες να διαχειρίζονται τα αντικείμενα της βάσης δεδομένων με έναν πιο αντικειμενοστρεφή τρόπο. Τέλος, η JAX-RS επιτρέπει στους/στις εκπαιδευόμενους/-ες να δημιουργούν υπηρεσίες web REST, έναν τρόπο δημιουργίας εφαρμογών που μπορούν να επικοινωνούν με άλλες εφαρμογές μέσω του Διαδικτύου. Επιπλέον, θα πρέπει να κατανοήσουν πώς να χρησιμοποιούν τις βιβλιοθήκες της JAVA για την εργασία με αρχεία και streams, και πώς να χρησιμοποιούν τις βιβλιοθήκες της JAVA για τη δημιουργία εφαρμογών Διαδικτύου. Επίσης, οι εκπαιδευόμενοι/-ες θα διδαχθούν πώς να χρησιμοποιούν τη Spring, ένα framework, για τη δημιουργία εφαρμογών JAVA. Η Spring επιτρέπει στους/στις εκπαιδευόμενους/-ες να διαχωρίζουν τα configurations της εφαρμογής τους από τον κώδικα, ενώ η δυνατότητα παρέμβασης που παρέχει η Spring επιτρέπει στους/στις εκπαιδευόμενους/-ες

να προσθέτουν λειτουργικότητα στην εφαρμογή τους, χωρίς παρέμβαση στον πηγαίο κώδικα. Τέλος, θα πρέπει να κατανοήσουν πώς να χρησιμοποιούν τις βιβλιοθήκες της Spring για τη δημιουργία εφαρμογών, πώς να διαχειρίζονται το configuration της εφαρμογής και πώς να προσθέτουν νέες συμπεριφορές στην εφαρμογή τους.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Deitel, M. H. & Deitel, J. P. (2015). *JAVA How To Program (Early Objects)*. Αθήνα: Εκδόσεις Κλειδάριθμος.
2. Schildt, H. & Coward, D. (2024). *Java: The Complete Reference*. Ohio: McGraw Hill.

Συμπληρωματικές

1. Loy, M., Niemeyer, P. & Leuck, D. (2023). *Learning Java: An Introduction to Real-World Programming with JAVA*. California: O'Reilly.
2. Ullenboom, C. (2024). *Spring Boot 3 and Spring Framework 6*. Bonn: Rheinwerk Publishing.

2.3.E • ΣΥΣΤΗΜΑΤΑ ΕΛΕΓΧΟΥ ΕΚΔΟΣΕΩΝ-WORKFLOWS ΚΑΙ ΣΥΝΕΡΓΑΤΙΚΕΣ ΤΕΧΝΙΚΕΣ (GIT, GITLAB, GITHUB)

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Συστήματα ελέγχου εκδόσεων-workflows και συνεργατικές τεχνικές (Git, GitLab, GitHub)» εισάγει τους/τις εκπαιδευόμενους/-ες στα εργαλεία ελέγχου εκδόσεων κατά τη διαδικασία ανάπτυξης μιας εφαρμογής/ενός λογισμικού. Στην παρούσα μαθησιακή ενότητα, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με συχνά προβλήματα στην ανάπτυξη λογισμικού και θα διδαχθεί τον ορισμό και τη λειτουργία ενός VCS (Version Control System) γενικά. Επιπλέον, θα διδαχθεί τα βασικά του git (λήψη και εγκατάσταση), δημιουργώντας έργα στο git (project directory, “git init”, “git status”, “.git” directory κ.ά.), τη βασική ροή εργασιών του Git, την προσθήκη, την τροποποίηση, την καταχώρηση αρχείων (δημιουργία νέου αρχείου, “tracked vs untracked”, “3 stages”, “git add”, “git commit”, “git log” κ.λπ.), τα στάδια επιθεώρησης αλλαγών (diff “working directory” vs “last commit”, “git diff”), και την αφαίρεση αρχείου από το αποθετήριο git (“git ls-files”, “git rm <file>”, “git rm <file1> <file2>...”, “git rm -r <dir>”). Έτσι, οι εκπαιδευόμενοι/-ες εξοικειώνονται με τη ροή εργασιών/εντολών, τις τεχνικές διακλαδώσεων (branching)-συγχώνευσης (merging)-επίλυσης συγκρούσεων, με το gitlab (απομακρυσμένο καταναμεμημένο σύστημα ελέγχου εκδόσεων) και τη δημιουργία υπό-εκδόσεων SVN (SubVersion),

καθώς και τον χειρισμό συγκεντρωτικού συστήματος ελέγχου εκδόσεων. Προαιρετικά, θα μπορούν να δουν τον τρόπο εγκατάστασης και χειρισμού του Tortoise SVN.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Περιγράφουν τι είναι τα Συστήματα Ελέγχου Εκδόσεων λογισμικού και τις κατηγορίες τους,
- Απαριθμούν τους τρόπους με τους οποίους τα Συστήματα Ελέγχου Εκδόσεων λογισμικού ενισχύουν τη συνεργατικότητα στον κύκλο ανάπτυξης λογισμικού,
- Χρησιμοποιούν Συστήματα Ελέγχου Εκδόσεων, να τα εγκαθιστούν, να τα ρυθμίζουν και να τα χρησιμοποιούν στην ανάπτυξη εφαρμογών, με ιδιαίτερη εστίαση στη συνεργασία τους με άλλους/-ες,
- Παρέχουν προς λήψη (download) ενιαία αρχεία (.zip), μέσα στα οποία θα υπάρχουν αναρτημένα όλα τα αρχεία πηγαίου κώδικα για κάθε αποθετήριο (repository),
- Ενσωματώνουν τη χρήση Συστημάτων Ελέγχου Εκδόσεων λογισμικού σε όλες τις γλώσσες προγραμματισμού στις οποίες αναπτύσσουν εφαρμογές,
- Ανανεώνουν τις υλοποιήσεις των εφαρμογών τους με κατάλληλα «κλαδιά» (branches), ώστε να μπορεί να αναζητηθεί η πιο πρόσφατη έκδοση υλοποίησης κάθε αρχείου κώδικα, καθώς και παλαιότερες υλοποιήσεις του ίδιου αρχείου κώδικα,
- Απαριθμούν τις δημοφιλέστερες υπηρεσίες φιλοξενίας αποθετηρίων και τα πλεονεκτήματα που αυτές προσφέρουν.

Βασικές έννοιες και λέξεις-κλειδιά

Version Control Systems (VCS) / Repository / Commit / Branch / Merge / Pull Request / Fork / Clone / Push/Pull / Conflict Resolution / Version History / Collaboration.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (2), Σύνολο (2).

Διάρθρωση των μαθησιακών εννοιών σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΑ ΣΥΣΤΗΜΑΤΑ ΕΛΕΓΧΟΥ ΕΚΔΟΣΕΩΝ ΛΟΓΙΣΜΙΚΟΥ
2	ΕΙΣΑΓΩΓΗ ΣΤΟ GIT-ΠΡΟΧΩΡΗΜΕΝΕΣ ΤΕΧΝΙΚΕΣ GIT
3	ΕΙΣΑΓΩΓΗ ΣΤΟ GITHUB-ΠΡΟΧΩΡΗΜΕΝΕΣ ΤΕΧΝΙΚΕΣ GITHUB
4	ΕΙΣΑΓΩΓΗ ΣΤΟ GITLAB-ΠΡΟΧΩΡΗΜΕΝΕΣ ΤΕΧΝΙΚΕΣ GITLAB
5	WORKFLOWS ΚΑΙ ΣΥΝΕΡΓΑΤΙΚΕΣ ΤΕΧΝΙΚΕΣ ΣΤΑ ΣΥΣΤΗΜΑΤΑ ΕΛΕΓΧΟΥ ΕΚΔΟΣΕΩΝ
6	ΑΣΦΑΛΕΙΑ ΚΑΙ ΠΡΟΣΤΑΣΙΑ ΔΕΔΟΜΕΝΩΝ-ΕΡΓΑΛΕΙΑ ΚΑΙ ΠΟΡΟΙ-ΕΦΑΡΜΟΓΗ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΤΑΣΕΙΣ ΣΤΑ ΣΥΣΤΗΜΑΤΑ ΕΛΕΓΧΟΥ ΕΚΔΟΣΕΩΝ

2.3.E.1 | ΕΙΣΑΓΩΓΗ ΣΤΑ ΣΥΣΤΗΜΑΤΑ ΕΛΕΓΧΟΥ ΕΚΔΟΣΕΩΝ ΛΟΓΙΣΜΙΚΟΥ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη θεμελιώδη σημασία των Συστημάτων Ελέγχου Εκδόσεων (Version Control Systems-VCS), μια έννοια που είναι κρίσιμη για την κατανόηση της σύγχρονης ανάπτυξης λογισμικού. Θα εξεταστεί η βασική λειτουργία των VCS, συμπεριλαμβανομένων των στόχων και των πλεονεκτημάτων τους. Οι εκπαιδευόμενοι/-ες θα διδαχθούν πώς τα VCS διευκολύνουν την ομαδική εργασία, επιτρέποντας την παράλληλη εργασία και την επίλυση conflicts. Θα εξεταστεί, επίσης, η χρήση των VCS για την παρακολούθηση αλλαγών, την αναίρεση αλλαγών και τη διατήρηση ιστορικού των αλλαγών. Οι εκπαιδευόμενοι/-ες θα διδαχθούν την αξία των VCS στην ανάπτυξη και τη συντήρηση της ποιότητας του λογισμικού, καθώς και την προστασία από απώλεια δεδομένων. Αυτή η υποενότητα θα αποτελέσει τη βάση για την κατανόηση των επόμενων κεφαλαίων, που αφορούν συγκεκριμένα συστήματα ελέγχου εκδόσεων, όπως το Git, το GitLab και το GitHub, για τα οποία προτείνεται το Microsoft Visual Studio 2019, νεότερο ή παρεμφερές λογισμικό IDE. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τις διάφορες κατηγορίες των συστημάτων ελέγχου εκδόσεων, όπως είναι τα κεντρικά/κατανεμημένα/υβριδικά συστήματα. Αυτή η γνώση θα επιτρέψει στους/στις εκπαιδευόμενους/-ες να κατανοήσουν τη σημασία των συστημάτων ελέγχου εκδόσεων στη σύγχρονη ανάπτυξη λογισμικού. Οι εκπαιδευόμενοι/-ες θα διδαχθούν την αναγκαιότητα της μάθησης και προσαρμογής στις νέες τεχνολογίες και εργαλεία που εμφανίζονται στον τομέα της ανάπτυξης λογισμικού. Θα κατανοήσουν, επίσης, την αξία της συνεργατικής εργασίας και της αποτελεσματικής διαχείρισης των εκδόσεων στην παραγωγή υψηλής ποιότητας λογισμικού. Το κεφάλαιο αυτό θα προετοιμάσει τους/τις εκπαιδευόμενους/-ες, εξοπλίζοντάς τους με τις απαραίτητες γνώσεις και δεξιότητες, για να ανταποκριθούν στις απαιτήσεις της σύγχρονης ανάπτυξης λογισμικού.

2.3.E.2 | ΕΙΣΑΓΩΓΗ ΣΤΟ GIT-ΠΡΟΧΩΡΗΜΕΝΕΣ ΤΕΧΝΙΚΕΣ GIT

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τα βασικά του Git, ένα από τα πιο δημοφιλή συστήματα ελέγχου εκδόσεων. Θα εξεταστεί η δομή του Git, συμπεριλαμβανομένων των αποθετηρίων (repositories), των branches, των commits και των tags. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τις βασικές εντολές του Git, όπως το `git init`, `git add`, `git commit`, `git push`, `git pull` και `git clone`. Θα διδαχθούν πώς να δημιουργούν ένα νέο αποθετήριο, πώς να προσθέτουν αρχεία σε ένα αποθετήριο, πώς να κάνουν commit τις αλλαγές τους και πώς να ανεβάζουν (push) τις αλλαγές τους σε ένα απομακρυσμένο αποθετήριο. Επίσης, θα διδαχθούν πώς να κατεβάζουν (pull) τις αλλαγές από ένα απομακρυσμένο αποθετήριο και πώς να κλωνοποιούν ένα αποθετήριο. Τέλος, θα εξεταστεί η σημασία της συνεργατικής εργασίας με το Git και πώς το Git επιτρέπει στους/στις προγραμματιστές/-ίστριες να συνεργάζονται απο-

τελεσματικά σε ένα project. Οι εκπαιδευόμενοι/-ες θα διδαχθούν την αναγκαιότητα της συνεχούς εκμάθησης και προσαρμογής στις νέες τεχνολογίες και τα εργαλεία που εμφανίζονται στον τομέα της ανάπτυξης λογισμικού. Θα κατανοήσουν, επίσης, την αξία της συνεργατικής εργασίας και της αποτελεσματικής διαχείρισης των εκδόσεων στην παραγωγή υψηλής ποιότητας λογισμικού. Επιπλέον, θα εξεταστεί η σημασία της διαχείρισης των εκδόσεων στην παραγωγή υψηλής ποιότητας λογισμικού και πώς το Git μπορεί να βοηθήσει στην επίτευξη αυτού του στόχου. Οι εκπαιδευόμενοι/-ες θα διδαχθούν, επίσης, πώς το Git μπορεί να βοηθήσει στην αποφυγή και στην επίλυση των συγκρούσεων κατά τη συνεργατική εργασία, καθιστώντας τη διαδικασία ανάπτυξης λογισμικού πιο αποτελεσματική.

2.3.E.3 | ΕΙΣΑΓΩΓΗ ΣΤΟ GITHUB-ΠΡΟΧΩΡΗΜΕΝΕΣ ΤΕΧΝΙΚΕΣ GITHUB

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν το GitHub, μια από τις πιο δημοφιλείς και ισχυρές πλατφόρμες διαχείρισης αποθετηρίων Git. Θα συγκριθούν και θα αξιολογηθούν οι δυνατότητες, οι λειτουργίες και οι υπηρεσίες που προσφέρουν στους/στις χρήστες/-τριές τους, όπως η φιλοξενία, η δημιουργία, η κοινοποίηση, η αναζήτηση, η αξιολόγηση, η επίβλεψη και η ασφάλεια των αποθετηρίων Git. Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα διδαχθούν πιο προχωρημένες τεχνικές Git, όπως η συγχώνευση και η επαναφορά. Θα μάθουν πώς να συγχωνεύουν διακλαδώσεις (branches), με την εντολή git merge, και πώς να επαναφέρουν αλλαγές, με την εντολή git revert. Επίσης, θα διδαχθούν πώς να χρησιμοποιούν το GitHub, έναν δημοφιλή ιστότοπο για τη φιλοξενία αποθετηρίων Git. Θα μάθουν πώς να δημιουργούν ένα αποθετήριο στο GitHub, πώς να ανεβάζουν (push) τις αλλαγές τους σε ένα αποθετήριο GitHub και πώς να κατεβάζουν (pull) τις αλλαγές από ένα αποθετήριο GitHub. Θα διδαχθούν, ακόμη, πώς να συνεργάζονται μέσω του GitHub, πώς να χρησιμοποιούν τα αιτήματα ενσωμάτωσης (pull requests) για την επιθεώρηση κώδικα και πώς να διαχειρίζονται τα ζητήματα (issues) που ανακύπτουν. Τέλος, οι εκπαιδευόμενοι/-ες θα κατανοήσουν πώς το GitHub μπορεί να βοηθήσει στην αποτελεσματική διαχείριση των εκδόσεων και στην αποτελεσματική συνεργατική εργασία. Θα εξεταστεί, επίσης, η σημασία της συνεργατικής εργασίας και της αποτελεσματικής διαχείρισης των εκδόσεων στην παραγωγή υψηλής ποιότητας λογισμικού.

2.3.E.4 | ΕΙΣΑΓΩΓΗ ΣΤΟ GITLAB-ΠΡΟΧΩΡΗΜΕΝΕΣ ΤΕΧΝΙΚΕΣ GITLAB

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν πώς να χρησιμοποιούν το GitLab, έναν ισχυρό ιστότοπο για τη φιλοξενία αποθετηρίων Git. Θα μάθουν τους τρόπους με τους οποίους θα δημιουργούν ένα αποθετήριο στο GitLab, με ποιο τρόπο θα αναρτούν (push) τις αλλαγές τους σε ένα αποθετήριο GitLab και με ποιον τρόπο θα πραγματοποιούν λήψη (pull) των αλλαγών από ένα αποθετήριο GitLab. Επίσης, οι εκπαιδευόμενοι/-ες θα διδαχθούν τους τρόπους με τους οποίους θα συνεργάζονται μέσω του GitLab, πώς θα χρησιμοποιούν τα αιτήματα ενσωμά-

τωσης (merge requests) κατά τον έλεγχο του κώδικα, και πώς θα διαχειρίζονται ζητήματα (issues) που θα ανακύπτουν. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν πιο προχωρημένες τεχνικές GitLab, όπως η χρήση των διακλαδώσεων (branches) και των αιτημάτων ενσωμάτωσης. Θα μάθουν πώς να δημιουργούν διακλαδώσεις για νέες λειτουργίες ή διορθώσεις, πώς να μεταβαίνουν μεταξύ διακλαδώσεων και πώς να συγχωνεύουν τις αλλαγές από τη μια διακλάδωση στην άλλη. Τέλος, θα διδαχθούν πώς τα αιτήματα ενσωμάτωσης και οι διακλαδώσεις μπορούν να χρησιμοποιηθούν για την υλοποίηση προχωρημένων σεναρίων εργασίας, όπως της ανάπτυξης λειτουργιών, της διόρθωση σφαλμάτων και της διαχείρισης αποθετηρίων με πολλούς συνεισφέροντες (contributors). Οι εκπαιδευόμενοι/-ες θα διδαχθούν, επίσης, πώς το GitLab μπορεί να βοηθήσει στην αποτελεσματική διαχείριση των εκδόσεων και στην αποτελεσματική συνεργατική εργασία.

2.3.E.5 | WORKFLOWS ΚΑΙ ΣΥΝΕΡΓΑΤΙΚΕΣ ΤΕΧΝΙΚΕΣ

ΣΤΑ ΣΥΣΤΗΜΑΤΑ ΕΛΕΓΧΟΥ ΕΚΔΟΣΕΩΝ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τα διάφορα workflows που μπορούν να χρησιμοποιηθούν στα συστήματα ελέγχου εκδόσεων. Πιο συγκεκριμένα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τους τρόπους με τους οποίους θα χρησιμοποιούν τα workflows, για να διαχειρίζονται τις εκδόσεις του λογισμικού, τους τρόπους συνεργασίας με άλλους/-ες χρήστες/-τριες, και πώς να διαχειρίζονται αλλαγές στον κώδικα. Επιπλέον, θα διδαχθούν τα διάφορα workflows και τις τεχνικές που μπορούν να εφαρμόσουν για να οργανώσουν, να συντονίσουν και να βελτιώσουν τη συνεργασία με το Git, όπως το κεντρικό, το χαρακτηριστικό, το κλαδικό, το αναδιπλούμενο, το αναθεωρητικό, το διαδραστικό κ.ά. Θα διδαχθούν, επίσης, πώς να χρησιμοποιούν τα workflows, για να διαχειρίζονται τις εκδόσεις του λογισμικού, πώς να συνεργάζονται με άλλους χρήστες και πώς να διαχειρίζονται αλλαγές που θα απαιτούνται στον κώδικα. Επίσης, οι εκπαιδευόμενοι/-ες θα διδαχθούν πώς να συνεργάζονται αποτελεσματικά με άλλους/-ες χρήστες/-τριες σε ένα σύστημα ελέγχου εκδόσεων. Θα μάθουν πώς να χρησιμοποιούν τα εργαλεία συνεργασίας που παρέχονται από τα συστήματα ελέγχου εκδόσεων, όπως τα αιτήματα ενσωμάτωσης (merge requests), ο σχολιασμός, καθώς επίσης και τα issues τα οποία θα απαιτούν επίλυση. Θα διδαχθούν, επίσης, πώς να χρησιμοποιούν τα εργαλεία συνεργασίας που παρέχονται από τα συστήματα ελέγχου εκδόσεων. Τέλος, οι εκπαιδευόμενοι/-ες θα διδαχθούν τους τρόπους με τους οποίους τα workflows και οι συνεργατικές τεχνικές μπορούν να βοηθήσουν στην αποτελεσματική διαχείριση των εκδόσεων και στην παραγωγή υψηλής ποιότητας λογισμικού. Θα μελετηθεί, τέλος, η σημασία της συνεργατικής εργασίας και της αποτελεσματικής διαχείρισης των εκδόσεων στην παραγωγή υψηλής ποιότητας λογισμικού.

2.3.E.6 | ΑΣΦΑΛΕΙΑ ΚΑΙ ΠΡΟΣΤΑΣΙΑ ΔΕΔΟΜΕΝΩΝ-ΕΡΓΑΛΕΙΑ ΚΑΙ

ΠΟΡΟΙ-ΕΦΑΡΜΟΓΗ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΤΑΣΕΙΣ ΣΤΑ ΣΥΣΤΗΜΑΤΑ ΕΛΕΓΧΟΥ ΕΚΔΟΣΕΩΝ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τις καλύτερες πρακτικές για την ασφάλεια και την προστασία των δεδομένων σε ένα σύστημα ελέγχου εκδόσεων. Θα διδαχθούν πώς να χρησιμοποιούν τα εργαλεία και τις τεχνικές που είναι διαθέσιμες για την προστασία των δεδομένων και την ασφάλεια των συστημάτων ελέγχου εκδόσεων, όπως είναι τα δικαιώματα πρόσβασης, τα πρωτόκολλα ασφάλειας και οι μέθοδοι κρυπτογράφησης. Επίσης, οι εκπαιδευόμενοι/-ες θα διδαχθούν για τα διάφορα εργαλεία και τους διάφορους πόρους, που είναι διαθέσιμοι για την ενίσχυση της αποτελεσματικότητας των συστημάτων ελέγχου εκδόσεων, και θα μάθουν πώς να χρησιμοποιούν τα εργαλεία αυτά, για να βελτιώσουν την παραγωγικότητα και την αποτελεσματικότητα των συστημάτων ελέγχου εκδόσεων. Επίσης, οι εκπαιδευόμενοι/-ες θα διδαχθούν τις τρέχουσες τάσεις και τις μελλοντικές προοπτικές στα συστήματα ελέγχου εκδόσεων και, στη συνέχεια, θα μάθουν πώς οι τεχνολογικές εξελίξεις και οι νέες πρακτικές μπορούν να επηρεάσουν τη χρήση των συστημάτων ελέγχου εκδόσεων και πώς μπορούν να προετοιμαστούν για αυτές. Τέλος, οι εκπαιδευόμενοι/-ες θα διδαχθούν πώς να εφαρμόζουν τις γνώσεις και τις δεξιότητες που απέκτησαν για το Git σε πραγματικά έργα λογισμικού, καθώς και ποια είναι τα εργαλεία/πόροι που μπορούν να χρησιμοποιήσουν για να βελτιώσουν την ποιότητα και την απόδοση του κώδικά τους. Θα μάθουν πώς να συνδέονται και να συνεργάζονται με άλλα άτομα ή οργανισμούς που χρησιμοποιούν το Git, όπως συνάδελφοι/-ισσες, πελάτες/-ισσες, εκπαιδευτές/-τριες και ερευνητές/-τριες, και θα μάθουν να αντιμετωπίζουν τις προκλήσεις και τις δυσκολίες που προκύπτουν κατά την εφαρμογή του Git σε περιβάλλοντα υψηλής πολυπλοκότητας και απαιτήσεων, όπως είναι η ασφάλεια, η απομακρυσμένη εργασία, η επαναχρησιμοποίηση και η ενσωμάτωση κώδικα.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Dawson, C. & Straub, B. (2016). *Building Tools with GitHub-Customize Your Workflow*. California: O'Reilly.
2. Omos, U. V. (2020). *Big Little Book on Git-Gitlab Github GitOps DevOps & CI/CD: Git-Gitlab Github GitOps DevOps & CI/CD*. Gridlockaz.

Συμπληρωματικές

1. Hutten, D. (2017). *Git-Learn Version Control with Git-A step-by-step Ultimate beginners Guide*. CreateSpace Independent Publishing Platform.
2. Gajda, W. (2013). *Git Recipes-A Problem-Solution Approach*. New York: Apress.

2.4 • ΕΞΑΜΗΝΟ Δ΄

2.4.A • ΑΝΑΛΥΣΗ ΚΑΙ ΣΧΕΔΙΑΣΜΟΣ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Ανάλυση και σχεδιασμός πληροφοριακών συστημάτων» εισάγει τους/τις εκπαιδευόμενους/-ες στους τρόπους και στα μεθοδολογικά εργαλεία ανάλυσης ενός περιγραφόμενου προβλήματος σε απλούστερες διαδικασίες και στα στάδια σχεδιασμού του λογισμικού που θα το υλοποιεί, μέσα από τη δημιουργία διαγράμματος, το οποίο δείχνει τα είδη και τις σχέσεις εξάρτησης των δρώντων με το περιγραφόμενο σύστημα, καθώς και με τις διακριτές μεταξύ τους λειτουργίες εντός του συστήματος. Ο/Η εκπαιδευόμενος/-η θα διδαχθεί τη διαφορά των όρων «Δεδομένα» και «Πληροφορία», καθώς και τη σημασία που έχει το φαινόμενο των «αναδυόμενων ιδιοτήτων» μέσα σε ένα Πληροφοριακό Σύστημα. Επιπλέον, θα διδαχθεί μεθοδολογίες ανάλυσης Πληροφοριακών Συστημάτων, όπως SSADM/RUP/Scrum. Εν συνεχεία, ο/η εκπαιδευόμενος/-η θα μάθει τον τρόπο δημιουργίας Διαγράμματος Ροής Δεδομένων (ΔΡΔ) που προκύπτει από κάποιο αναλυόμενο σύστημα, είτε αυτό αφορά σε διάγραμμα μηδενικού επιπέδου είτε σε διαγράμματα κατώτερων επιπέδων (1ου, 2ου κ.λπ.). Επίσης, θα διδαχθεί την ανάλυση απαιτήσεων: επιχειρηματικές διαδικασίες, εντοπισμός απαιτήσεων (λειτουργικές, μη λειτουργικές), καταγραφή απαιτήσεων, ιστορίες χρηστών (user stories κ.λπ.). Επιπλέον, θα γίνεται εκτενής αναφορά σε ελέγχους του κύκλου της πληροφορίας εντός του διαγράμματος και της επικοινωνίας αυτού με τους δράστες, τις επεξεργασίες και τα αρχεία (βάσεις δεδομένων), με σκοπό τη διόρθωση «μαύρων τρυπών» κατά την επικοινωνία με αρχεία (βάσεις δεδομένων) κ.ά. Τέλος, θα πραγματοποιείται μετατροπή του διαγράμματος ροών δεδομένων σε Μοντέλο Οντοτήτων Συσχετίσεων.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναγνωρίζουν τις επεξεργασίες από τις οποίες αποτελείται ένα Πληροφοριακό Σύστημα, δημιουργώντας ένα Διάγραμμα Ροής Δεδομένων, το οποίο αποτελείται από την επικοινωνία όλων των επεξεργασιών μεταξύ τους, με τους/τις χρήστες/-τριες του Πληροφοριακού συστήματος και με τα αρχεία (βάσεις δεδομένων),
- Αναλύουν τις δομημένες περιγραφές κάθε επεξεργασίας,
- Διασφαλίζουν τη μη ύπαρξη «μαύρων τρυπών» κατά την αλληλεπίδραση του σχεδιαζόμενου συστήματος με αρχεία/βάσεις Δεδομένων,
- Διακρίνουν την ύπαρξη «μαύρων τρυπών» κατά τη βελτίωση ενός ήδη υπάρχοντος συστήματος με αρχεία/βάσεις δεδομένων,

- Διορθώνουν την ύπαρξη «μαύρων τρυπών», κατά τη βελτίωση ενός ήδη υπάρχοντος συστήματος με αρχεία/βάσεις δεδομένων,
- Αναλύουν κάθε επεξεργασία ενός Πληροφοριακού Συστήματος σε δομημένες περιγραφές πρώτου και κατώτερων επιπέδων,
- Μετατρέπουν το Διάγραμμα Ροής Δεδομένων σε Μοντέλο Οντοτήτων Συσχετίσεων.

Βασικές έννοιες και λέξεις-κλειδιά

Αντικειμενοστρεφής ανάλυση / Απαιτήσεις / Διάγραμμα μετάβασης καταστάσεων / Διάγραμμα οντοτήτων συσχετίσεων / Διάγραμμα ροής Δεδομένων / Δομημένη ανάλυση / Έγγραφο προδιαγραφών των απαιτήσεων / Λεξικό δεδομένων / Πληροφοριακό Σύστημα / Σχεδίαση / Disaster Recovery and Contingency Plan / RUP / Security Plan.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (3), Σύνολο (3).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΑ ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ
2	ΠΡΟΔΙΑΓΡΑΦΕΣ ΑΠΑΙΤΗΣΕΩΝ
3	ΑΝΑΛΥΣΗ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
4	ΜΟΝΤΕΛΑ ΔΟΜΗΜΕΝΗΣ ΑΝΑΛΥΣΗΣ
5	ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΜΕΘΟΔΟΛΟΓΙΕΣ ΑΝΑΠΤΥΞΗΣ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
6	ΑΣΦΑΛΕΙΑ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Κείμενο αναφοράς

2.4.A.1 | ΕΙΣΑΓΩΓΗ ΣΤΑ ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ

Στη μαθησιακή υποενότητα «Εισαγωγή στα πληροφοριακά συστήματα», οι εκπαιδευόμενοι/-ες θα διδαχθούν τη διαφορά μεταξύ των όρων «Δεδομένα» και «Πληροφορία» και θα εξοικειωθούν με την έννοια του Πληροφοριακού Συστήματος (ΠΣ), παρουσιάζοντας τα χαρακτηριστικά και τα πλεονεκτήματα που παρέχει στις σημερινές επιχειρήσεις και οργανισμούς. Μέσω της υποενότητας, οι εκπαιδευόμενοι/-ες θα εξετάσουν τις συνιστώσες ενός ΠΣ (ανθρώπους, διαδικασίες, λογισμικό-Software, δεδομένα-data, υλικό-Hardware) και τους τομείς/κατηγορίες που περιλαμβάνει κάθε μια από αυτές. Στο πλαίσιο της μαθησιακής υποενότητας, κρίνεται σκόπιμη η αναφορά στον ρόλο του ανθρώπινου παράγοντα στην ανάπτυξη Πληροφοριακών Συστημάτων. Η εν λόγω εκπαιδευτική υποενότητα παρέχει την απαραίτητη βάση για την κατανόηση του τι περιμένουν οι πελάτες/-ισσες και οι χρήστες/-τριες από ένα σύστημα

λογισμικού. Επιπρόσθετα, παρουσιάζεται η ανάγκη αναγνώρισης και κατανόησης των προβλημάτων στη διαχείριση της πληροφορίας και των διαδικασιών στο περιβάλλον μιας σύγχρονης επιχείρησης/οργανισμού. Επίσης, θα εξεταστούν οι μέθοδοι, με τη βοήθεια των οποίων μπορούν να αντιμετωπιστούν οι δυσκολίες που προκαλούν μια προβληματική κατάσταση, ενώ θα δοθούν χρήσιμες πληροφορίες για την αποφυγή αυτών. Στη συνέχεια, γίνεται αναφορά σε σύγχρονα έργα ΠΣ και περιγράφεται η διάκρισή τους, με κριτήρια το μέγεθος (έκταση), την αρχιτεκτονική και το είδος των επιχειρηματικών δραστηριοτήτων που υποστηρίζουν (π.χ. ERP, CRM κ.ά.). Τέλος, παρουσιάζεται η έννοια του υπολογιστικού νέφους (cloud computing) και οι κατηγορίες μοντέλων/μορφών στις οποίες διακρίνεται, δίνοντας έμφαση στο γεγονός ότι η ανάπτυξη του Διαδικτύου έχει επηρεάσει τον τρόπο λειτουργίας των ΠΣ.

2.4.A.2 | ΠΡΟΔΙΑΓΡΑΦΕΣ ΑΠΑΙΤΗΣΕΩΝ

Στο πλαίσιο της μαθησιακής υποενότητας «Προδιαγραφές απαιτήσεων», αναλύεται η σημασία που έχει μια ολοκληρωμένη ανάλυση απαιτήσεων για την ουσιαστική λειτουργία ενός Πληροφοριακού Συστήματος (ΠΣ). Με βάση τα παραπάνω, περιγράφεται η διαδικασία συλλογής πληροφορίας και εξαγωγής των απαιτήσεων, και αναλύονται τα επιθυμητά χαρακτηριστικά που θα πρέπει να συγκεντρώνουν (ορθότητα, συνέπεια, πληρότητα, ρεαλισμό, αναγκαιότητα, επαληθευσσιμότητα). Επίσης, αναφέρεται η έννοια της προδιαγραφής, το πώς σχετίζεται με τον καθορισμό των απαιτήσεων και οι τρόποι με τους οποίους προδιαγράφονται οι τελευταίες. Επιπλέον, με χρήση παραδειγμάτων μέσα από την πραγματική ζωή, επισημαίνεται η διάκριση των απαιτήσεων σε: απαιτήσεις συστήματος και απαιτήσεις λογισμικού, καθώς επίσης και ο τρόπος με τον οποίο γίνεται ο διαχωρισμός αυτός. Στο πλαίσιο της υποενότητας, γίνεται ιδιαίτερη αναφορά στις απαιτήσεις λογισμικού και στην ταξινόμηση αυτών σε λειτουργικές και μη λειτουργικές απαιτήσεις, χρησιμοποιώντας παραδείγματα (με χρήση σεναρίου κατασκευής μιας εφαρμογής) για κάθε κατηγορία ξεχωριστά. Ειδικά για τις μη λειτουργικές απαιτήσεις, κρίνεται σκόπιμος ο διαχωρισμός τους σε απαιτήσεις χρήσης, αξιοπιστίας, επιδόσεων, υποστήριξης, υλοποίησης κ.ά., περιγράφοντας τα χαρακτηριστικά της κάθε ομάδας. Επίσης, οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με τη μορφή και τα συστατικά στοιχεία που έχει το «Έγγραφο Προδιαγραφών των Απαιτήσεων» από το λογισμικό. Συγκεκριμένα, μελετάται η δομημένη και λεπτομερής καταγραφή των απαιτήσεων του λογισμικού, με έμφαση στο ότι όσο πιο εύστοχη είναι η προσπάθεια ανάπτυξης λογισμικού τόσο πιο σταθερό θα είναι το περιβάλλον χρήσης.

2.4.A.3 | ΑΝΑΛΥΣΗ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Στη μαθησιακή υποενότητα «Ανάλυση πληροφοριακών συστημάτων», αναλύεται η σημασία που έχει το στάδιο της ανάλυσης, τονίζοντας ότι η ανάλυση αποτελεί το πρώτο βήμα της σχεδίασης και επισημαίνοντας ότι αποτελεί το «πλάνο», βάσει του οποίου θα σχεδιαστεί και μετέπειτα θα υλοποιηθεί το σύστημα. Στο πλαίσιο της ενό-

τητας, εξετάζονται οι πιο δημοφιλείς μέθοδοι για την ανάπτυξη ΠΣ. Παρουσιάζεται η παραδοσιακή μέθοδος της δομημένης ανάλυσης (η οποία αποτελεί αντικείμενο της μαθησιακής υποεπάρκειας που ακολουθεί), καθώς επίσης και η νεότερη στρατηγική της αντικειμενοστρεφούς ανάλυσης. Επίσης, αναλύονται οι παραλλαγές, τα δυνατότα και αδύνατα σημεία κάθε μεθόδου, τονίζοντας ότι δεν υπάρχει μια συγκεκριμένη στρατηγική ανάπτυξης συστήματος και ότι οι οργανισμοί αναπτύσσουν τις δικές τους προσεγγίσεις ή υιοθετούν μεθόδους που προσφέρονται από προμηθευτές λογισμικού ή συμβούλους. Έχοντας συγκεντρώσει όλες τις απαιτούμενες πληροφορίες από την προηγούμενη μαθησιακή υποεπάρκεια, οι εκπαιδευόμενοι/-ες αναλύουν με δομημένο τρόπο και καταγράφουν την απαιτούμενη λειτουργικότητα και τους περιορισμούς του συστήματος, με σκοπό να περιγράψουν βήμα-βήμα τι ακριβώς θα πρέπει να κάνει το σύστημα (τρέχουσες και μελλοντικές ανάγκες). Οι εκπαιδευόμενοι/-ες εξοικειώνονται με τις απαραίτητες προϋποθέσεις για την ανάλυση ενός ΠΣ, δίνοντας ιδιαίτερη έμφαση στη μοντελοποίηση, δηλαδή την οργάνωση της πληροφορίας μέσω διαγραμμάτων (βλ. επόμενη μαθησιακή υποεπάρκεια). Λαμβάνουν συγκεκριμένες γνώσεις και αναπτύσσουν δεξιότητες, ώστε να διακρίνουν τα είδη και τις σχέσεις εξάρτησης των δρώντων με το περιγραφόμενο σύστημα, καθώς και τις διακριτές μεταξύ τους λειτουργίες εντός του συστήματος. Τέλος, οι εκπαιδευόμενοι/-ες εξοικειώνονται με τα μοντέλα παράστασης λογισμικού, για να περιγράψουν το λογισμικό και να αποδώσουν τις απαιτήσεις με κατανοητό τρόπο. Στόχος είναι η διάκριση των μοντέλων αυτών σε στατικά και δυναμικά μοντέλα και η περιγραφή του τρόπου λειτουργίας τους. Στη μαθησιακή υποεπάρκεια που ακολουθεί, γίνεται αναλυτική περιγραφή των χαρακτηριστικών των παραπάνω μοντέλων.

2.4.A.4 | ΜΟΝΤΕΛΑ ΔΟΜΗΜΕΝΗΣ ΑΝΑΛΥΣΗΣ

Στη μαθησιακή υποεπάρκεια «Μοντέλα δομημένης ανάλυσης», οι εκπαιδευόμενοι/-ες έρχονται σε επαφή με τις κατηγορίες διαγραμμάτων, τα οποία περιγράφουν το λογισμικό από συγκεκριμένη οπτική γωνία. Με τη βοήθεια των διαγραμμάτων, οι εκπαιδευόμενοι/-ες θα διακρίνουν την ύπαρξη «μαύρων τρυπών» και θα τις διορθώσουν, διασφαλίζοντας την πληρότητα του σχεδιαζόμενου συστήματος. Συγκεκριμένα, θα αναλυθεί ο όρος Διαγράμματα Ροής Δεδομένων (ΔΡΔ) (Data Flow Diagram) περιγράφοντας τον τρόπο με τον οποίο τα δεδομένα «ρέουν» και μετασχηματίζονται σε νέα δεδομένα από μονάδες λογισμικού. Αναλύονται τα χαρακτηριστικά ενός ΔΡΔ, οι συμβολισμοί και οι συμβάσεις που χρησιμοποιούνται για την κατασκευή ενός ΔΡΔ μηδενικού επιπέδου. Επίσης, ελέγχεται η ορθότητα ενός ΔΡΔ (αν είναι συντακτικά και σημασιολογικά σωστό) και, με τη χρήση παραδειγμάτων, απεικονίζεται ένα λογισμικό, αποσυνθέτοντάς το σε ΔΡΔ χαμηλότερου επιπέδου (επιπέδου 1, 2 κ.λπ.). Επίσης, αναλύεται ο όρος Διαγράμματα Οντοτήτων-Συσχετίσεων (ΔΟΣ) (Entity-Relationship Diagram). Περιγράφεται τι είναι μια οντότητα και αναλύονται τα χαρακτηριστικά της. Επίσης, με τη χρήση πρακτικών παραδειγμάτων, απεικονίζονται οι οντότητες των δεδομένων που

χειρίζεται η εφαρμογή λογισμικού και οι συσχετίσεις μεταξύ αυτών. Κρίνεται σκόπιμη η ανάλυση της μεθοδολογίας μετατροπής ενός ΔΡΔ σε Μοντέλο Οντοτήτων Συσχετίσεων. Επιπλέον, αναλύεται ο όρος Διαγράμματα Μετάβασης Καταστάσεων (ΔΜΚ) (State Transition Diagram). Οι εκπαιδευόμενοι/-ες εξοικειώνονται με τη χρονική σειρά εκτέλεσης των εντολών, δηλαδή με το πώς γίνεται η μετάβαση από τη μια κατάσταση στην άλλη, ανάλογα με τα εξωτερικά γεγονότα. Επίσης, αναλύονται οι συμβολισμοί, τα χαρακτηριστικά και οι κανόνες κατασκευής ενός ΔΜΚ και, με τη χρήση παραδειγμάτων, απεικονίζεται ένα λογισμικό. Τέλος, αναλύεται ο όρος Λεξικό Δεδομένων (Data Dictionary), επισημαίνοντας ότι περιλαμβάνει όλα τα στοιχεία δεδομένων που σχετίζονται με το σύστημα (π.χ. όνομα του είδους δεδομένων, χρήση, περιγραφή). Στόχος είναι οι εκπαιδευόμενοι/-ες να έρθουν σε επαφή με τα εργαλεία υποστήριξης ανάπτυξης λογισμικού (CASE), για τη δημιουργία ενός λεξικού δεδομένων.

2.4.A.5 | ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΜΕΘΟΔΟΛΟΓΙΕΣ ΑΝΑΠΤΥΞΗΣ

ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Σε αυτή τη μαθησιακή υποενότητα, αναλύεται ο τρόπος με τον οποίο πραγματοποιείται η σχεδίαση ενός έργου, βασιζόμενου στο «Έγγραφο προδιαγραφών των απαιτήσεων». Οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με τον τρόπο που θα συμπεριφέρεται το σύστημα κατά τη φάση της υλοποίησης, αλλά και τις σχέσεις που έχουν τα διάφορα υποσυστήματα μεταξύ τους. Επισημαίνεται ότι, μέσα από τη σχεδίαση, περιγράφεται το σύστημα γενικά και, με σταδιακή εκλέπτυνση, περιγράφεται η εσωτερική δομή των μονάδων σε τέτοιο σημείο, που να μπορούν να υλοποιηθούν με τη χρήση μιας γλώσσας προγραμματισμού. Αναλύεται τι σημαίνει κακή σχεδίαση και το γεγονός ότι, αν αυτή δεν αντιμετωπιστεί σε αυτό το στάδιο, μπορεί να οδηγήσει σε ένα «προβληματικό» σύστημα. Επίσης, παρουσιάζονται οι πιο γνωστές προσεγγίσεις/μοντέλα του Κύκλου Ζωής Ανάπτυξης Συστήματος (ΚΣΑΖ). Στην παρούσα μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες έρχονται σε επαφή με τις μεθοδολογίες SSADM (Structured System Analysis Design Method), RUP (Rational Unified Process) και Scrum, αναλύοντας τα χαρακτηριστικά, τα πλεονεκτήματα και τα μειονεκτήματα κάθε μιας από αυτές. Συγκεκριμένα, αναλύονται τα είδη των Πληροφοριακών Συστημάτων που στοχεύει η κάθε μεθοδολογία, τα εργαλεία ανάλυσης μοντελοποίησης και σχεδιασμού που χρησιμοποιούνται και τα παραδοτέα που παράγονται σε κάθε περίπτωση.

2.4.A.6 | ΑΣΦΑΛΕΙΑ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Η υποενότητα αυτή εξετάζει τη σημασία της προστασίας ενός Πληροφοριακού Συστήματος. Οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με τα είδη των κινδύνων που απειλούν τα δεδομένα ενός ΠΣ (όπως αστοχία υλικού και λογισμικού, ανθρώπινο σφάλμα, ιοί, κακόβουλες επιθέσεις, φυσικές καταστροφές). Επίσης, θα λάβουν γνώσεις και θα αναπτύξουν δεξιότητες για την αντιμετώπιση των απειλών, χρησιμοποιώντας τεχνικές, οι οποίες έχουν τη μικρότερη δυνατή επίπτωση στην αποδοτικότητα

τα του συστήματος. Συγκεκριμένα, αναλύονται οι τρεις βασικές αρχές για τη σωστή και απρόσκοπτη λειτουργία ενός ΠΣ, οι οποίες αναφέρονται στην εμπιστευτικότητα (confidentiality), την ακεραιότητα (integrity) και τη διαθεσιμότητα (availability). Στο πλαίσιο της υποενότητας, περιγράφεται η έννοια της πολιτικής ασφάλειας, επισημαίνοντας τον σκοπό, τα χαρακτηριστικά της και τις περιοχές τις οποίες καλύπτει (π.χ. πολιτική προστασίας πληροφορίας, πολιτική φυσικής ασφάλειας, πολιτική διαχείρισης χρηστών, πολιτική προστασίας από ιούς, πολιτική συνέχειας επιχειρηματικών λειτουργιών). Επίσης, κρίνεται σκόπιμη η αναφορά στον όρο «μέτρα ασφάλειας», εντάσσοντας μέσα σε αυτά τις κατηγορίες των οργανωτικών μέτρων ασφάλειας, των τεχνικών μέτρων ασφάλειας και των μέτρων φυσικής ασφάλειας, αναλύοντας τις υποκατηγορίες από τις οποίες αποτελούνται. Τέλος, αναφέρεται η έννοια του Σχεδίου Ασφάλειας (Security Plan) και του Σχεδίου Ανάκαμψης από Καταστροφές (Disaster Recovery and Contingency Plan), αναλύοντας τον σκοπό, τα χαρακτηριστικά και τις ενέργειες που προβλέπουν.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Hoffer, J., George J. & Valacich, J. (2011). *Πληροφοριακά συστήματα. Σύγχρονη Ανάλυση & Σχεδίαση*. Θεσσαλονίκη: Εκδόσεις Τζιόλα.
2. Langer, A. M. (2012). *Guide to software development: Designing and managing the life cycle*. New York: Springer Science & Business Media.

Συμπληρωματικές

1. Information system-Definition, examples & facts. (2001, October 12). Retrieved from: <https://www.britannica.com/topic/information-system>
2. Rajaraman, V. (2004). *Analysis and design of information systems*. PHI Learning Pvt.

2.4.B • ΤΕΧΝΟΛΟΓΙΑ ΛΟΓΙΣΜΙΚΟΥ

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Τεχνολογία λογισμικού» εισάγει τους/τις εκπαιδευόμενους/-ες στον κύκλο ζωής του λογισμικού. Επιπλέον, παρουσιάζονται όλα τα θεωρητικά χαρακτηριστικά της μοντελοποίησης (περιπτώσεις χρήσης, ροής και αποθήκευσης δεδομένων καθώς και αποτύπωσης των ρόλων των εμπλεκομένων [stakeholders]), μαζί με μια αναλυτική παρουσίαση της γλώσσας μοντελοποίησης UML. Επίσης, παρουσιάζονται τα πιο διαδεδομένα μοντέλα κύκλου ζωής λογισμικού, όπως το μοντέλο του Καταρράκτη και οι παραλλαγές αυτού (παράλληλη ανάπτυξη, V-model),

άλλα επαναληπτικά και επαυξητικά μοντέλα (Σπειροειδές κ.ά.), ενώ διδάσκεται και η μέθοδος Agile. Επιπροσθέτως, ο/η εκπαιδευόμενος/-η διδάσκεται τεχνικές συγγραφής ποιοτικού κώδικα, μορφοποίησης και τεκμηρίωσης κώδικα, αποτελεσματικές μεθόδους για τη διαχείριση έργων (Τεχνική PERT), την κατασκευή διαγραμμάτων PERT και Gantt, την έννοια του κρίσιμου μονοπατιού, τεχνικές διαχείρισης κινδύνων και τεχνικές ελέγχου λογισμικού (επαλήθευση και επικύρωση), με εργαλεία testing (automation, SOAP services, web UI, performance, Security). Τέλος, ο/η εκπαιδευόμενος/-η διδάσκεται μεθόδους και μηχανισμούς ελέγχου της ποιότητας λογισμικού (μετρικές λογισμικού, δένδρα απόφασης), παρέχοντας τεκμηρίωση κατά την παράδοση κάθε έργου, ελέγχους εκδόσεων και συντήρησης του λογισμικού.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναγνωρίζουν τα βασικά στοιχεία της τεχνολογίας του λογισμικού,
- Κατονομάζουν εργαλεία ανάλυσης, σχεδίασης, μεθόδων υλοποίησης, ελέγχων και συντήρησης/διαχείρισης της ανάπτυξης του λογισμικού,
- Απαριθμούν εργαλεία διασφάλισης της ποιότητας του λογισμικού,
- Αναπτύσσουν οδηγούς τεκμηρίωσης και άλλα έγγραφα για τις εφαρμογές που δημιουργούν (π.χ. Administrative Manual, Manual τελικών χρηστών/-τριών, Business Requirements document, Infrastructure Requirements document, User Interface Specifications document, Test Scenarios document, Test Case document κ.λπ.),
- Δοκιμάζουν σε κατάλληλο περιβάλλον τις εφαρμογές που υλοποιούν, πριν αυτές χρησιμοποιηθούν στην παραγωγική διαδικασία-production (π.χ. functional testing, service interface testing, installation testing, UI testing, system integration testing, disaster recovery testing, Hardware and environment testing, performance testing, security testing, regression testing κ.λπ.),
- Υιοθετούν θετική στάση ως προς διάφορα εργαλεία που μπορούν να χρησιμοποιηθούν στην τεχνολογία λογισμικού.

Βασικές έννοιες και λέξεις-κλειδιά

Αρχιτεκτονικά διαγράμματα/Δομικά διαγράμματα / Διαγράμματα συμπεριφοράς / Έλεγχος λογισμικού / Κύκλος ζωής λογισμικού / Μετρικές λογισμικού/Ποιότητα λογισμικού / Agile model / Incremental model / ISO 9126 / PERT / UML / Waterfall model.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (2), Εργαστήριο (2), Σύνολο (4).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΚΥΚΛΟΣ ΖΩΗΣ ΑΝΑΠΤΥΞΗΣ ΣΥΣΤΗΜΑΤΟΣ ΚΑΙ ΜΟΝΤΕΛΑ ΥΛΟΠΟΙΗΣΗΣ
2	ΓΛΩΣΣΑ ΜΟΝΤΕΛΟΠΟΙΗΣΗΣ UML
3	ΔΙΑΧΕΙΡΙΣΗ ΕΡΓΩΝ
4	ΕΛΕΓΧΟΣ ΛΟΓΙΣΜΙΚΟΥ
5	ΠΟΙΟΤΗΤΑ ΛΟΓΙΣΜΙΚΟΥ
6	ΜΕΤΡΗΣΗ ΠΟΙΟΤΗΤΑΣ ΛΟΓΙΣΜΙΚΟΥ

Κείμενο αναφοράς

2.4.B.1 | ΚΥΚΛΟΣ ΖΩΗΣ ΑΝΑΠΤΥΞΗΣ ΣΥΣΤΗΜΑΤΟΣ ΚΑΙ ΜΟΝΤΕΛΑ ΥΛΟΠΟΙΗΣΗΣ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες έρχονται σε επαφή με τις θεμελιώδεις έννοιες της τεχνολογίας λογισμικού και του κύκλου ζωής ενός συστήματος λογισμικού. Αναλύεται η σημασία του λογισμικού καλής ποιότητας και ποιες είναι οι δυσκολίες που εμφανίζονται κατά τη διαδικασία παραγωγής του. Επίσης, αναλύονται οι φάσεις από τις οποίες αποτελείται ο κύκλος ζωής ανάπτυξης ενός συστήματος και ο τρόπος με τον οποίο αυτές αλληλεπιδρούν μεταξύ τους. Συγκεκριμένα, γίνεται εκτεταμένη ανάλυση στις φάσεις της ανάλυσης, του σχεδιασμού, της υλοποίησης και της λειτουργίας-συντήρησης. Ερμηνεύεται ο σκοπός και η σημασία κάθε φάσης στην ανάπτυξη του λογισμικού, ποιες εργασίες γίνονται και ποια προϊόντα παράγονται σε κάθε μια. Επιπλέον, μελετάται ο τρόπος με τον οποίο διαδέχονται η μια την άλλη, σε κάθε ένα από τα γνωστά μοντέλα υλοποίησης. Στο πλαίσιο της υποενότητας, δίνεται ιδιαίτερη έμφαση στο γεγονός ότι, ανάλογα με τις απαιτήσεις του έργου (ανάπτυξης λογισμικού), γίνεται να χρησιμοποιηθούν διαφορετικές προσεγγίσεις. Επιπρόσθετα, παρουσιάζονται οι πιο γνωστές προσεγγίσεις του Κύκλου Ζωής Ανάπτυξης Συστήματος (ΚΣΑΖ). Οι προσεγγίσεις που παρουσιάζονται αφορούν το μοντέλο του καταρράκτη (waterfall model) με τις παραλλαγές του, το μοντέλο του πρωτοτύπου (prototype model), το επαυξητικό μοντέλο (incremental model), το επαναληπτικό μοντέλο (iterative model), το σπειροειδές μοντέλο (spiral model) και το ευέλικτο μοντέλο (agile model). Για κάθε μια από αυτές, εξετάζεται ο βαθμός προσαρμοστικότητας και ευελιξίας, και αναλύονται τα πλεονεκτήματα και μειονεκτήματα που εμφανίζουν συγκριτικά.

2.4.B.2 | ΓΛΩΣΣΑ ΜΟΝΤΕΛΟΠΟΙΗΣΗΣ UML

Στην υποενότητα αυτή, οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με την έννοια της αντικειμενοστρεφούς προσέγγισης. Η εν λόγω υποενότητα εστιάζεται στην κατανόηση του ότι οποιαδήποτε οντότητα του πραγματικού κόσμου μπορεί να παρασταθεί ως αντικείμενο (με έναν πραγματιστικό τρόπο), το οποίο μπορεί να μοντελοποιηθεί με

ιδιότητες (attributes) και μεθόδους (methods)/λειτουργίες (operations) που ανήκουν σε αυτό. Αναλύονται τα βασικά χαρακτηριστικά ενός αντικειμένου (κατάσταση-state, ταυτότητα-identity, συμπεριφορά-behavior) και γίνεται αναφορά στην έννοια της κλάσης/τάξης (class) αντικειμένων και τα χαρακτηριστικά αυτής. Στο πλαίσιο της υποενότητας, οι εκπαιδευόμενοι/-ες θα γνωρίσουν την αποσύνθεση σε αντικείμενα και θα μελετήσουν τα πλεονεκτήματα αυτής, τα οποία οδηγούν σε σχέδια λογισμικού που μπορούν να μοντελοποιηθούν, με τη χρήση της γλώσσας UML (Unified Modeling Language). Σκοπός της υποενότητας είναι η ανάλυση των βασικών στοιχείων της γλώσσας UML (οντότητες, σχέσεις και διαγράμματα) και η εξέταση των τρόπων παράστασης ορισμένων αντικειμενοστρεφών δομών, χρησιμοποιώντας ορισμένες κατηγορίες διαγραμμάτων της UML. Συγκεκριμένα, εξετάζονται τα διαγράμματα συμπεριφοράς ή δυναμικά διαγράμματα, τα οποία περιγράφουν τη δυναμική συμπεριφορά ενός συστήματος και διακρίνονται σε διαγράμματα περιπτώσεων χρήσης (use case diagrams), διαγράμματα δραστηριοτήτων (activity diagrams), διαγράμματα καταστάσεων (state machine diagrams), διαγράμματα ακολουθίας (sequence diagrams) και διαγράμματα επικοινωνίας (communication diagrams). Επίσης, εξετάζονται τα δομικά διαγράμματα, τα οποία περιγράφουν την εσωτερική λογική δομή ενός συστήματος και διακρίνονται σε διαγράμματα πακέτων (package diagrams), διαγράμματα αντικειμένων (object diagrams), διαγράμματα κλάσεων (class diagrams). Τέλος, εξετάζονται τα αρχιτεκτονικά διαγράμματα ή διαγράμματα υλοποίησης, τα οποία περιγράφουν τη φυσική δομή ενός συστήματος και διακρίνονται σε διαγράμματα συνεργασίας (collaboration diagrams), διαγράμματα ανάπτυξης (deployment diagrams) και διαγράμματα συστατικών (component diagrams).

2.4.B.3 | ΔΙΑΧΕΙΡΙΣΗ ΕΡΓΩΝ

Τα θέματα που θα παρουσιαστούν στη συγκεκριμένη υποενότητα αφορούν τον σκοπό και τη χρήση της τεχνικής PERT (Program Evaluation and Review Technique). Οι εκπαιδευόμενοι/-ες εξοικειώνονται με τα πλεονεκτήματα της συγκεκριμένης μεθόδου για τη διαχείριση έργων Πληροφορικής (και όχι μόνο), τα οποία εστιάζονται στην κατάρτιση ενός μεγάλου έργου σε τυπικά υποέργα (ή δραστηριότητες), στη δημιουργία ενός χρονοδιαγράμματος για την ολοκλήρωση του έργου, στην καταγραφή των εκτιμήσεων για τις αισιόδοξες και τις απαισιόδοξες προβλέψεις της χρονικής διάρκειας για κάθε ένα από τα τυπικά υποέργα (τεχνική εκτίμησης LOC-Lines Of Code) και στον υπολογισμό της πιθανότητας ολοκλήρωσης του έργου μέσα σε συγκεκριμένα χρονικά περιθώρια. Στο πλαίσιο της υποενότητας, οι εκπαιδευόμενοι/-ες θα ασχοληθούν με την κατασκευή διαγραμμάτων PERT και Gantt, ερμηνεύοντας τα κύρια χαρακτηριστικά τους, όπως αυτά του κρίσιμου σημείου (milestone) και του κρίσιμου μονοπατιού. Θα παρουσιαστούν οι έννοιες του αναμενόμενου ή εκτιμώμενου χρόνου και της διακύμανσης μιας δραστηριότητας. Επιπλέον, με τη χρήση πρακτικών παραδειγμάτων, θα διδαχθεί ο τρόπος με τον οποίο γίνεται ο υπολογισμός των παραπάνω χαρα-

κτηριστικών. Επίσης, κρίνεται σκόπιμη η αναφορά στον στατιστικό πίνακα κανονικής κατανομής και η περιγραφή του τρόπου με τον οποίο χρησιμοποιείται για την εύρεση της πιθανότητας ολοκλήρωσης του έργου σε συγκεκριμένο χρόνο, αλλά και την εύρεση του χρόνου ολοκλήρωσης ενός έργου δοσμένης της πιθανότητας. Τέλος, μέσω της υποεπένδυσης, γίνεται αναφορά στις έννοιες του ανθρωπομήνα, του ανθρώπινου δυναμικού και του τι είναι και πώς χρησιμοποιείται το διάγραμμα ανάθεσης σε ανθρώπινο δυναμικό.

2.4.B.4 | ΕΛΕΓΧΟΣ ΛΟΓΙΣΜΙΚΟΥ

Στη μαθησιακή υποεπένδυση «Ελέγχος λογισμικού» ελέγχονται συγκεκριμένα χαρακτηριστικά του λογισμικού και διερευνάται το κατά πόσο οι ατέλειες, που είναι λογικό να υπάρχουν, επηρεάζουν την ποιότητα του λογισμικού, ιδιαίτερα στο κομμάτι της ασφάλειας, της αξιοπιστίας και της ευχρηστίας. Οι εκπαιδευόμενοι/-ες εξοικειώνονται με τα διαφορετικά επίπεδα ελέγχου λογισμικού (levels of testing), αναλύοντας το είδος των δοκιμών που περιλαμβάνονται σε κάθε ένα από αυτά. Συγκεκριμένα, παρουσιάζονται δοκιμές που αφορούν ελέγχους μονάδων (Unit Testing), ελέγχους ενσωμάτωσης (Integration Testing) και ελέγχους του συστήματος (System Testing). Επίσης, γίνεται ιδιαίτερη αναφορά στους ελέγχους αποδοχής (Acceptance Testing), αλλά και στους επιμέρους ελέγχους στους οποίους αυτοί διακρίνονται (π.χ. user acceptance tests, operational acceptance tests, contact and regulation tests, alpha & beta tests). Στο πλαίσιο της υποεπένδυσης, μέσα από παραδείγματα ελέγχων λογισμικού, αναλύονται οι έννοιες και ο σκοπός των ελέγχων καταπόνησης (Stress testing), αντοχής (Endurance testing) και κλιμάκωσης (Scalability testing). Για όλους τους παραπάνω ελέγχους, οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με τους όρους functional testing, service interface testing, installation testing, UI testing, system integration testing, disaster recovery testing, Hardware and environment testing, performance testing, security testing, regression testing, αντιλαμβανόμενοι/-ες τη σημασία που έχουν, πριν το λογισμικό χρησιμοποιηθεί στην παραγωγική διαδικασία, δηλαδή πριν κυκλοφορήσει στην αγορά. Τέλος, αναλύονται οι διαφορετικές κατηγορίες ελέγχων δοκιμών του λογισμικού, χρησιμοποιώντας διάφορες τεχνικές και επισημαίνοντας τον τρόπο με τον οποίο χρησιμοποιούνται (και τις διαφορές μεταξύ τους). Κρίνεται σκόπιμη η παρουσίαση των τεχνικών του μαύρου κουτιού (black box technique), του άσπρου κουτιού (white box technique) και του γκριζού κουτιού (grey box technique).

2.4.B.5 | ΠΟΙΟΤΗΤΑ ΛΟΓΙΣΜΙΚΟΥ

Σε αυτή τη μαθησιακή υποεπένδυση, οι εκπαιδευόμενοι/-ες εξοικειώνονται με τη σημασία που έχει η ποιότητα του λογισμικού τόσο για τους χρήστες/-τριες όσο και για τις επιχειρήσεις. Αναφέρεται τι κάνει το λογισμικό υψηλής ποιότητας και παρουσιάζονται οι τρόποι με τους οποίους οι εκπαιδευόμενοι/-ες μπορούν να βεβαιωθούν ότι ένα λογισμικό είναι όντως υψηλής ποιότητας. Στο πλαίσιο της υποεπένδυσης, οι εκ-

παιδευόμενοι/-ες θα έρθουν σε επαφή με τις μεθόδους μέσα από τις οποίες μπορούν να πετύχουν ποιότητα λογισμικού. Συγκεκριμένα, γίνεται αναφορά στον ρόλο των εγγράφων, όπως των Business Requirements document, Infrastructure Requirements document, User Interface Specifications document, στα οποία καταγράφονται οι απαιτήσεις του/της πελάτη/-ισσας και χρησιμοποιούνται ως συμφωνία ή ως βάση με το τι θα κάνει το λογισμικό. Επίσης, δίνεται έμφαση στη σημασία που έχει ο σχεδιασμός στη διασφάλιση της αξιοπιστίας του λογισμικού. Επιπλέον, αναλύεται η μέθοδος της αναθεώρησης του πηγαιού κώδικα, πραγματοποιώντας διαδικασίες ελέγχου και κατάλληλα εργαλεία. Μέσα από την υποενότητα, οι εκπαιδευόμενοι/-ες εξοικειώνονται με τη χρήση συστημάτων ελέγχου εκδόσεων (version control system) και συστημάτων ανίχνευσης σφαλμάτων (bugs). Τέλος, αναφέρεται η σημασία της ανάπτυξης οδηγών τεκμηρίωσης (π.χ. Administrative Manual, Manual τελικών χρηστών/-τριών). Η εκπαιδευτική υποενότητα εστιάζεται στον ρόλο που διαδραματίζουν γνωστά πρότυπα από Διεθνείς Οργανισμούς Προτύπων (πχ ISO 9001, ISO 9000-3, CMM) για τον καθορισμό της ποιότητας ενός λογισμικού. Σκόπιμη κρίνεται η αναλυτική περιγραφή του προτύπου ISO 9126, το οποίο θεωρείται το πλέον σημαντικό και αφορά την ποιότητα λογισμικού. Επομένως, όσον αφορά το πρότυπο ISO 9126, αναλύονται με λεπτομέρεια οι διαφορετικές κατηγορίες χαρακτηριστικών (λειτουργικότητα, ευχρηστία, αποδοτικότητα, αξιοπιστία, συντηρησιμότητα, μεταφερισιμότητα), στις οποίες διακρίνονται και τα επιμέρους χαρακτηριστικά κάθε μιας από αυτές. Τέλος, μέσα από πρακτικά παραδείγματα, π.χ. μέσω της σύγκρισης δυο ιστοτόπων, θα εξεταστεί το κάθε ένα από τα περιγραφόμενα χαρακτηριστικά.

2.4.B.6 | ΜΕΤΡΗΣΗ ΠΟΙΟΤΗΤΑΣ ΛΟΓΙΣΜΙΚΟΥ

Στη μαθησιακή υποενότητα «Μέτρηση ποιότητας λογισμικού», γίνεται αναφορά στον προσδιορισμό και τη χρησιμότητα έγκυρων μετρήσεων που χρησιμοποιούνται ως μηχανισμοί ελέγχου της ποιότητας λογισμικού, σε τομείς που περιλαμβάνουν το μέγεθος, την πολυπλοκότητα και το κόστος. Για την καλύτερη εμπέδωση των στόχων της μαθησιακής υποενότητας, αναφέρεται ο τρόπος με τον οποίο γίνεται ποιοτική σύγκριση δύο προγραμμάτων που εκτελούν την ίδια διαδικασία. Υπό το πρίσμα του παραπάνω ζητήματος, οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με την έννοια της μετρικής των εσωτερικών και εξωτερικών χαρακτηριστικών λογισμικού. Εξοικειώνονται με τις εξωτερικές και εσωτερικές μετρικές λογισμικού, δίνοντας ιδιαίτερη έμφαση στις τελευταίες, και αναλύεται πώς η εφαρμογή τους παρέχει το πλεονέκτημα στους μηχανικούς ανάπτυξης του συστήματος να αξιολογήσουν το λογισμικό πριν δοθεί προς χρήση. Πιο συγκεκριμένα, παρουσιάζονται οι (ποσοτικές) μετρικές του Halstead, και εξηγείται ο τρόπος με τον οποίο οι εν λόγω μετρικές μπορούν να μετρήσουν αριθμήσιμα στοιχεία του λογισμικού, σχετιζόμενα με το μέγεθος του πηγαιού κώδικα. Στη συνέχεια, αναλύονται τα δέντρα απόφασης (decision tree) και η πολύτιμη βοήθεια την οποία προσφέρουν στη λήψη της καλύτερης απόφασης, όταν χρειάζεται

να γίνει επιλογή μεταξύ διαφορετικών ενδεχομένων που αφορούν κόστη. Μέσα από πραγματικά παραδείγματα της ανθρώπινης δραστηριότητας, η εν λόγω υποενότητα παρουσιάζει τον τρόπο με τον οποίο κατασκευάζεται ένα δέντρο απόφασης. Έτσι, αναλύονται τα βασικά είδη κόμβων (κόμβος απόφασης, κόμβος πιθανοτήτων, τελικός κόμβος) και τι αναπαριστά το κάθε ένα από αυτά. Τέλος, οι εκπαιδευόμενοι/-ες θα έρθουν σε επαφή με την έννοια EMV (Expected Monetary Value-αναμενόμενο κέρδος) και μαθαίνουν να υπολογίζουν την τιμή του χαρακτηριστικού αυτού για κάθε κόμβο του δέντρου απόφασης.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Bennett, S., Farmer, R. & McRobb, S. (2010). *Object-oriented systems analysis and design using UML*. UK Higher Education Computing Computer Science.
2. Blokdik, G. (2019). *Software development life cycle SDLC a complete guide-2020 edition*. 5starcooks.

Συμπληρωματικές

1. Kaner, C., Bach, J. & Pettichord, B. (2001). *Lessons learned in software testing: A context-driven approach*. New Jersey: Wiley & Sons.
2. Ξένος, Μ. (2021). *Ποιότητα Λογισμικού*. Αθήνα: Εκδόσεις Γκότση.

2.4.Γ • ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ IV (ASP.NET, MVC, C#10)- ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΕΦΑΡΜΟΓΩΝ ΓΡΑΦΙΚΗΣ ΔΙΕΠΑΦΗΣ ΧΡΗΣΤΗ (GUI) II

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Γλώσσα προγραμματισμού IV (ASP.NET, MVC, C#10)-Προγραμματισμός εφαρμογών γραφικής διεπαφής χρήστη (GUI) II» εισάγει τους/τις εκπαιδευόμενους/-ες στη λογική του προγραμματισμού με το Framework ASP.NET, μέσω της γλώσσας προγραμματισμού C#10. Η συγκεκριμένη μαθησιακή ενότητα είναι επηρεασμένη από πολλές μαθησιακές ενότητες όλων των εξαμήνων κατάρτισης, αλλά κατά βάση αντλεί στοιχεία από τη μαθησιακή ενότητα «Γλώσσα Προγραμματισμού II (C++14)-Αντικειμενοστρεφής προγραμματισμός» του εξαμήνου Β΄ και από τη μαθησιακή ενότητα «Γλώσσα προγραμματισμού III (JAVA9/JAVAFX)-Προγραμματισμός εφαρμογών γραφικής διεπαφής χρήστη (GUI) I» του εξαμήνου Γ΄. Στην παρούσα μαθησιακή ενότητα, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με προχωρημένα θέματα αντικειμενοστρεφούς προγραμματισμού, όπως αυτά των δομών επιλογής, επανάληψης, συναρτήσεων, εξαιρέσεων και υπερφόρτωσης, της δημιουργίας κλάσεων και αντικειμένων, της χρήσης μεθόδων κληρονομικότητας και πολυμορφι-

σμού, μέσω των ασφαλών τεχνικών ενθυλάκωσης των πεδίων κάθε αντικειμένου, δημιουργώντας έτσι πιο ανθεκτικές και πιο ασφαλείς εφαρμογές γραφικού περιβάλλοντος μέσω GUI APIs, βάσει του μοντέλου Model View Control (MVC). Με βάση τα παραπάνω, οι εφαρμογές αυτές θα επεξεργάζονται πολυμεσικά αρχεία ή/και θα συνδέονται με βάση δεδομένων, δημιουργώντας ολοκληρωμένες εφαρμογές.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναπτύσσουν εφαρμογές κονσόλας ή/και εφαρμογές γραφικού περιβάλλοντος με χρήση γλώσσας προγραμματισμού C#,
- Συνεισφέρουν με πρωτότυπο κώδικα στην κοινότητα των προγραμματιστών εφαρμογών,
- Τροποποιούν υπάρχοντα κώδικα, ανάλογα με τις προδιαγραφές των GUI εφαρμογών που αναπτύσσουν,
- Αποφασίζουν για τη χρήση ή μη κατάλληλων βιβλιοθηκών, σχετικών με την ανάπτυξη εφαρμογών,
- Συνεργάζονται με άλλους/-ες εκπαιδευόμενους/-ες, συγκροτώντας ομάδες, με σκοπό τη δημιουργία και την επίλυση προβλημάτων των εφαρμογών GUI,
- Υιοθετούν θετική στάση ως προς τις υπάρχουσες τεχνολογίες ανάπτυξης εφαρμογών GUI,
- Αναγνωρίζουν τις βασικές προγραμματιστικές αρχές του αντικειμενοστρεφούς προγραμματισμού, βάσει των οποίων έχει υλοποιηθεί μια εφαρμογή C#,
- Επιλέγουν τα κατάλληλα μέλη κλάσεων (πεδία, μεθόδους κ.ά.) που θα χρησιμοποιήσουν στις εφαρμογές που θα αναπτύσσουν,
- Αναγνωρίζουν τη σημασία του πολυμορφισμού και της κληρονομικότητας.

Βασικές έννοιες και λέξεις-κλειδιά

Visual C#10.NET / .NET 6 / ASP.NET / MVC Architecture / Object-Oriented Programming / Windows Forms / Custom Controls / Entity Framework / LINQ Security / Deployment.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (4), Σύνολο (4).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΗ C#10 ΚΑΙ ΣΤΟ .NET 6 FRAMEWORK
2	ΚΑΤΑΝΟΗΣΗ ΤΟΥ ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΟΥΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΣΤΗ C#10
3	ΕΞΕΡΕΥΝΗΣΗ ΤΗΣ ΣΥΝΤΑΞΗΣ ΚΑΙ ΤΩΝ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ ΤΗΣ C#10
4	ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ GUI ΜΕ ΤΑ WINDOWS FORMS
5	ΠΡΟΗΓΜΕΝΕΣ ΕΝΝΟΙΕΣ GUI: ΠΡΟΣΑΡΜΟΣΜΕΝΑ ΣΤΟΙΧΕΙΑ ΕΛΕΓΧΟΥ ΚΑΙ ΓΡΑΦΙΚΑ
6	ΕΙΣΑΓΩΓΗ ΣΤΗΝ ASP.NET ΚΑΙ ΣΤΗΝ ΑΡΧΙΤΕΚΤΟΝΙΚΗ MVC
7	ΔΗΜΙΟΥΡΓΙΑ ΕΦΑΡΜΟΓΗΣ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ASP.NET MVC-ΕΡΓΑΣΙΑ ΜΕ ΔΕΔΟΜΕΝΑ: ENTITY FRAMEWORK ΚΑΙ LINQ
8	ASP.NET MVC ROUTING, CONTROLLERS, VIEWS, LAYOUT-ΑΣΦΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ΕΦΑΡΜΟΓΩΝ ASP.NET MVC

Κείμενο αναφοράς

2.4.Γ.1 | ΕΙΣΑΓΩΓΗ ΣΤΗ C#10 ΚΑΙ ΣΤΟ .NET 6 FRAMEWORK

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τα εισαγωγικά στοιχεία της αντικειμενοστρεφούς γλώσσας προγραμματισμού C#10 στο περιβάλλον .NET 6. Οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με την εγκατάσταση και τη ρύθμιση του περιβάλλοντος ανάπτυξης (IDE-Integrated Development Environment), το οποίο προτείνεται να είναι το Microsoft Visual Studio 2019 ή νεότερο (ή κάποιο παρεμφερές αντίστοιχο IDE), και με τις βασικές έννοιες της γλώσσας, όπως τους τύπους δεδομένων, τις μεταβλητές και τις λειτουργίες τους. Αυτή η υποενότητα θα παρουσιάσει στους/στις εκπαιδευόμενους/-ες την αρχιτεκτονική της γλώσσας C#10 και της πλατφόρμας .NET 6, παρέχοντας μια επισκόπηση των κύριων χαρακτηριστικών και των δυνατοτήτων τους. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τους τρόπους με τους οποίους θα εγκαταστήσουν και θα ρυθμίσουν το IDE, ενώ θα μάθουν πώς να δημιουργήσουν και να εκτελέσουν απλά προγράμματα σε C#10, και πώς να χρησιμοποιήσουν τις βασικές δομές δεδομένων και τις λειτουργίες της γλώσσας προγραμματισμού C#10. Τέλος, οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με την αλληλεπίδρασή τους με το IDE στο οποίο θα εργάζονται, και με τη διαχείριση των σφαλμάτων και την αποσφαλμάτωση των προγραμμάτων τους. Στο τέλος αυτής της μαθησιακής υποενότητας, οι εκπαιδευόμενοι/-ες θα έχουν μια καλή γνώση των βασικών εννοιών της γλώσσας προγραμματισμού C#10 και της .NET 6, και θα είναι έτοιμοι/-ες να προχωρήσουν σε πιο προηγμένα θέματα προγραμματισμού.

2.4.Γ.2 | ΚΑΤΑΝΟΗΣΗ ΤΟΥ ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΟΥΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΣΤΗ C#10

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα εξερευνήσουν τις βασικές έννοιες του αντικειμενοστρεφούς προγραμματισμού (OOP) στη γλώσσα προ-

γραμματισμού C#10. Αυτή η υποεπενότητα θα παρουσιάσει την ιδέα της κλάσης (class) και των γνωρισμάτων της, που έχουν ήδη διδαχθεί σε προγενέστερες αντικειμενοστρεφείς γλώσσες προγραμματισμού, όπως η γλώσσα προγραμματισμού C++14, η γλώσσα προγραμματισμού JAVA9 και η γλώσσα προγραμματισμού Python3. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τον συντακτικό τρόπο με τον οποίο θα ορίζουν κλάσεις, πώς να ορίζουν τα μέλη κάθε κλάσης και πώς να δημιουργούν αντικείμενα από αυτές τις κλάσεις στη γλώσσα προγραμματισμού C#10. Επιπλέον, θα εξεταστεί η έννοια της κληρονομικότητας (inheritance), η οποία επιτρέπει τη δημιουργία νέων κλάσεων που επεκτείνουν ή τροποποιούν τη συμπεριφορά των υπαρχουσών κλάσεων. Ο, ήδη γνωστός ως έννοια, πολυμορφισμός (polymorphism), μια άλλη κεντρική έννοια του OOP, θα παρουσιαστεί ως ένας τρόπος για την αλλαγή της συμπεριφοράς των αντικειμένων με βάση την κλάση τους. Τέλος, οι εκπαιδευόμενοι/-ες θα διδαχθούν τον τρόπο πραγματοποίησης της απόκρυψης δεδομένων (encapsulation) ενός αντικειμένου που έχει δηλωθεί στη γλώσσα προγραμματισμού C#10, εξασφαλίζοντας την ακεραιότητα και την ασφάλεια των ευαίσθητων πεδίων του. Στο τέλος αυτής της μαθησιακής υποεπενότητας, οι εκπαιδευόμενοι/-ες θα έχουν αποκτήσει ισχυρές βάσεις γύρω από τον OOP στη C#10 και θα είναι έτοιμοι/-ες να εφαρμόσουν αυτές τις έννοιες στα projects τους.

2.4.Γ.3 | ΕΞΕΡΕΥΝΗΣΗ ΤΗΣ ΣΥΝΤΑΞΗΣ ΚΑΙ ΤΩΝ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ ΤΗΣ C#10

Σε αυτή τη μαθησιακή υποεπενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν επιπλέον στοιχεία, σχετικά με τη σύνταξη της C#10, καθώς επίσης και τα γενικά χαρακτηριστικά της. Αυτή η υποεπενότητα παρουσιάζει τις εκφράσεις λάμδα (anonymous functions ή lambda expressions), τα generics, τα exceptions (try, catch, throw) και τα attributes. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τους τρόπους με τους οποίους θα χρησιμοποιούν αυτά τα χαρακτηριστικά, για να γράφουν κώδικα πιο αποτελεσματικά και με μεγαλύτερη ασφάλεια. Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα διδαχθούν τον τρόπο δήλωσης και χρήσης των εκφράσεων λάμδα, που, όπως έχει ήδη αναφερθεί σε προγενέστερες γλώσσες αντικειμενοστρεφούς προγραμματισμού, είναι ανώνυμες συναρτήσεις, οι οποίες μπορούν να χρησιμοποιηθούν για τη δημιουργία απλών συναρτήσεων και μηχανισμών ελέγχου. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη χρήση των generics, τα οποία επιτρέπουν τη δημιουργία κλάσεων, διεπαφών και μεθόδων, μέσω της χρήσης παραμέτρων, παρέχοντας έτσι μεγαλύτερη ευελιξία κατά την υλοποίηση ενός έργου. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν τους τρόπους χρήσης των exceptions ως σημαντικά εργαλεία για τη διαχείριση των σφαλμάτων κατά την εκτέλεση ενός προγράμματος, ενώ, τέλος, θα διδαχθούν και τα attributes, τα οποία θα παρέχουν μεταδεδομένα για τον κώδικα, χρήσιμα για την επεξεργασία του κατά τον χρόνο εκτέλεσής του. Στο τέλος αυτής της μαθησιακής υποεπενότητας, οι εκπαιδευόμενοι/-ες θα έχουν αποκτήσει ισχυρές βάσεις γύρω από τις βασικές έννοιες της γλώσσας προγραμματισμού C#10 και θα είναι έτοιμοι/-ες να εφαρμόσουν αυτές τις έννοιες στα projects τους.

2.4.Γ.4 | ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ GUI ΜΕ ΤΑ WINDOWS FORMS

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τον τρόπο με τον οποίο θα δημιουργούν εφαρμογές γραφικής διεπαφής χρήστη (Graphical User Interface-GUI), χρησιμοποιώντας φόρμες του λειτουργικού συστήματος Microsoft Windows (MS Windows Forms). Αυτή η υποενότητα παρουσιάζει τα βασικά στοιχεία ελέγχου (Controls), όπως τα κουμπιά (Buttons), τα κείμενα (TextBoxes) και τα μενού (Menus), και τους τρόπους μέσω των οποίων οι εκπαιδευόμενοι/-ες θα διαχειρίζονται τα γεγονότα (events) κατά την αλληλεπίδραση με μια εφαρμογή. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν με ποιον τρόπο θα δημιουργούν και θα ρυθμίζουν τα στοιχεία ελέγχου, με ποιον τρόπο θα διαχειρίζονται τα γεγονότα που προκαλούνται από τις ενέργειες του/της χρήστη/-τριας (π.χ. κλικ του ποντικιού ή τα πλήκτρα του πληκτρολογίου), και, τέλος, με ποιον τρόπο θα δημιουργούν εφαρμογές, οι οποίες θα ανταποκρίνονται στις ανάγκες των χρηστών/-τριών. Επιπλέον, οι εκπαιδευόμενοι/-ες θα εξετάσουν πιο προηγμένες τεχνικές, όπως είναι η δημιουργία προσαρμοσμένων στοιχείων ελέγχου (custom controls) και η χρήση γραφικών GDI+ (Graphics Device Interface Plus) για την απεικόνιση πιο πολύπλοκων εφαρμογών γραφικής διεπαφής χρήστη (GUI). Στο τέλος αυτής της μαθησιακής υποενότητας, οι εκπαιδευόμενοι/-ες θα έχουν αποκτήσει ισχυρές βάσεις γύρω από τις βασικές έννοιες του προγραμματισμού γραφικής διεπαφής χρήστη (GUI) μέσα από τα Windows Forms, και θα είναι έτοιμοι/-ες να εφαρμόσουν αυτές τις έννοιες στα projects τους.

2.4.Γ.5 | ΠΡΟΗΓΜΕΝΕΣ ΕΝΝΟΙΕΣ GUI: ΠΡΟΣΑΡΜΟΣΜΕΝΑ ΣΤΟΙΧΕΙΑ ΕΛΕΓΧΟΥ ΚΑΙ ΓΡΑΦΙΚΑ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα επεκτείνουν τις γνώσεις τους στον προγραμματισμό γραφικής διεπαφής χρήστη (Graphical User Interface-GUI), παρουσιάζοντας προηγμένες τεχνικές εφαρμογής προσαρμοσμένων στοιχείων ελέγχου (custom controls) και προηγμένες τεχνικές χρήσης γραφικών GDI+ (Graphics Device Interface Plus) για την απεικόνιση πιο πολύπλοκων γραφικών. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους μέσω των οποίων θα δημιουργούν προσαρμοσμένα στοιχεία ελέγχου, τα οποία μπορούν να τροποποιήσουν στις ανάγκες των εφαρμογών που υλοποιούν, παρέχοντας μια πιο ευμετάβλητη (custom) εμπειρία προς τους/τις τελικούς/-ές χρήστες/-τριες. Επιπλέον, οι εκπαιδευόμενοι/-ες θα εξετάσουν τεχνικές μέσω των οποίων θα είναι σε θέση να δημιουργήσουν πιο πολύπλοκα γραφικά με τη χρήση της τεχνολογίας GDI+, τα οποία επιτρέπουν την απεικόνιση σχημάτων, όπως είναι οι γραμμές, οι κύκλοι, οι εικόνες και το κείμενο σε ένα παράθυρο. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους μέσω των οποίων θα διαχειρίζονται γεγονότα (events), τα οποία θα προκαλούνται από προσαρμοσμένα στοιχεία ελέγχου, όπως τα γεγονότα κλικ (event-clicks) ή τα γεγονότα πληκτρολόγησης (event-keys), για να δημιουργήσουν μια γραφική διεπαφή, η οποία να ανταποκρίνεται στις ενέργειες του/της τελικού/-ής χρήστη/-τριας. Στο τέλος αυτής της μαθησιακής υποενότητας,

οι εκπαιδευόμενοι/-ες θα έχουν αποκτήσει ισχυρές βάσεις γύρω από τις προηγμένες τεχνικές προγραμματισμού γραφικής διεπαφής χρήστη (GUI) και θα είναι έτοιμοι/-ες να εφαρμόσουν αυτές τις τεχνικές στα projects τους.

2.4.Γ.6 | ΕΙΣΑΓΩΓΗ ΣΤΗΝ ASP.NET ΚΑΙ ΣΤΗΝ ΑΡΧΙΤΕΚΤΟΝΙΚΗ MVC

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα γνωρίσουν για πρώτη φορά το περιβάλλον του ASP.NET και της αρχιτεκτονικής MVC (Model-View-Controller architecture). Πιο συγκεκριμένα, η παρούσα μαθησιακή υποενότητα παρουσιάζει στους/στις εκπαιδευόμενους/-ες το ASP.NET (Active Server Pages.NET), το οποίο αποτελεί μια δυναμική πλατφόρμα ανάπτυξης web εφαρμογών, η οποία επιτρέπει και την επεξεργασία τους. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους μέσω των οποίων θα δημιουργούν εφαρμογές web με τη χρήση του ASP.NET MVC, μιας προσέγγισης που χρησιμοποιεί την αρχιτεκτονική MVC για τη διαχείριση των δεδομένων (data), της εμφάνισης (view) και της λογικής ελέγχου (controller). Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα διδαχθούν την αρχιτεκτονική MVC, η οποία είναι μια μέθοδος οργάνωσης του κώδικα και διαχωρίζει την εφαρμογή σε τρία κύρια στοιχεία: το μοντέλο (model), την προβολή (view) και τον ελεγκτή (controller). Το Μοντέλο αναπαριστά τα δεδομένα και τη λογική της εφαρμογής, η Προβολή είναι υπεύθυνη για την παρουσίαση των δεδομένων στον/στην τελικό/-ή χρήστη/-τρια, και, τέλος, ο Ελεγκτής χειρίζεται την επικοινωνία μεταξύ του Μοντέλου και της Προβολής. Στο τέλος αυτής της μαθησιακής υποενότητας, οι εκπαιδευόμενοι/-ες θα έχουν αποκτήσει ισχυρές βάσεις γύρω από τις κύριες έννοιες της τεχνολογίας ASP.NET και της αρχιτεκτονικής MVC, και θα είναι έτοιμοι/-ες να εφαρμόσουν αυτές τις έννοιες στα projects τους.

2.4.Γ.7 | ΔΗΜΙΟΥΡΓΙΑ ΕΦΑΡΜΟΓΗΣ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ASP.NET MVC-ΕΡΓΑΣΙΑ ΜΕ ΔΕΔΟΜΕΝΑ: ENTITY FRAMEWORK ΚΑΙ LINQ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα δημιουργήσουν ένα σύνολο εφαρμογών ASP.NET MVC. Πιο συγκεκριμένα, αυτή η μαθησιακή υποενότητα παρουσιάζει τη διαδικασία δημιουργίας μιας εφαρμογής ASP.NET MVC από την αρχή, περιλαμβάνοντας τη δημιουργία των models (models), των views (views) και των controllers (controllers) που απαιτούνται. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους με τους οποίους θα δημιουργούν τα models, τα οποία αναπαριστούν τα δεδομένα και τη λογική της εφαρμογής, τα views, που αναπαριστούν την παρουσίαση των δεδομένων στον/στη χρήστη/-τρια, και τα controllers, που χειρίζονται την επικοινωνία μεταξύ των models και των views. Επιπλέον, θα μάθουν πώς να διαμορφώνουν την εφαρμογή τους με τη χρήση ρυθμίσεων του ASP.NET MVC, οι οποίες περιλαμβάνουν τη ρύθμιση των διαδρομών, τη ρύθμιση ασφάλειας, τη ρύθμιση των συνεδριών και άλλων χαρακτηριστικών της εφαρμογής. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν το Entity Framework και το LINQ (Language-Integrated Query), τα οποία είναι δύο ισχυρά εργαλεία για τη διαχείριση δεδομένων στο .NET Framework. Πιο

συγκεκριμένα, θα διδαχθούν τρόπους με τους οποίους θα δημιουργούν, θα ανακτούν, θα ενημερώνουν και θα διαγράφουν δεδομένα από μια βάση δεδομένων, χρησιμοποιώντας το Entity Framework, καθώς επίσης και πώς να εκτελούν ερωτήματα στα δεδομένα αυτά με τη χρήση του LINQ. Στο τέλος αυτής της μαθησιακής υποενότητας, οι εκπαιδευόμενοι/-ες θα έχουν αποκτήσει ισχυρές βάσεις γύρω από τις βασικές έννοιες της δημιουργίας μιας εφαρμογής ASP.NET MVC και θα είναι έτοιμοι/-ες να εφαρμόσουν αυτές τις έννοιες στα projects τους.

2.4.Γ.8 | ASP.NET MVC ROUTING, CONTROLLERS, VIEWS, LAYOUT-ΑΣΦΑΛΕΙΑ

ΑΝΑΠΤΥΞΗΣ ΕΦΑΡΜΟΓΩΝ ASP.NET MVC

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους μέσω των οποίων θα διαχειρίζονται τη ροή των εφαρμογών τους, με τη χρήση του routing και των controllers στο ASP.NET MVC. Πιο αναλυτικά, θα μάθουν πώς να δημιουργούν routes, πώς να διαχειρίζονται τα requests και τα responses και πώς να δημιουργούν και να χρησιμοποιούν controllers. Αυτή η υποενότητα εισάγει τους/τις εκπαιδευόμενους/-ες στη δημιουργία και στη χρήση των views και των layouts στο ASP.NET MVC. Οι εκπαιδευόμενοι/-ες θα διδαχθούν πώς να δημιουργούν views για την παρουσίαση των δεδομένων τους, πώς να χρησιμοποιούν τα Razor templates για τη δημιουργία δυναμικών ιστοτόπων HTML και πώς να δημιουργούν και να χρησιμοποιούν layouts για την επαναχρησιμοποίηση του κώδικα των views. Η υποενότητα αυτή, ακόμη, εισάγει τους/τις εκπαιδευόμενους/-ες στην εφαρμογή ασφάλειας στις εφαρμογές τους, με τη χρήση του ASP.NET MVC. Οι εκπαιδευόμενοι/-ες θα διδαχθούν πώς να εφαρμόζουν τις υπηρεσίες αυθεντικοποίησης και εξουσιοδότησης, πώς να διαχειρίζονται τις ευπάθειες ασφάλειας και πώς να προστατεύουν τα δεδομένα των χρηστών/-τριών τους. Τέλος, οι εκπαιδευόμενοι/-ες θα μάθουν πώς να δοκιμάζουν και να αναπτύσσουν τις εφαρμογές τους με τη χρήση του ASP.NET MVC, τον τρόπο με τον οποίο θα γράφουν και θα εκτελούν unit tests, πώς θα διαχειρίζονται σφάλματα και πώς θα αναπτύσσουν και θα αναβαθμίζουν τις εφαρμογές τους σε ένα παραγωγικό περιβάλλον.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Price, J. M. (2021). *C# 10 and .NET 6-Modern Cross-Platform Development: Build apps, websites, and services with ASP.NET Core 6, Blazor, and EF Core 6 using Visual Studio 2022 and Visual Studio Code*. Birmingham: Packt Publishing.
2. Sharp, J. (2008). *Microsoft Visual C# 2008*. Αθήνα: Εκδόσεις Κλειδάριθμος.

Συμπληρωματικές

1. Freeman, A. (2017). *Pro ASP.NET Core MVC 2*. New York: Apress.
2. Troelsen, A. & Japikse, P. (2022). *Pro C# 10 with .NET 6: Foundational Principles and Practices in Programming*. New York: Apress.

2.4.Δ • ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ V (VB.NET 16, MVC)- ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΕΦΑΡΜΟΓΩΝ ΓΡΑΦΙΚΗΣ ΔΙΕΠΑΦΗΣ ΧΡΗΣΤΗ (GUI) III

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Γλώσσα προγραμματισμού V (VB.NET 16, MVC)-Προγραμματισμός εφαρμογών γραφικής διεπαφής χρήστη (GUI) III» εισάγει τους/τις εκπαιδευόμενους/-ες στη γλώσσα προγραμματισμού VB.NET 16. Η συγκεκριμένη μαθησιακή ενότητα είναι επηρεασμένη από πολλές μαθησιακές ενότητες όλων των εξαμήνων κατάρτισης, αλλά κατά βάση αντλεί στοιχεία από τη μαθησιακή ενότητα «Γλώσσα Προγραμματισμού II (C++14)-Αντικειμενοστρεφής προγραμματισμός» του εξαμήνου Β΄, από τη μαθησιακή ενότητα «Γλώσσα προγραμματισμού III (JAVA9/JAVAFX)-Προγραμματισμός εφαρμογών γραφικής διεπαφής χρήστη (GUI) I» του εξαμήνου Γ΄ και από τη μαθησιακή ενότητα «Γλώσσα προγραμματισμού IV (ASP.NET, MVC, C#10)-Προγραμματισμός εφαρμογών γραφικής διεπαφής χρήστη (GUI) II» του εξαμήνου Δ΄. Στην παρούσα μαθησιακή ενότητα, ο/η εκπαιδευόμενος/-η θα έρθει σε επαφή με προχωρημένα θέματα αντικειμενοστρεφούς προγραμματισμού, όπως αυτά των δομών επιλογής, επανάληψης, συναρτήσεων, εξαιρέσεων και υπερφόρτωσης, της δημιουργίας κλάσεων και αντικειμένων, της χρήσης μεθόδων κληρονομικότητας και πολυμορφισμού μέσω των ασφαλών τεχνικών ενθυλάκωσης των πεδίων κάθε αντικειμένου, δημιουργώντας έτσι πιο ανθεκτικές και ασφαλείς εφαρμογές γραφικού περιβάλλοντος μέσω GUI APIs, βάσει του μοντέλου Model View Control (MVC). Με βάση τα παραπάνω, οι εφαρμογές αυτές θα επεξεργάζονται πολυμεσικά αρχεία ή/και θα συνδέονται με βάση δεδομένων, δημιουργώντας ολοκληρωμένες εφαρμογές.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Αναπτύσσουν εφαρμογές κονσόλας ή/και εφαρμογές γραφικού περιβάλλοντος, με τη χρήση γλώσσας προγραμματισμού VB.NET 16,
- Συνεισφέρουν με πρωτότυπο κώδικα στην κοινότητα των προγραμματιστών εφαρμογών,
- Τροποποιούν υπάρχοντα κώδικα, ανάλογα με τις προδιαγραφές των GUI εφαρμογών που αναπτύσσουν,
- Αποφασίζουν για τη χρήση ή μη κατάλληλων βιβλιοθηκών, σχετικών με την ανάπτυξη εφαρμογών,
- Συνεργάζονται με άλλους/-ες εκπαιδευόμενους/-ες, συγκροτώντας ομάδες, με σκοπό τη δημιουργία και την επίλυση προβλημάτων των εφαρμογών GUI,
- Υιοθετούν θετική στάση ως προς τις υπάρχουσες τεχνολογίες ανάπτυξης εφαρμογών GUI,
- Αναγνωρίζουν τις βασικές προγραμματιστικές αρχές του αντικειμενοστρεφούς προγραμματισμού, βάσει των οποίων έχει υλοποιηθεί μια εφαρμογή VB.NET 16,
- Επιλέγουν τα κατάλληλα μέλη κλάσεων (πεδία, μεθόδους κ.ά.) που θα χρησιμοποιήσουν στις εφαρμογές που θα αναπτύσσουν,
- Αναγνωρίζουν τη σημασία του πολυμορφισμού και της κληρονομικότητας.

Βασικές έννοιες και λέξεις-κλειδιά

Visual Basic 16 .NET / .NET 4.8 / ASP.NET / MVC Architecture / Object-Oriented Programming / Windows Forms / Custom Controls / Entity Framework / LINQ / Security / Deployment.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (5), Σύνολο (5).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΕΙΣΑΓΩΓΗ ΣΤΗ VB16 ΚΑΙ ΣΤΟ .NET 4.8.1 FRAMEWORK
2	ΚΑΤΑΝΟΗΣΗ ΤΟΥ ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΟΥΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΣΤΗ VB16
3	ΕΞΕΡΕΥΝΗΣΗ ΤΗΣ ΣΥΝΤΑΞΗΣ ΚΑΙ ΤΩΝ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ ΤΗΣ VB16
4	ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ GUI ΜΕ ΤΑ WINDOWS FORMS
5	ΠΡΟΗΓΜΕΝΕΣ ΕΝΝΟΙΕΣ GUI: ΠΡΟΣΑΡΜΟΣΜΕΝΑ ΣΤΟΙΧΕΙΑ ΕΛΕΓΧΟΥ ΚΑΙ ΓΡΑΦΙΚΑ
6	ΕΙΣΑΓΩΓΗ ΣΤΗΝ ASP.NET ΚΑΙ ΣΤΗΝ ΑΡΧΙΤΕΚΤΟΝΙΚΗ MVC
7	ΔΗΜΙΟΥΡΓΙΑ ΕΦΑΡΜΟΓΗΣ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ASP.NET MVC-ΕΡΓΑΣΙΑ ΜΕ ΔΕΔΟΜΕΝΑ: ENTITY FRAMEWORK ΚΑΙ LINQ
8	ASP.NET MVC ROUTING, CONTROLLERS, VIEWS, LAYOUT-ΑΣΦΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ΕΦΑΡΜΟΓΩΝ ASP.NET MVC

Κείμενο αναφοράς

2.4.Δ.1 | ΕΙΣΑΓΩΓΗ ΣΤΗ VB16 ΚΑΙ ΣΤΟ .NET 4.8.1 FRAMEWORK

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τα εισαγωγικά στοιχεία της αντικειμενοστρεφούς γλώσσας προγραμματισμού Visual Basic 16 στο περιβάλλον .NET 4.8.1. Οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με την εγκατάσταση και τη ρύθμιση του περιβάλλοντος ανάπτυξης (IDE-Integrated Development Environment), το οποίο προτείνεται να είναι το Microsoft Visual Studio 2019 ή νεότερο (ή κάποιο παρεμφερές αντίστοιχο IDE), και με τις βασικές έννοιες της γλώσσας, όπως είναι οι τύποι δεδομένων, οι μεταβλητές και οι λειτουργίες τους. Αυτή η υποενότητα θα παρουσιάσει στους/στις εκπαιδευόμενους/-ες την αρχιτεκτονική της γλώσσας Visual Basic 16 και της πλατφόρμας .NET 4.8.1, παρέχοντας μια επισκόπηση των κύριων χαρακτηριστικών και των δυνατοτήτων τους. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τους τρόπους με τους οποίους θα εγκαταστήσουν και θα ρυθμίσουν το IDE, θα μάθουν πώς θα δημιουργήσουν και θα εκτελέσουν απλά προγράμματα σε Visual Basic 16, καθώς επίσης και πώς θα χρησιμοποιήσουν τις βασικές δομές δεδομένων και τις λειτουργίες της γλώσσας προγραμματισμού Visual Basic 16. Τέλος, οι εκπαιδευόμενοι/-ες θα εξοικειωθούν με την αλληλεπίδρασή τους με το IDE στο οποίο θα εργάζονται, με τη διαχείριση των σφαλμάτων και με την αποσφαλμάτωση των προγραμμάτων τους. Στο τέλος αυτής της μαθησιακής υποενότητας, οι εκπαιδευόμενοι/-ες θα έχουν μια καλή γνώση των βασικών εννοιών της γλώσσας προγραμματισμού Visual Basic 16 και της .NET 4.8.1 και θα είναι έτοιμοι/-ες να προχωρήσουν σε πιο προηγμένα θέματα προγραμματισμού.

2.4.Δ.2 | ΚΑΤΑΝΟΗΣΗ ΤΟΥ ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΟΥΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΣΤΗ VB16

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα εξερευνήσουν τις βασικές έννοιες του αντικειμενοστρεφούς προγραμματισμού (OOP) στη γλώσσα προγραμματισμού Visual Basic 16. Αυτή η υποενότητα θα παρουσιάσει την ιδέα της κλάσης (class) και των γνωρισμάτων της, που έχουν ήδη διδαχθεί σε προγενέστερες αντικειμενοστρεφείς γλώσσες προγραμματισμού (γλώσσα προγραμματισμού C++14, γλώσσα προγραμματισμού JAVA9 και γλώσσα προγραμματισμού Python3). Οι εκπαιδευόμενοι/-ες θα διδαχθούν τον συντακτικό τρόπο με τον οποίο θα ορίζουν κλάσεις, πώς θα ορίζουν τα μέλη κάθε κλάσης και πώς θα δημιουργούν αντικείμενα από αυτές τις κλάσεις στη γλώσσα προγραμματισμού Visual Basic 16. Επιπλέον, θα εξεταστεί η έννοια της κληρονομικότητας (Inherits), η οποία επιτρέπει τη δημιουργία νέων κλάσεων που επεκτείνουν ή τροποποιούν τη συμπεριφορά των υπάρχουσών κλάσεων. Ο, ήδη γνωστός ως έννοια, πολυμορφισμός (polymorphism), μια άλλη κεντρική έννοια του OOP, θα παρουσιαστεί ως τρόπος για την αλλαγή της συμπεριφοράς των αντικειμένων με βάση την κλάση τους. Τέλος, οι εκπαιδευόμενοι/-ες θα διδαχθούν τον τρόπο πραγματοποίησης της απόκρυψης δεδομένων (encapsulation) ενός αντικειμένου που έχει δηλωθεί στη γλώσσα προγραμματισμού Visual Basic 16, εξασφαλίζοντας την ακεραιότητα και την ασφάλεια των ευαίσθητων πεδίων του. Στο τέλος αυτής της μαθησιακής υποενότητας, οι εκπαιδευόμενοι/-ες θα έχουν αποκτήσει ισχυρές βάσεις γύρω από τον OOP στη Visual Basic 16 και θα είναι έτοιμοι/-ες να εφαρμόσουν αυτές τις έννοιες στα projects τους.

2.4.Δ.3 | ΕΞΕΡΕΥΝΗΣΗ ΤΗΣ ΣΥΝΤΑΞΗΣ ΚΑΙ ΤΩΝ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ ΤΗΣ VB16

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν επιπλέον στοιχεία, σχετικά με τη σύνταξη της Visual Basic 16 και τα γενικά χαρακτηριστικά της. Αυτή η υποενότητα παρουσιάζει τις εκφράσεις λάμδα (anonymous functions ή lambda expressions), τα generic types, τα exceptions (Try, Catch, Throw, Finally) και τα attributes. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τους τρόπους με τους οποίους θα χρησιμοποιούν αυτά τα χαρακτηριστικά, για να γράφουν κώδικα πιο αποτελεσματικά, με μεγαλύτερη ασφάλεια. Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα διδαχθούν τον τρόπο δήλωσης και χρήσης των εκφράσεων λάμδα, που, όπως έχει ήδη αναφερθεί σε προγενέστερες γλώσσες αντικειμενοστρεφούς προγραμματισμού, είναι ανώνυμες συναρτήσεις, ικανές να χρησιμοποιηθούν για τη δημιουργία απλών συναρτήσεων και μηχανισμών ελέγχου. Επιπλέον, οι εκπαιδευόμενοι/-ες θα διδαχθούν τη χρήση των generics, τα οποία επιτρέπουν τη δημιουργία κλάσεων, μεταβλητών και συναρτήσεων μέσω της χρήσης παραμέτρων, παρέχοντας έτσι μεγαλύτερη ευελιξία κατά την υλοποίηση ενός έργου. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν τους τρόπους χρήσης των exceptions ως σημαντικά εργαλεία για τη διαχείριση των σφαλμάτων κατά την εκτέλεση ενός προγράμματος, ενώ, τέλος, θα διδαχθούν και τα attributes, τα οποία θα παρέχουν μεταδεδομένα για τον κώδικα και μπορούν να χρησιμοποιηθούν

για την επεξεργασία του κατά τον χρόνο εκτέλεσης του. Στο τέλος αυτής της μαθησιακής υποενότητας, οι εκπαιδευόμενοι/-ες θα έχουν αποκτήσει ισχυρές βάσεις γύρω από τις βασικές έννοιες της γλώσσας προγραμματισμού Visual Basic 16 και θα είναι έτοιμοι/-ες να εφαρμόσουν αυτές τις έννοιες στα projects τους.

2.4.Δ.4 | ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ GUI ΜΕ ΤΑ WINDOWS FORMS

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τον τρόπο με τον οποίο θα δημιουργούν εφαρμογές γραφικής διεπαφής χρήστη (Graphical User Interface-GUI), χρησιμοποιώντας φόρμες του λειτουργικού συστήματος Microsoft Windows (MS Windows Forms). Αυτή η υποενότητα παρουσιάζει τα βασικά στοιχεία ελέγχου (Controls), όπως τα κουμπιά (Buttons), τα κείμενα (TextBoxes) και τα μενού (Menus), και τους τρόπους με τους οποίους οι εκπαιδευόμενοι/-ες θα διαχειρίζονται τα γεγονότα (events) κατά την αλληλεπίδρασή τους με μια εφαρμογή. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν με ποιον τρόπο θα δημιουργούν και θα ρυθμίζουν τα στοιχεία ελέγχου, με ποιον τρόπο θα διαχειρίζονται τα γεγονότα που προκαλούνται από τις ενέργειες του/της χρήστη/-τριας (π.χ. τα κλικ του ποντικιού ή τα πλήκτρα του πληκτρολογίου), και, τέλος, με ποιον τρόπο θα δημιουργούν εφαρμογές, οι οποίες να ανταποκρίνονται στις ανάγκες των χρηστών/-τριών. Επιπλέον, οι εκπαιδευόμενοι/-ες θα εξετάσουν πιο προηγμένες τεχνικές, όπως είναι η δημιουργία προσαρμοσμένων στοιχείων ελέγχου (custom controls) και η χρήση γραφικών GDI+ (Graphics Device Interface Plus) για την απεικόνιση πιο πολύπλοκων εφαρμογών γραφικής διεπαφής χρήστη (GUI). Στο τέλος αυτής της μαθησιακής υποενότητας, οι εκπαιδευόμενοι/-ες θα έχουν αποκτήσει ισχυρές βάσεις γύρω από τις βασικές έννοιες του προγραμματισμού γραφικής διεπαφής χρήστη (GUI) μέσα από τα Windows Forms και θα είναι έτοιμοι/-ες να εφαρμόσουν αυτές τις έννοιες στα projects τους.

2.4.Δ.5 | ΠΡΟΗΓΜΕΝΕΣ ΕΝΝΟΙΕΣ GUI: ΠΡΟΣΑΡΜΟΣΜΕΝΑ ΣΤΟΙΧΕΙΑ ΕΛΕΓΧΟΥ ΚΑΙ ΓΡΑΦΙΚΑ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα επεκτείνουν τις γνώσεις τους στον προγραμματισμό γραφικής διεπαφής χρήστη (Graphical User Interface-GUI), παρουσιάζοντας προηγμένες τεχνικές εφαρμογής προσαρμοσμένων στοιχείων ελέγχου (custom controls) και προηγμένες τεχνικές χρήσης γραφικών GDI+ (Graphics Device Interface Plus) για την απεικόνιση πιο πολύπλοκων γραφικών. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους με τους οποίους θα δημιουργούν προσαρμοσμένα στοιχεία ελέγχου, τα οποία μπορούν να τροποποιήσουν τις ανάγκες των εφαρμογών που υλοποιούν, παρέχοντας μια πιο ευμετάβλητη (custom) εμπειρία προς τους/τις τελικούς/-ές χρήστες/-τριες. Επιπλέον, οι εκπαιδευόμενοι/-ες θα εξετάσουν τεχνικές με τις οποίες θα είναι σε θέση να δημιουργήσουν πιο πολύπλοκα γραφικά με τη χρήση της τεχνολογίας GDI+, τα οποία επιτρέπουν την απεικόνιση σχημάτων, όπως είναι οι γραμμές, οι κύκλοι, οι εικόνες και το κείμενο σε ένα παράθυρο. Επιπλέον, οι

εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους με τους οποίους θα διαχειρίζονται γεγονότα (events), τα οποία θα προκαλούνται από προσαρμοσμένα στοιχεία ελέγχου, όπως τα γεγονότα κλικ (event-clicks) ή τα γεγονότα πληκτρολόγησης (event-keys), για να δημιουργήσουν μια γραφική διεπαφή που να ανταποκρίνεται στις ενέργειες του/της τελικού/-ής χρήστη/-τριας. Στο τέλος αυτής της μαθησιακής υποενότητας, οι εκπαιδευόμενοι/-ες θα έχουν αποκτήσει ισχυρές βάσεις γύρω από τις προηγμένες τεχνικές προγραμματισμού γραφικής διεπαφής χρήστη (GUI) και θα είναι έτοιμοι/-ες να εφαρμόσουν αυτές τις τεχνικές στα projects τους.

2.4.Δ.6 | ΕΙΣΑΓΩΓΗ ΣΤΗΝ ASP.NET ΚΑΙ ΣΤΗΝ ΑΡΧΙΤΕΚΤΟΝΙΚΗ MVC

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα γνωρίσουν για πρώτη φορά το περιβάλλον του ASP.NET και της αρχιτεκτονικής MVC (Model-View-Controller architecture). Πιο συγκεκριμένα, η παρούσα μαθησιακή υποενότητα παρουσιάζει στους/στις εκπαιδευόμενους/-ες το ASP.NET (Active Server Pages.NET), το οποίο αποτελεί μια δυναμική πλατφόρμα ανάπτυξης web εφαρμογών και επιτρέπει την επεξεργασία τους. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους με τους οποίους θα δημιουργούν εφαρμογές web με τη χρήση του ASP.NET MVC, μιας προσέγγισης που χρησιμοποιεί την αρχιτεκτονική MVC για τη διαχείριση των δεδομένων (data), της εμφάνισης (view) και της λογικής ελέγχου (controller). Πιο αναλυτικά, οι εκπαιδευόμενοι/-ες θα διδαχθούν την αρχιτεκτονική MVC, η οποία είναι μια μέθοδος οργάνωσης του κώδικα και διαχωρίζει την εφαρμογή σε τρία κύρια στοιχεία: το μοντέλο (model), την προβολή (view) και τον ελεγκτή (controller). Το μοντέλο αναπαριστά τα δεδομένα και τη λογική της εφαρμογής, η προβολή είναι υπεύθυνη για την παρουσίαση των δεδομένων στον/στην τελικό/-ή χρήστη/-τρια, και, τέλος, ο ελεγκτής χειρίζεται την επικοινωνία μεταξύ του μοντέλου και της προβολής. Στο τέλος αυτής της μαθησιακής υποενότητας, οι εκπαιδευόμενοι/-ες θα έχουν αποκτήσει ισχυρές βάσεις γύρω από τις κύριες έννοιες της τεχνολογίας ASP.NET και της αρχιτεκτονικής MVC, και θα είναι έτοιμοι/-ες να εφαρμόσουν αυτές τις έννοιες στα projects τους.

2.4.Δ.7 | ΔΗΜΙΟΥΡΓΙΑ ΕΦΑΡΜΟΓΗΣ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ASP.NET MVC-ΕΡΓΑΣΙΑ ΜΕ ΔΕΔΟΜΕΝΑ: ENTITY FRAMEWORK ΚΑΙ LINQ

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα δημιουργήσουν ένα σύνολο εφαρμογών ASP.NET MVC. Πιο συγκεκριμένα, αυτή η μαθησιακή υποενότητα παρουσιάζει τη διαδικασία δημιουργίας μιας εφαρμογής ASP.NET MVC από την αρχή, περιλαμβάνοντας τη δημιουργία των models (models), των views (views) και των controllers (controllers) που απαιτούνται. Οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους με τους οποίους θα δημιουργούν τα models, που αναπαριστούν τα δεδομένα και τη λογική της εφαρμογής, τα views, που αναπαριστούν την παρουσίαση των δεδομένων στον/στη χρήστη/-τρια, και τα controllers, που χειρίζονται την επικοινωνία μεταξύ των models και των views. Επιπλέον, θα μάθουν πώς να διαμορφώνουν την

εφαρμογή τους με τη χρήση ρυθμίσεων του ASP.NET MVC, οι οποίες περιλαμβάνουν τη ρύθμιση των διαδρομών, τη ρύθμιση ασφάλειας, τη ρύθμιση των συνεδριών και άλλων χαρακτηριστικών της εφαρμογής. Στη συνέχεια, οι εκπαιδευόμενοι/-ες θα διδαχθούν το Entity Framework και το LINQ (Language-Integrated Query), τα οποία είναι δύο ισχυρά εργαλεία για τη διαχείριση δεδομένων στο .NET Framework. Πιο συγκεκριμένα, θα διδαχθούν τρόπους με τους οποίους θα δημιουργούν, θα ανακτούν, θα ενημερώνουν και θα διαγράφουν δεδομένα από μια βάση δεδομένων, χρησιμοποιώντας το Entity Framework, καθώς επίσης και πώς θα εκτελούν ερωτήματα στα δεδομένα αυτά με τη χρήση του LINQ. Στο τέλος αυτής της μαθησιακής υποενότητας, οι εκπαιδευόμενοι/-ες θα έχουν αποκτήσει ισχυρές βάσεις γύρω από τις βασικές έννοιες της δημιουργίας μιας εφαρμογής ASP.NET MVC και θα είναι έτοιμοι/-ες να εφαρμόσουν αυτές τις έννοιες στα projects τους.

2.4.Δ.8 | ASP.NET MVC ROUTING, CONTROLLERS, VIEWS, LAYOUT-ΑΣΦΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ΕΦΑΡΜΟΓΩΝ ASP.NET MVC

Σε αυτή τη μαθησιακή υποενότητα, οι εκπαιδευόμενοι/-ες θα διδαχθούν τρόπους με τους οποίους θα διαχειρίζονται τη ροή των εφαρμογών τους με τη χρήση του routing και των controllers στο ASP.NET MVC. Πιο αναλυτικά, θα μάθουν πώς να δημιουργούν routes, πώς να διαχειρίζονται τα requests και τα responses και πώς να δημιουργούν και να χρησιμοποιούν controllers. Αυτή η υποενότητα εισάγει τους/τις εκπαιδευόμενους/-ες στη δημιουργία και τη χρήση των views και των layouts στο ASP.NET MVC. Οι εκπαιδευόμενοι/-ες θα διδαχθούν πώς να δημιουργούν views για την παρουσίαση των δεδομένων τους, πώς να χρησιμοποιούν τα Razor templates για τη δημιουργία δυναμικών ιστοτόπων HTML και πώς να δημιουργούν και να χρησιμοποιούν layouts για την επαναχρησιμοποίηση του κώδικα των views. Αυτή η υποενότητα εισάγει τους/τις εκπαιδευόμενους/-ες στην εφαρμογή ασφάλειας στις εφαρμογές τους με τη χρήση του ASP.NET MVC. Οι εκπαιδευόμενοι/-ες θα διδαχθούν πώς να εφαρμόζουν τις υπηρεσίες αυθεντικοποίησης και εξουσιοδότησης, πώς να διαχειρίζονται τις ευπάθειες ασφάλειας και πώς να προστατεύουν τα δεδομένα των χρηστών/-τριών τους. Τέλος, στη μαθησιακή υποενότητα αυτή, οι εκπαιδευόμενοι/-ες θα μάθουν πώς να δοκιμάζουν και να αναπτύσσουν τις εφαρμογές τους με τη χρήση του ASP.NET MVC και τον τρόπο με τον οποίο θα γράφουν και θα εκτελούν unit tests, πώς να διαχειρίζονται σφάλματα, αλλά και πώς να αναπτύσσουν και να αναβαθμίζουν τις εφαρμογές τους σε ένα παραγωγικό περιβάλλον.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Halvorson, M. (2016). *Microsoft Visual Basic 2013 Βήμα-Βήμα*. Αθήνα: Εκδόσεις Κλειδάριθμος.
2. Yamacli, S. (2019). *Beginner's Guide to Visual Basic .NET Programming: A Practical Approach to VB.NET*. Independently published.

Συμπληρωματικές

1. Guleria, P. (2022). *ASP.NET and VB.NET in 30 Days: Acquire a Solid Foundation in the Fundamentals of Windows and Web Application Development*. New Delhi: BPB Publications.
2. Hussain, K. & Hussain, F. (2023). *Mastering VB.NET: A Comprehensive Guide to Visual Basic .NET Programming*. Singapore: Sonar Publishing.

2.4.Ε • ΕΦΑΡΜΟΣΜΕΝΕΣ ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΣΤΗ ΣΥΓΧΡΟΝΗ ΑΓΟΡΑ ΕΡΓΑΣΙΑΣ

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Εφαρμοσμένες γλώσσες προγραμματισμού στη σύγχρονη αγορά εργασίας» εισάγει τους/τις εκπαιδευόμενους/-ες στην εκμάθηση των πιο διαδεδομένων και τελευταίας τεχνολογίας γλωσσών προγραμματισμού που χρησιμοποιούνται στην αγορά εργασίας. Σκοπός της μαθησιακής ενότητας δεν είναι ο/η εκπαιδευόμενος/-η να εντρυφήσει σε μια εις βάθος ανάλυση κάθε γλώσσας προγραμματισμού, αλλά να έχει την τεχνογνωσία του πεδίου εφαρμογής της και της λογικής πάνω στην οποία έχει στηριχθεί, ώστε εύκολα να μπορεί να εγκλιματιστεί σε μια ομάδα έργου που χρησιμοποιεί αυτή την τεχνολογία ανάπτυξης εφαρμογών. Οι γλώσσες προγραμματισμού που θα διδαχθούν οι εκπαιδευόμενοι/-ες είναι η γλώσσα R και το πεδίο εφαρμογής της, η γλώσσα Swift και το πεδίο εφαρμογής της, η γλώσσα Kotlin και το πεδίο εφαρμογής της, η γλώσσα Go και το πεδίο εφαρμογής της, η γλώσσα Rust και το πεδίο εφαρμογής της, η γλώσσα Perl και το πεδίο εφαρμογής της, η γλώσσα Xml και το πεδίο εφαρμογής της, η γλώσσα Ruby και το πεδίο εφαρμογής της και η γλώσσα TypeScript και το πεδίο εφαρμογής της. Για όλες τις παραπάνω γλώσσες, θα πρέπει να εξηγηθούν οι λόγοι για τους οποίους προτιμάται η κάθε γλώσσα έναντι των άλλων, να προταθεί ένα IDE (σε αυτό θα γίνει η συγγραφή του κώδικα) και να δοθούν στους/στις εκπαιδευόμενους/-ες δύο απλά παραδείγματα σε κάθε γλώσσα.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Επιλέγουν την κατάλληλη γλώσσα προγραμματισμού, με την οποία θα πραγματοποιήσουν την υλοποίηση ενός λογισμικού ή μιας εφαρμογής,
- Αναγνωρίζουν την αναγκαιότητα χρήσης μιας συγκεκριμένης γλώσσας προγραμματισμού, με την οποία θα πραγματοποιήσουν την υλοποίηση ενός λογισμικού ή μιας εφαρμογής,
- Αναπτύσσουν εφαρμογές που καλύπτουν ένα ευρύτατο φάσμα των αναγκών της σύγχρονης αγοράς εργασίας,
- Αποφασίζουν για τη χρήση της τεχνολογίας υλοποίησης που θα επιλεγεί, σύμφωνα με τις προδιαγραφές της εφαρμογής προς ανάπτυξη,
- Συνεισφέρουν με πρωτότυπο κώδικα στην κοινότητα των προγραμματιστών εφαρμογών,
- Υιοθετούν θετική στάση ως προς την ανάπτυξη εφαρμογών που καλύπτουν ένα ευρύτατο φάσμα των αναγκών της σύγχρονης αγοράς εργασίας.

Βασικές έννοιες και λέξεις-κλειδιά

IDE(integrated development environment) / API(Application Programming Interface) / Interoperability / Metaprogramming / Script / Virtual machine / Chatbot.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (2), Σύνολο (2).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	Η ΓΛΩΣΣΑ R ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ
2	Η ΓΛΩΣΣΑ SWIFT ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ
3	Η ΓΛΩΣΣΑ KOTLIN ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ
4	Η ΓΛΩΣΣΑ GO ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ
5	Η ΓΛΩΣΣΑ RUST ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ
6	Η ΓΛΩΣΣΑ PERL ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ
7	Η ΓΛΩΣΣΑ XML ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ
8	Η ΓΛΩΣΣΑ RUBY ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ
9	Η ΓΛΩΣΣΑ TYPESCRIPT ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ

2.4.E.1 | Η ΓΛΩΣΣΑ R ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ

Σε αυτή την υποενότητα, παρουσιάζονται η γλώσσα R και το πεδίο εφαρμογής της, το οποίο είναι η στατιστική επεξεργασία και η ανάλυση δεδομένων, και τα πλεονεκτήματα της γλώσσας R έναντι των άλλων γλωσσών. Τονίζονται ιδιαίτερα οι δυνατότητες στατιστικής επεξεργασίας που προσφέρει η R μέσω των πακέτων και βιβλιοθηκών για στατιστική ανάλυση που είναι διαθέσιμα: η γραμμική παλινδρόμηση, η ανάλυση διακύμανσης, η ανάλυση διάσπασης. Άλλο ένα σημαντικό πλεονέκτημα της R είναι οι ισχυρές λειτουργίες επεξεργασίας δεδομένων (π.χ. εισαγωγή, καθαρισμός και μετασχηματισμός δεδομένων) που παρέχει. Διαθέτει επίσης εκτενείς δυνατότητες για τον συνδυασμό δεδομένων από διάφορες πηγές. Επιπροσθέτως, η R παρέχει εξαιρετικές δυνατότητες για τη δημιουργία όμορφων και εκφραστικών γραφικών παραστάσεων, όπως των διαγραμμάτων, των γραφημάτων και των πινάκων. Ως ολοκληρωμένο περιβάλλον ανάπτυξης για τη γλώσσα R προτείνεται το Rstudio. Γίνεται εγκατάσταση του Rstudio και παρουσιάζεται ο τρόπος με τον οποίο γίνεται η εγκατάσταση των πακέτων και των βιβλιοθηκών. Επίσης, παρουσιάζονται οι βασικές λειτουργίες του Rstudio. Τέλος, γίνεται παρουσίαση δύο ενδεικτικών ασκήσεων. Η πρώτη άσκηση προτείνεται να αφορά την εξαγωγή στατιστικών μεγεθών (π.χ. μέση τιμή, ελάχιστη, μέγιστη) από ένα δείγμα τιμών θερμοκρασιών, όπου οι θερμοκρασίες θα βρίσκονται σε αρχείο. Η δεύτερη άσκηση προτείνεται να αφορά τη δημιουργία δισδιάστατου γραφήματος, το οποίο απεικονίζει τη μέση θερμοκρασία ανά ημέρα, για διάστημα μιας εβδομάδας.

2.4.E.2 | Η ΓΛΩΣΣΑ SWIFT ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ

Εδώ, παρουσιάζονται η γλώσσα Swift και το πεδίο εφαρμογής της, το οποίο είναι η ανάπτυξη εφαρμογών για συσκευές της Apple, και τα πλεονεκτήματα της γλώσσας Swift έναντι των άλλων γλωσσών, τα οποία, μεταξύ άλλων, περιλαμβάνουν την ενσωματωμένη υποστήριξη για iOS και macOS, γεγονός που την καθιστά κύρια γλώσσα προγραμματισμού για εφαρμογές των συσκευών της Apple, καθώς και τα ενσωματωμένα ισχυρά χαρακτηριστικά ασφάλειας που βοηθούν τους/τις προγραμματιστές/-τριες να αποφύγουν σφάλματα και προβλήματα ασφάλειας. Ως ολοκληρωμένο περιβάλλον ανάπτυξης, προτείνεται το Xcode, το επίσημο IDE της Apple για την ανάπτυξη εφαρμογών. Γίνεται εγκατάσταση του Xcode και παρουσιάζονται οι βασικές λειτουργίες του. Σημειώνεται εδώ ότι το Xcode λειτουργεί σε λειτουργικό σύστημα macOS. Για την εγκατάστασή του σε Windows, απαιτείται η εγκατάσταση μιας εικονικής μηχανής (virtual machine), η οποία θα τρέχει λειτουργικό σύστημα macOS σε υπολογιστή με Windows (όπως το VMware). Τέλος, γίνεται παρουσίαση δύο ενδεικτικών ασκήσεων. Η πρώτη άσκηση προτείνεται να αφορά τη δημιουργία εφαρμογής, η οποία θα υπολογίζει τον δείκτη μάζας σώματος, βάσει ύψους και βάρους που θα δίνονται από τον/τη χρήστη/-τρια. Η δεύτερη

άσκηση προτείνεται να αφορά τη δημιουργία εφαρμογής «Υπενθύμισης Καθημερινών Εργασιών», που θα επιτρέπει στον/στη χρήστη/-τρια να καταχωρεί και να διαχειρίζεται καθημερινές εργασίες.

2.4.E.3 | Η ΓΛΩΣΣΑ KOTLIN ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ

Σε αυτή την υποενότητα, παρουσιάζονται η γλώσσα Kotlin και το πεδίο εφαρμογής της, το οποίο είναι η ανάπτυξη εφαρμογών για συσκευές Android, και τα πλεονεκτήματα της γλώσσας Kotlin έναντι των άλλων γλωσσών, τα οποία, μεταξύ άλλων, περιλαμβάνουν την πλήρη συμβατότητα με την JAVA, την κύρια γλώσσα προγραμματισμού για εφαρμογές android και τα ενσωματωμένα ισχυρά χαρακτηριστικά ασφάλειας που διαθέτει και βοηθούν τους/τις προγραμματιστές/-τριες να αποφύγουν σφάλματα και προβλήματα ασφάλειας. Ως ολοκληρωμένο περιβάλλον ανάπτυξης, προτείνεται το Android Studio. Γίνεται εγκατάσταση του Android Studio και παρουσιάζονται οι βασικές λειτουργίες του. Επίσης, δημιουργείται μια εικονική συσκευή (virtual device), στην οποία θα εξεταστούν οι εφαρμογές. Τέλος, γίνεται παρουσίαση δύο ενδεικτικών ασκήσεων. Η πρώτη άσκηση προτείνεται να αφορά τη δημιουργία εφαρμογής, η οποία θα υπολογίζει τον δείκτη μάζας σώματος, βάσει ύψους και βάρους που θα δίνονται από τον/τη χρήστη/-τρια. Η δεύτερη άσκηση προτείνεται να αφορά τη δημιουργία εφαρμογής «Υπενθύμισης Καθημερινών Εργασιών», που θα επιτρέπει στον/στη χρήστη/-τρια να καταχωρεί και να διαχειρίζεται καθημερινές εργασίες. Σημειώνεται εδώ ότι είναι ακριβώς τα ίδια παραδείγματα που χρησιμοποιήθηκαν και για τη γλώσσα Swift, με σκοπό να αναδειχτούν οι διαφορές των δύο γλωσσών, οι οποίες χρησιμοποιούνται κυρίως για την ανάπτυξη mobile εφαρμογών.

2.4.E.4 | Η ΓΛΩΣΣΑ GO ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ

Σε αυτή την υποενότητα, παρουσιάζεται η γλώσσα Go (ή Golang) και το πεδίο εφαρμογής της, το οποίο είναι η ανάπτυξη back-end διαδικτυακών εφαρμογών και υπηρεσιών, που διαχειρίζονται πολλά αιτήματα ταυτόχρονα (π.χ. web services, κατανεμημένα συστήματα, cloud υπηρεσίες), και τα πλεονεκτήματα της γλώσσας Go έναντι των άλλων γλωσσών, τα οποία, μεταξύ άλλων, περιλαμβάνουν τη γρήγορη εκτέλεση και την αποδοτική διαχείριση μνήμης, καθιστώντας την κατάλληλη για εφαρμογές που απαιτούν υψηλές αποδόσεις. Επιπρόσθετα, επειδή διαθέτει ενσωματωμένες δυνατότητες παραλληλισμού, είναι κατάλληλη για προγράμματα που απαιτούν πολλαπλές νηματικές διεργασίες. Ως ολοκληρωμένο περιβάλλον ανάπτυξης, προτείνεται το Visual Studio Code, ένα open-source εργαλείο, το οποίο με την εγκατάσταση της επέκτασης “VS Code Go” υποστηρίζει τη γλώσσα Go. Γίνεται εγκατάσταση και παρουσιάζονται οι βασικές λειτουργίες του εργαλείου. Επίσης, παρουσιάζεται και το εργαλείο GoLand, το οποίο όμως δεν παρέχεται δωρεάν. Τέλος, γίνεται παρουσίαση δύο ενδεικτικών ασκήσεων. Η πρώτη άσκηση προτείνεται να αφορά τη δημιουργία

ενός διακομιστή http, που ακούει στη θύρα 8080, αποκρίνεται σε αιτήσεις HTTP και επιστρέφει μια στατική ιστοσελίδα. Η δεύτερη άσκηση προτείνεται να αφορά τη δημιουργία εφαρμογής, η οποία θα προσομοιώνει τη διαχείριση τραπεζικών λογαριασμών (ανάληψη, κατάθεση, υπόλοιπο) και θα επιτρέπει την ταυτόχρονη πρόσβαση χρηστών στην εφαρμογή.

2.4.E.5 | Η ΓΛΩΣΣΑ RUST ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ

Εδώ, παρουσιάζονται η γλώσσα Rust και το πεδίο εφαρμογής της, το οποίο είναι η ανάπτυξη εφαρμογών (η ασφάλεια, η αξιοπιστία και η απόδοση είναι κρίσιμες, π.χ. πυρήνες λειτουργικού συστήματος, διακομιστές και διαδικτυακά πρωτόκολλα, εφαρμογές ασφάλειας) και τα πλεονεκτήματα της γλώσσας Rust έναντι των άλλων γλωσσών, τα οποία, μεταξύ άλλων, περιλαμβάνουν: το ισχυρό σύστημα ασφάλειας (αποτρέπει σφάλματα μνήμης και σφάλματα σχετικά με την ασφάλεια του προγράμματος), την αποδοτική χρήση της μνήμης, καθιστώντας την κατάλληλη για εφαρμογές που απαιτούν υψηλή απόδοση, και τα ισχυρά εργαλεία για τη διαχείριση νημάτων και τον ασύγχρονο προγραμματισμό, κάνοντάς την κατάλληλη για εφαρμογές που απαιτούν παράλληλισμό. Ως ολοκληρωμένο περιβάλλον ανάπτυξης, προτείνεται το Visual Studio Code, ένα open-source εργαλείο, το οποίο έχει ενσωματωμένο το Rust Language Server. Επίσης, διαθέτει και άλλες επεκτάσεις για τη γλώσσα Rust, οι οποίες βελτιώνουν την υποστήριξή της. Γίνεται εγκατάσταση και παρουσιάζονται οι βασικές λειτουργίες του εργαλείου. Τέλος, γίνεται παρουσίαση δύο ενδεικτικών ασκήσεων. Η πρώτη άσκηση προτείνεται να αφορά τη δημιουργία τυχαίου password με 8 αλφαριθμητικούς χαρακτήρες. Η δεύτερη άσκηση προτείνεται να αφορά τη συμπίεση δεδομένων ενός φακέλου σε ένα αρχείο τύπου tar (tarball) και αποσυμπίεση του συμπιεσμένου αρχείου.

2.4.E.6 | Η ΓΛΩΣΣΑ PERL ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ

Σε αυτή την υποενότητα, παρουσιάζονται η γλώσσα Perl και το πεδίο εφαρμογής της, το οποίο είναι η επεξεργασία μεγάλου όγκου δεδομένων (από διάφορες πηγές), η ανάλυση κειμένου (π.χ. η ανάλυση log files), η ανάπτυξη scripts για τη διαχείριση συστήματος (π.χ. έλεγχος δικαιωμάτων πρόσβασης) και η δημιουργία CGI (Common Gateway Interface) σεναρίων, και τα πλεονεκτήματα της γλώσσας Perl έναντι των άλλων γλωσσών, τα οποία, μεταξύ άλλων, περιλαμβάνουν ισχυρές δυνατότητες επεξεργασίας κειμένου, με αποτέλεσμα να είναι κατάλληλη για αναζητήσεις μοτίβων, αντικαταστάσεις, διαχωρισμό κειμένου και πολλές άλλες λειτουργίες, σχετικές με το κείμενο. Επιπλέον, επιτρέπει την εκτέλεση εξωτερικών εντολών και διεργασιών συστήματος, κάτι που την καθιστά κατάλληλη για διαχείριση συστήματος και αυτοματισμούς. Ως ολοκληρωμένο περιβάλλον ανάπτυξης, προτείνεται το Padre, ένα open-source εργαλείο. Γίνεται εγκατάσταση και παρουσιάζονται οι βασικές λειτουργίες του εργαλείου. Εναλλακτικά, χρησιμοποιείται και το Visual Studio Code, το οποίο έχει χρησιμοποιηθεί στο πλαίσιο του μαθήματος για άλλες γλώσσες. Τέλος, γίνεται παρουσίαση δύο ενδεικτι-

κών ασκήσεων. Η πρώτη άσκηση προτείνεται να αφορά τη δημιουργία προγράμματος, το οποίο μετράει πόσες φορές εμφανίζεται μια λέξη μέσα σε ένα αρχείο. Η λέξη δίνεται από τον/τη χρήστη/-τρια μέσω φόρμας html. Το αποτέλεσμα εκτυπώνεται, επίσης, στη σελίδα. Η δεύτερη άσκηση προτείνεται να αφορά τη δημιουργία προγράμματος, το οποίο αντικαθιστά μια λέξη με μια άλλη μέσα σε ένα αρχείο. Οι λέξεις δίνονται από τον/τη χρήστη/-τρια μέσω φόρμας html.

2.4.E.7 | Η ΓΛΩΣΣΑ XML ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ

Εδώ, παρουσιάζεται η Xml (Extensible Markup Language) και το πεδίο εφαρμογής της, το οποίο είναι η περιγραφή της δομής δεδομένων (data structure) και η μορφοποίηση δεδομένων και χρησιμοποιείται κυρίως σε ιστοσελίδες, αλλά και για ανταλλαγή δεδομένων μεταξύ διαφορετικών συστημάτων. Παρουσιάζονται τα πλεονεκτήματα της Xml έναντι των άλλων γλωσσών, τα οποία, μεταξύ άλλων, περιλαμβάνουν τη δυνατότητα διαμόρφωσης (customization) της δομής δεδομένων. Η Xml είναι επεκτάσιμη και επιτρέπει στους/στις χρήστες/-τριες να καθορίσουν τη δομή των δεδομένων τους, με τη χρήση Xml Schema ή DTD (Document Type Definition), οι οποίες ορίζουν τα πεδία και τους τύπους δεδομένων. Επιπροσθέτως, επιτρέπει τη διακήρυξη της δομής δεδομένων (structure declaration), που συμβάλλει στην εύκολη και αποτελεσματική ανταλλαγή δεδομένων. Επίσης, η Xml υποστηρίζεται από πολλές πλατφόρμες και εφαρμογές, καθιστώντας την κατάλληλη για τη διασύνδεση και την ανταλλαγή δεδομένων μεταξύ διαφορετικών συστημάτων. Δεν υπάρχει συγκεκριμένο ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) που σχεδιάστηκε αποκλειστικά για την Xml. Η Xml χρησιμοποιείται κυρίως για τη διακήρυξη και τη δομή δεδομένων, και δεν απαιτεί ένα IDE για ανάπτυξη. Παρόλα αυτά, υπάρχουν προγράμματα για την επεξεργασία Xml αρχείων. Προτείνεται το Visual Studio Code, το οποίο έχουν ήδη διδαχθεί οι εκπαιδευόμενοι/-ες και διαθέτει αρκετές επεκτάσεις για την εύκολη διαχείριση των αρχείων Xml. Τέλος, γίνεται παρουσίαση δύο ενδεικτικών ασκήσεων. Η πρώτη άσκηση προτείνεται να αφορά τη δημιουργία ατζέντας με εργασίες. Η δεύτερη άσκηση προτείνεται να αφορά τη δημιουργία βιβλιοθήκης μέσων (Media Library).

2.4.E.8 | Η ΓΛΩΣΣΑ RUBY ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ

Στην υποενότητα αυτή, παρουσιάζεται η γλώσσα Ruby και το πεδίο εφαρμογής της, το οποίο είναι η ανάπτυξη ιστοσελίδων, web εφαρμογών και εφαρμογών IoT (Internet of Things) και γενικότερα εφαρμογών που επικοινωνούν με συσκευές και αισθητήρες. Χρησιμοποιείται, επίσης, για την ανάπτυξης APIs (Application Programming Interfaces), για αλληλεπίδραση με άλλες εφαρμογές. Παρουσιάζονται τα πλεονεκτήματα της Ruby έναντι των άλλων γλωσσών, τα οποία, μεταξύ άλλων, περιλαμβάνουν: τη δυνατότητα μεταπρογραμματισμού (metaprogramming), δηλαδή τη δημιουργία κώδικα, που δημιουργεί και τροποποιεί δυναμικά κλάσεις και αντικείμενα, την αυτόματη διαχείριση μνήμης, ελαχιστοποιώντας τον κίνδυνο σφαλμάτων (π.χ. ανεπαρ-

κούς μνήμη-memory leaks), και την υποστήριξη πολυγλωσσικότητας, με αποτέλεσμα να χρησιμοποιείται για την ανάπτυξη πολυγλωσσικών εφαρμογών. Γίνεται αναφορά και στο Ruby on Rails, ένα πλαίσιο για την ανάπτυξη web εφαρμογών, το οποίο χρησιμοποιείται σε συνδυασμό με τη γλώσσα Ruby. Ως ολοκληρωμένο περιβάλλον ανάπτυξης, προτείνεται το Visual Studio Code, το οποίο έχουν ήδη διδαχθεί οι εκπαιδευόμενοι/-ες και διαθέτει αρκετές επεκτάσεις για τον αποδοτικότερο προγραμματισμό σε Ruby. Επίσης, παρουσιάζεται το εργαλείο RubyMine, το οποίο όμως δεν παρέχεται δωρεάν. Τέλος, γίνεται παρουσίαση δύο ενδεικτικών ασκήσεων. Η πρώτη άσκηση προτείνεται να αφορά τη δημιουργία chatbot, το οποίο θα δίνει πέντε πιθανές απαντήσεις, αναλόγως τι θα ρωτήσει ο/η χρήστης/-τρια (ενδεικτικές απαντήσεις: «Σίγουρα», «Ωραία», «Μείνε ήσυχος», «Ξέρω τι κάνω» και «Ηρέμησε»). Η δεύτερη άσκηση προτείνεται να αφορά τη δημιουργία του παιχνιδιού ναρκαλιευτής. Σε έναν πίνακα (board) του παιχνιδιού ναρκαλιευτής, στον οποίο υπάρχουν μόνο οι θέσεις στις οποίες υπάρχουν οι νάρκες, θα πρέπει να υπολογιστούν τα νούμερα τα οποία μπαίνουν στις υπόλοιπες θέσεις.

2.4.E.9 | Η ΓΛΩΣΣΑ TYPESCRIPT ΚΑΙ ΤΟ ΠΕΔΙΟ ΕΦΑΡΜΟΓΗΣ ΤΗΣ

Τέλος, παρουσιάζεται η γλώσσα Typescript, η οποία είναι ένα υποσύνολο της JavaScript και υποστηρίζει μια ευρεία γκάμα τύπων δεδομένων, καθώς και ενσωματωμένη αντιστοίχιση τύπων (type interface). Επίσης, διαθέτει εκτενή βιβλιοθήκη εργαλείων για την ανάπτυξη web εφαρμογών (όπως το Angular και το React). Το πεδίο εφαρμογής της Typescript είναι η ανάπτυξη web εφαρμογών και η ανάπτυξη εφαρμογών Single Page (SPAs), που αλληλεπιδρούν με τον/τη χρήστη/-τρια χωρίς ανανεώσεις σελίδας. Τα πλεονεκτήματα της Typescript περιλαμβάνουν: το εκτεταμένο σύστημα τύπων δεδομένων και τις βιβλιοθήκες με προηγμένες δυνατότητες ανάπτυξης web εφαρμογών. Ως ολοκληρωμένο περιβάλλον ανάπτυξης, προτείνεται το Visual Studio Code, το οποίο έχουν ήδη διδαχθεί οι εκπαιδευόμενοι/-ες και διαθέτει ενσωματωμένη υποστήριξη Typescript και αρκετές επεκτάσεις για τον αποδοτικότερο προγραμματισμό σε Typescript. Επίσης, το Visual Studio Code διαθέτει δυνατότητα αυτοσυμπληρούμενου κώδικα (autocompletion) για την Typescript. Τέλος, γίνεται παρουσίαση δύο ενδεικτικών ασκήσεων. Η πρώτη άσκηση προτείνεται να αφορά την ανάλυση ενός αρχείου που περιέχει παρενθέσεις της μορφής [], {}, () και να γίνεται επαλήθευση ότι όλα τα ζεύγη έχουν αντιστοιχηθεί σωστά. Η δεύτερη άσκηση προτείνεται να αφορά τη δημιουργία ενός γράφου, του οποίου οι κορυφές θα αντιπροσωπεύουν πόλεις της Ελλάδας. Οι ακμές του θα αντιπροσωπεύουν δρόμους που συνδέουν τις πόλεις. Στη συνέχεια, θα υπολογίζεται η συντομότερη διαδρομή μεταξύ δύο πόλεων.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Μπάγκος, Π. (2015). *Βιοπληροφορική*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/5028>
2. Νικολαΐδης, Β. (2022). *Προγραμματισμός σε R*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://dx.doi.org/10.57713/kallipos-100>
3. Σαμψών, Δ. (2003). *Η Γλώσσα Σήμανσης XML*. [Πανεπιστημιακές εκδόσεις]: http://www.fme.aegean.gr/sites/default/files/dsampsom_xml_lectures-notes-dec2003.pdf

Συμπληρωματικές

1. Ο επίσημος ιστότοπος της γλώσσας R: <https://www.r-project.org/>
2. Ο επίσημος ιστότοπος της γλώσσας Swift: <https://www.swift.org/>
3. Ο επίσημος ιστότοπος της γλώσσας Kotlin: <https://kotlinlang.org/>
4. Ο επίσημος ιστότοπος της γλώσσας Go: <https://go.dev/>
5. Ο επίσημος ιστότοπος της γλώσσας Rust: <https://www.rust-lang.org/>
6. Ο επίσημος ιστότοπος της γλώσσας Perl: <https://www.perl.org/>
7. Ο επίσημος ιστότοπος της γλώσσας Xml: <https://www.xml.com/>
8. Ο επίσημος ιστότοπος της γλώσσας Ruby: <https://www.ruby-lang.org/>
9. Ο επίσημος ιστότοπος της γλώσσας Typescript: <https://www.typescriptlang.org/>

2.4.ΣΤ • ΔΙΑΘΕΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Περίληψη της μαθησιακής ενότητας

Η μαθησιακή ενότητα «Διαθεματική εργασία» εισάγει τους/τις εκπαιδευόμενους/-ες στη δημιουργία μιας εργασίας, η οποία θα ενσωματώνει τη συνολική εμπειρία τους από όλα τα εξάμηνα κατάρτισης. Ειδικότερα, ο/η εκπαιδευόμενος/-η θα επιλέγει ένα θέμα, σε συνεργασία με τους/τις εκπαιδευτές/-τριες και επιβλέποντες/-ουσες, και θα το υλοποιεί, πραγματοποιώντας έρευνα ή/και ακολουθώντας τις διευκρινιστικές οδηγίες των εκπαιδευτών/-τριών του. Η συγκεκριμένη μαθησιακή ενότητα δίνει το ελεύθερο, ώστε το επιλεγόμενο θέμα να αφορά είτε βιβλιογραφική είτε προγραμματιστική εργασία υλοποίησης, ενώ αποτελεί την πρώτη ευκαιρία για τον/την εκπαιδευόμενο/-η να δημιουργήσει μια επαγγελματικού επιπέδου εφαρμογή στο portfolio του, αποθηκεύοντάς την σε κάποιο online αποθετήριο (git κ.λπ.), έχοντας τη δυνατότητα, στη συνέχεια, να την αναρτήσει στο βιογραφικό σημείωμά του/της. Η διαδικασία εκπόνησης της διαθεματικής εργασίας θα επικουρείται από τις παρεμβάσεις/προτάσεις των εκπαιδευτών/-τριών και θα υλοποιείται από τους/τις σπουδαστές/-τριες σε ατομικό ή ομαδικό επίπεδο, το πλήθος του οποίου θα έχει ως μέγιστο επιτρεπόμενο όριο τα δύο μέλη ανά επιλεγόμενο

θέμα υλοποίησης. Στο πέρας του ακαδημαϊκού εξαμήνου, θα πρέπει να παρουσιάζεται η εργασία στην τελική μορφή της σε εκτυπωμένο κείμενο-δοκίμιο για τις βιβλιογραφικού τύπου εκπονήσεις, ή σε συνδυασμό τελικού binary αρχείου εφαρμογής, συνοδευόμενου από τεκμηρίωση (documentation) για τις υλοποιήσεις του Project, μαζί με αρχείο παρουσίασης των σημείων κλειδιών (milestones) κατά τη διαδικασία εκπόνησης.

Προσδοκώμενα μαθησιακά αποτελέσματα

Όταν ολοκληρώσουν τη μαθησιακή ενότητα, οι εκπαιδευόμενοι/-ες θα είναι ικανοί/-ές να:

- Διακρίνουν τα πλεονεκτήματα της έρευνας και της διασταύρωσης στοιχείων από αξιόπιστες επιστημονικές πηγές,
- Αναπτύσσουν βιβλιογραφικές ή/και προγραμματιστικές εφαρμογές,
- Διαχειρίζονται εργαλεία versioning μέσα από αποθετήρια υλοποίησης κώδικα,
- Αποφασίζουν για τη χρήση των πληροφοριών που θα χρησιμοποιούν και τον τρόπο επεξεργασίας τους,
- Συνεισφέρουν με πρωτότυπα θέματα και πρόσθετα στην κοινότητα των επιστημονικών κειμένων της διεθνούς επιστημονικής βιβλιογραφίας,
- Υιοθετούν θετική στάση ως προς την ανάπτυξη επαγγελματικού επιπέδου εργασιών.

Βασικές έννοιες και λέξεις-κλειδιά

Διαχείριση έργων (Project management) / Ανάλυση εργασιών (Work breakdown structure) / Πακέτο εργασίας (Work package) / Δραστηριότητα (Activity) / Σημείο κλειδί (Milestone) / Διάγραμμα Gantt / Βιβλιογραφία / Σύστημα αναφοράς APA.

Αριθμός ωρών διδασκαλίας της μαθησιακής ενότητας ανά εβδομάδα

Θεωρία (0), Εργαστήριο (2), Σύνολο (2).

Διάρθρωση των μαθησιακών ενοτήτων σε υποενότητες

A/A	ΤΙΤΛΟΣ ΜΑΘΗΣΙΑΚΗΣ ΥΠΟΕΝΟΤΗΤΑΣ
1	ΘΕΜΑΤΟΛΟΓΙΑ ΚΑΙ ΓΕΝΙΚΕΣ ΟΔΗΓΙΕΣ
2	ΕΙΣΑΓΩΓΗ ΣΤΗ ΔΙΑΧΕΙΡΙΣΗ ΕΡΓΩΝ
3	ΣΧΕΔΙΑΣΜΟΣ ΕΡΓΟΥ
4	ΥΛΟΠΟΙΗΣΗ ΕΡΓΟΥ
5	ΣΥΝΤΑΞΗ ΤΕΛΙΚΗΣ ΑΝΑΦΟΡΑΣ
6	ΠΑΡΟΥΣΙΑΣΗ ΚΑΙ ΔΗΜΟΣΙΕΥΣΗ ΕΡΓΟΥ

2.4.ΣΤ.1 | ΘΕΜΑΤΟΛΟΓΙΑ ΚΑΙ ΓΕΝΙΚΕΣ ΟΔΗΓΙΕΣ

Στόχος της υποενότητας αυτής είναι η επιλογή του θέματος της εργασίας και η παρουσίαση των προδιαγραφών και των οδηγιών της διαθεματικής εργασίας που θα υλοποιήσουν οι εκπαιδευόμενοι/-ες. Η θεματολογία της εργασίας μπορεί να αφορά ένα ή και περισσότερα θέματα που καλύφθηκαν καθ' όλη τη διάρκεια σπουδών. Η εργασία μπορεί να είναι είτε βιβλιογραφική είτε προγραμματιστική. Ως βιβλιογραφική εργασία, νοείται η έρευνα των σπουδαστών/-τριών σε ένα θέμα της επιλογής τους, η σύνθεση των ευρημάτων τους και η εξαγωγή συμπερασμάτων. Ενδεικτικά, αναφέρονται: η έρευνα για εκπαιδευτικά παιχνίδια εκμάθησης κώδικα, η έρευνα για ιατρικές εφαρμογές για φορητές συσκευές. Ως προγραμματιστική εργασία, νοείται η ανάπτυξη ενός προγράμματος. Ενδεικτικά, αναφέρονται: ανάπτυξη μιας διαδικτυακής εφαρμογής, ενός παιχνιδιού ή μιας Android εφαρμογής. Σημειώνεται, επίσης, ότι η εργασία θα μπορεί να είναι είτε ατομική είτε ομαδική, με όριο τα δύο άτομα ανά ομάδα. Στη συνέχεια, τονίζεται ότι για όλες τις εργασίες, είτε βιβλιογραφικές είτε προγραμματιστικές, θα πρέπει στο τέλος να παραδοθεί μια αναφορά. Προτείνεται να γίνει μια παρουσίαση της εργασίας μέσα στην τάξη. Ακολουθεί μια σύντομη αναφορά στη δομή της εργασίας, η οποία προτείνεται να περιέχει τις εξής ενότητες: σκοπός της εργασίας, ανάλυση εργασιών και χρονοδιάγραμμα υλοποίησης, σχεδιασμός, αποτελέσματα, συμπεράσματα και βιβλιογραφία. Τέλος, οι σπουδαστές/-τριες χωρίζονται σε ομάδες και διαλέγουν θέματα εργασίας. Τα θέματα δύναται είτε να προταθούν από τον/την εκπαιδευτικό είτε από τους/τις σπουδαστές/-τριες, και να εγκριθούν στη συνέχεια από τον/την εκπαιδευτικό. Στη συνέχεια, οι σπουδαστές/-τριες τεκμηριώνουν γραπτώς το αντικείμενο και τους στόχους της εργασίας του. Η αναφορά αυτή θα είναι ένα από τα βασικά κεφάλαια της τελικής εργασίας.

2.4.ΣΤ.2 | ΕΙΣΑΓΩΓΗ ΣΤΗ ΔΙΑΧΕΙΡΙΣΗ ΕΡΓΩΝ

Στόχος της υποενότητας αυτής είναι η εξοικείωση των εκπαιδευομένων με τις βασικές έννοιες της διαχείρισης έργων, ώστε να μπορούν να σχεδιάσουν και να υλοποιήσουν μια εργασία. Αρχικά, γίνεται εισαγωγή στη διαχείριση έργων, παρουσιάζεται η έννοια του έργου και οι βασικές φάσεις ενός έργου (σύλληψη ιδέας, σχεδιασμός, υλοποίηση και ολοκλήρωση) και αναλύεται η έννοια της ανάλυσης των εργασιών του έργου (work breakdown structure), η έννοια των πακέτων εργασίας (work-packages), των δραστηριοτήτων (activities ή tasks) και των παραδοτέων (deliverables). Στη συνέχεια, παρουσιάζεται ο χρονοπρογραμματισμός (time planning) του έργου, το διάγραμμα Gantt και η έννοια των σημείων-κλειδιών (milestones). Ακολουθούν παραδείγματα ανάλυσης εργασιών έργων ανάπτυξης λογισμικού και έργων βιβλιογραφικής έρευνας. Σημειώνεται ότι η ανάλυση εργασιών βασίζεται στις τέσσερις βασικές φάσεις ενός έργου που παρουσιάστηκαν παραπάνω. Στη συνέχεια, οι εκπαιδευόμενοι/-ες καλού-

νται να «σπάσουν» το έργο τους (δηλαδή την εργασία τους) σε δραστηριότητες, να αναπτύξουν το χρονοδιάγραμμα υλοποίησης του έργου, να ορίσουν τα παραδοτέα και τα βασικά milestones του έργου. Τέλος, οι εκπαιδευόμενοι/-ες θα συντάξουν γραπτή αναφορά, η οποία θα περιέχει την ανάλυση των εργασιών του έργου, τη διάσπαση του έργου σε δραστηριότητες και τον χρονοπρογραμματισμό του έργου. Η αναφορά αυτή θα είναι ένα από τα βασικά κεφάλαια της τελικής εργασίας.

2.4.ΣΤ.3 | ΣΧΕΔΙΑΣΜΟΣ ΕΡΓΟΥ

Στόχος της υποενότητας αυτής είναι η εξοικείωση των εκπαιδευομένων με την αναγκαιότητα της σχεδίασης μιας εφαρμογής ή μιας έρευνας πριν από την υλοποίησή της. Στη φάση της σχεδίασης, καθορίζεται πώς θα υλοποιηθεί το αντικείμενο και οι στόχοι του έργου. Η υλοποίηση του έργου πραγματοποιείται με βάση αυτόν τον σχεδιασμό. Για τις προγραμματιστικές εργασίες, η φάση της σχεδίασης περιλαμβάνει: τον καθορισμό των προδιαγραφών της εφαρμογής, δηλαδή ποιες λειτουργικότητες θα καλύπτει, τον σχεδιασμό της δομής δεδομένων, τον σχεδιασμό της λογικής της εφαρμογής (application logic), δηλαδή τους κανόνες και τις διαδικασίες λειτουργίας της, τον σχεδιασμό των γραφικών διεπαφών χρήστη και τον σχεδιασμό της φυσικής αρχιτεκτονικής (εάν απαιτείται). Για τις βιβλιογραφικές εργασίες, η φάση σχεδίασης περιλαμβάνει: καθορισμό των λέξεων-κλειδιών, βάσει των οποίων θα διεξαχθεί η αναζήτηση, επιλογή των διαδικτυακών εργαλείων αναζήτησης που θα χρησιμοποιηθούν (π.χ. το Google, το Google Scholar ή διάφορα άλλα ψηφιακά αποθετήρια), και ο καθορισμός των κριτηρίων επιλογής της βιβλιογραφίας. Τα κριτήρια επιλογής της βιβλιογραφίας χωρίζονται σε ποιοτικά κριτήρια και κριτήρια σχετικότητας. Τα ποιοτικά κριτήρια αφορούν την αξιοπιστία των διαδικτυακών πηγών, και τα κριτήρια σχετικότητας αφορούν την απόφαση εάν ένα αποτέλεσμα αναζήτησης είναι σχετικό με το θέμα έρευνας. Τέλος, οι εκπαιδευόμενοι/-ες θα συντάξουν γραπτή αναφορά, η οποία θα περιέχει τον σχεδιασμό του έργου τους. Η αναφορά αυτή θα είναι ένα από τα βασικά κεφάλαια της τελικής εργασίας.

2.4.ΣΤ.4 | ΥΛΟΠΟΙΗΣΗ ΕΡΓΟΥ

Η υποενότητα αυτή αφορά την υλοποίηση του έργου από τους/τις εκπαιδευομένους/-ες, βάσει του αναλυτικού σχεδιασμού που πραγματοποίησαν στο προηγούμενο βήμα. Για τις προγραμματιστικές εργασίες, η φάση της υλοποίησης αφορά την ανάπτυξη της εφαρμογής σύμφωνα με τον τεχνικό σχεδιασμό. Επίσης, αφορά την εκσφαλμάτωση της εφαρμογής και τις διορθώσεις/βελτιώσεις (εάν απαιτηθεί). Τέλος, η φάση της υλοποίησης της εφαρμογής περιλαμβάνει και τη διενέργεια ελέγχων, ώστε να διαπιστωθεί η ορθή λειτουργία της εφαρμογής και η συμμόρφωσή της, σύμφωνα με τον σχεδιασμό. Για τις βιβλιογραφικές εργασίες, η φάση σχεδίασης περιλαμβάνει την εκπόνηση της βιβλιογραφικής έρευνας, σύμφωνα με τις λέξεις κλειδιά που έχουν επιλεγεί. Η αναζήτηση θα γίνει στα διαδικτυακά εργαλεία αναζήτησης που έχουν επιλεγεί στη φάση του σχεδιασμού. Στη συνέχεια, θα γίνει επιλογή της

βιβλιογραφίας τόσο με τα ποιοτικά κριτήρια όσο και με τα κριτήρια σχετικότητας που έχουν οριστεί στη φάση του σχεδιασμού. Η φάση της υλοποίησης της βιβλιογραφικής αναφοράς περιλαμβάνει τη μελέτη και παρουσίαση των επιλεγμένων αποτελεσμάτων. Τέλος, περιλαμβάνει τη σύνθεση των αποτελεσμάτων της έρευνας. Η σύνθεση παρέχει μια συνολική εικόνα της έρευνας, αναδεικνύει τις εξελίξεις, τις αντιφάσεις και τα κενά στη βιβλιογραφία, και προσφέρει κριτική ανάλυση και ερμηνεία των ευρημάτων. Τέλος, οι εκπαιδευόμενοι/-ες θα συντάξουν γραπτή αναφορά, η οποία θα περιέχει λεπτομέρειες της υλοποίησης. Οι προγραμματιστικές εργασίες θα παρουσιάζουν την εφαρμογή που αναπτύχθηκε και πιθανά προβλήματα κατά την υλοποίηση, καθώς και τα αποτελέσματα των ελέγχων. Οι βιβλιογραφικές εργασίες θα παρουσιάζουν τα αποτελέσματα της έρευνας, τη σύνθεση των αποτελεσμάτων και την εξαγωγή συμπερασμάτων.

2.4.ΣΤ.5 | ΣΥΝΤΑΞΗ ΤΕΛΙΚΗΣ ΑΝΑΦΟΡΑΣ

Η υποενότητα αυτή αφορά τη σύνταξη της γραπτής αναφοράς της εργασίας. Αρχικά, τονίζεται στους/στις εκπαιδευόμενους/-ες η σπουδαιότητα της γραπτής αναφοράς, η οποία τεκμηριώνει τα αποτελέσματα ενός έργου. Η περιγραφή της δομής της αναφοράς έχει αναφερθεί στην πρώτη υποενότητα και περιλαμβάνει: τον σκοπό της εργασίας, την ανάλυση εργασιών και τα χρονοδιάγραμμα υλοποίησης, τον σχεδιασμό, τα αποτελέσματα, τα συμπεράσματα και τη βιβλιογραφία. Τα περισσότερα από τα κεφάλαια αυτά τα έχουν συντάξει οι εκπαιδευόμενοι/-ες κατά τη διάρκεια του εξαμήνου. Εδώ, δίνεται η δυνατότητα για διορθώσεις. Στη συνέχεια, παρουσιάζονται οι καλές πρακτικές για τη σύνταξη μιας επαγγελματικής αναφοράς. Αναλυτικότερα, η αναφορά θα πρέπει να διαθέτει εξώφυλλο, περιεχόμενα και βιβλιογραφία, ομοιόμορφη μορφοποίηση κειμένου, σωστή στοίχιση και σωστή αρίθμηση εικόνων και πινάκων. Γίνεται αναφορά στην τεχνική αυτόματης δημιουργίας περιεχομένων, στη χρήση στυλ και στην αυτόματη αρίθμηση αντικειμένων για τον επεξεργαστή κειμένου που θα χρησιμοποιήσουν οι εκπαιδευόμενοι/-ες. Ιδιαίτερη σημασία δίνεται στη βιβλιογραφία. Τονίζεται στους/στις εκπαιδευόμενους/-ες η σπουδαιότητα μιας σωστής και αξιόπιστης βιβλιογραφίας στη σύνταξη μιας αναφοράς. Παρουσιάζεται ο τρόπος σύνταξης μιας βιβλιογραφικής αναφοράς, σύμφωνα με το σύστημα APA. Επίσης, παρουσιάζονται οι τεχνικές ανάπτυξης βιβλιογραφίας σε έναν επεξεργαστή κειμένου και η χρήση βιβλιογραφικών αναφορών μέσα στο κείμενο. Οι εκπαιδευόμενοι/-ες στην υποενότητα αυτή καλούνται να συντάξουν την τελική αναφορά, σύμφωνα με τις καλές πρακτικές που παρουσιάστηκαν, και να ενσωματώσουν τη βιβλιογραφία που χρησιμοποίησαν στην αναφορά τους. Επίσης, καλούνται να σχολιάσουν εάν η υλοποίηση έγινε σύμφωνα με το χρονοδιάγραμμα και εάν υπήρξαν αποκλίσεις από αυτό.

2.4.ΣΤ.6 | ΠΑΡΟΥΣΙΑΣΗ ΚΑΙ ΔΗΜΟΣΙΕΥΣΗ ΕΡΓΟΥ

Σε αυτό το σημείο, θα πρέπει να παρουσιαστεί το έργο μέσα στην τάξη και να δημοσιευθεί σε ένα online αποθετήριο. Το πρώτο θέμα της υποενότητας είναι η παρουσίαση του έργου των εκπαιδευομένων. Τονίζεται ιδιαίτερα η σημασία των δεξιοτήτων σύνταξης και εκπόνησης επαγγελματικών παρουσιάσεων. Παρουσιάζονται οι καλές πρακτικές για τη σύνταξη μιας επαγγελματικής παρουσίασης, γίνεται αναφορά στα έτοιμα πρότυπα (templates) παρουσιάσεων, τα οποία χρησιμοποιούνται ιδιαίτερα, και Προτείνεται οι εκπαιδευόμενοι/-ες να εκπονήσουν μια σύντομη παρουσίαση της εργασίας τους (περίπου δέκα λεπτά), την οποία θα παρουσιάσουν μέσα στην τάξη. Το δεύτερο θέμα της υποενότητας είναι η δημοσίευση του έργου σε κάποιο online αποθετήριο. Σημειώνεται εδώ ότι μπορούν να δημοσιευθούν τόσο οι προγραμματιστικές όσο και οι βιβλιογραφικές εργασίες. Γίνεται σύντομη αναφορά στα ψηφιακά αποθετήρια και στη σημασία τους για τους/τις νέους/-ες επαγγελματίες. Στη συνέχεια, οι εκπαιδευόμενοι/-ες καλούνται να αποφασίσουν ποια κομμάτια της εργασίας τους επιθυμούν να δημοσιεύσουν στο Διαδίκτυο (τον κώδικα της εφαρμογής, την αναφορά ή την παρουσίαση). Το πιο δημοφιλές online αποθετήριο για την απόθεση κώδικα είναι το Github (<https://github.com/>). Γίνεται σύντομη αναφορά στο Github και παρουσιάζεται η διαδικασία δημιουργίας αποθετηρίου (repository) για την απόθεση κώδικα και την τεκμηρίωσή του. Για τη δημοσίευση μιας βιβλιογραφικής εργασίας, μπορεί επίσης να χρησιμοποιηθεί το Github. Στη συνέχεια, η ηλεκτρονική διεύθυνση του αποθετηρίου δύναται να χρησιμοποιηθεί από τον/την εκπαιδευόμενο/-η (π.χ. στο βιογραφικό του). Τέλος, ειδικά για τη δημοσίευση στο Διαδίκτυο βιβλιογραφικών εργασιών, προτείνεται η παρουσίαση εναλλακτικών αποθετηρίων, όπως είναι το Google drive ή το Dropbox, τα οποία παράγουν links, που μπορούν να χρησιμοποιηθούν από τους/τις εκπαιδευόμενους/-ες.

Προτεινόμενες πηγές μελέτης

Κύριες

1. Kendall, K. & Kendall, J. (2023). *Ανάλυση & Σχεδίαση Συστημάτων*. Αθήνα: Εκδόσεις Μ. Γκιούρδα.
2. Phillips, J. (2007). *Διαχείριση έργων πληροφορικής*. Αθήνα: Εκδόσεις Μ. Γκιούρδα.

Συμπληρωματικές

1. Τσιρίκας, Α. (2021). *Οργάνωση & Διοίκηση Έργων*. Αθήνα: Εκδόσεις Rosili.
2. Φιτσιλής, Π. (χ.χ.). *Σχεδιασμός και Διαχείριση Έργων* [Ανοιχτά μαθήματα του Οικονομικού Πανεπιστημίου Αθηνών]: <https://eclass.aueb.gr/courses/INF194/>

3 | Διδακτική μεθοδολογία

Στο πλαίσιο των εκπαιδευτικών συναντήσεων, αξιοποιείται η συμμετοχική ή/και βιωματική διδασκαλία. Έχοντας ως σημείο εκκίνησης τις βασικές αρχές εκπαίδευσης ενηλίκων αλλά και τη σύνδεση της αρχικής επαγγελματικής κατάρτισης με το πραγματικό περιβάλλον εργασίας, η εκπαίδευση έχει ένα διπλό σημείο αναφοράς: την ενεργή ανταπόκριση στις μαθησιακές ανάγκες της συγκεκριμένης κάθε φορά ομάδας εκπαιδευομένων και τις ανάγκες που προκύπτουν στο περιβάλλον εργασίας της συγκεκριμένης ειδικότητας.

Ο/Η εκπαιδευτής/-τρια οργανώνει και καθοδηγεί την εκπαιδευτική πράξη, επιλύει τυχόν ανακύπτοντα προβλήματα, υποστηρίζει, ανατροφοδοτεί και ενδυναμώνει τους/τις εκπαιδευόμενους/-ες. Διαμεσολαβεί, διευκολύνει και ενισχύει τη διαδικασία μάθησης, σε ομαδικό και σε ατομικό επίπεδο, συνδέοντας την κατάρτιση με τον κόσμο της εργασίας.

Η συμμετοχική και βιωματική εκπαίδευση διαμορφώνει ένα δημιουργικό περιβάλλον μάθησης και ενισχύει την αλληλεπίδραση εκπαιδευτή/-τριας και εκπαιδευομένων. Προσφέρει τη δυνατότητα να γίνουν αντιληπτές αλλά και να αξιοποιηθούν στην εκπαιδευτική διαδικασία οι ανάγκες, οι ιδιαιτερότητες, οι δυνατότητες, οι γνώσεις, οι δεξιότητες και οι εμπειρίες της συγκεκριμένης ομάδας των καταρτιζομένων. Προσφέρει τη δυνατότητα να γίνουν πρακτικές και ρεαλιστικές συνδέσεις με το πραγματικό περιβάλλον εργασίας της συγκεκριμένης ειδικότητας.

Την υποστήριξη ενός αλληλεπιδραστικού περιβάλλοντος μάθησης ενισχύει η χρήση σύντομων εμπλουτισμένων εισηγήσεων και η συχνή εφαρμογή συμμετοχικών εκπαιδευτικών τεχνικών και μέσων. Ενδεικτικά, αναφέρουμε ότι η ενίσχυση της συμμετοχής των καταρτιζομένων υποβοηθείται ενεργά με την αξιοποίηση απλών τεχνικών όπως ο καταγισμός ιδεών, οι ερωτήσεις-απαντήσεις ή η συζήτηση, οι ατομικές ή/και ομαδικές ασκήσεις εφαρμογής ή επίλυσης προβλήματος, η προσομοίωση, η εργασία σε ομάδες, η μελέτη της εκάστοτε περίπτωσης. Οι εκπαιδευτικές δραστηριότητες που αξιοποιούν τις παραπάνω ή ανάλογες εκπαιδευτικές τεχνικές αντλούν τα θέματά τους μέσα από τη θεματολογία της κάθε μαθησιακής ενότητας και τα σχετικά ζητήματα που συνδέονται με το πραγματικό περιβάλλον εργασίας.

Η εκπαίδευση σε συγκεκριμένες –ατομικές ή/και ομαδικές– δραστηριότητες μέσα στην τάξη και στα εργαστήρια προετοιμάζει τα μέλη της ομάδας για τη συμμετοχή τους στην πρακτική άσκηση/μαθητεία. Η σταδιακή εξειδίκευση της γνώσης, η ανάπτυξη συγκεκριμένων δεξιοτήτων/ικανοτήτων, καθώς και η καλλιέργεια κατάλληλων στάσεων και συμπεριφορών σε κρίσιμα ζητήματα δεοντολογίας, υγείας και ασφάλειας κ.λπ., προετοιμάζουν τη συγκεκριμένη κάθε φορά ομάδα εκπαιδευομένων για τα επόμενα βήματα. Το πρόγραμμα κατάρτισης συνδυάζει την απόκτηση θεωρητικών γνώσεων με την ανάπτυξη αναγκαίων πρακτικών δεξιοτήτων για την αποτελεσματική άσκηση του επαγγέλματος.

Σε ανάλογη κατεύθυνση, στο πλαίσιο της πρακτικής εφαρμογής της ειδικότητας, δίνεται και η δυνατότητα ανάπτυξης διαθεματικών προγραμμάτων/σχεδίων δραστηριοτήτων (projects), με σύγχρονη εφαρμογή διαφορετικών μαθησιακών ενοτήτων και θεματικών. Οι συγκεκριμένες δραστηριότητες μπορούν να αναπτύσσονται σε μεγαλύτερη ή μικρότερη χρονική έκταση και να συμπεριλαμβάνουν, ενδεικτικά, επισκέψεις σε χώρους εργασίας και εγκαταστάσεις παραγωγής, συναντήσεις με έμπειρους επαγγελματίες της ειδικότητας ή ειδικούς του συγκεκριμένου παραγωγικού τομέα και κλάδου, υλοποίηση ομαδικών εργασιών με συνδυασμό διαφορετικών μαθησιακών ενοτήτων υπό την καθοδήγηση ομάδας εκπαιδευτών/-τριών ή ακόμη και δημιουργία ομάδων εκπαιδευομένων με στόχο την αμοιβαία άσκηση, μελέτη και αλληλοδιδασκαλία. Το σύνολο των παραπάνω δραστηριοτήτων μπορεί να αξιοποιηθεί και αυτόνομα, ανεξάρτητα δηλαδή από την υλοποίηση ενός συνολικότερου project.

Η ανάπτυξη συνεργατικού κλίματος μεταξύ των εκπαιδευομένων και η καλλιέργεια σχέσεων εμπιστοσύνης με τους/τις εκπαιδευτές/-τριες συντελούν στη δημιουργία μιας κοινότητας μάθησης η οποία μπορεί να αποδειχθεί ιδιαίτερα σημαντική για την ενίσχυση των διεργασιών μάθησης τόσο κατά τη διάρκεια της εκπαιδευτικής διαδικασίας όσο και πέρα από αυτήν, σε περιόδους μελέτης ή/και προετοιμασίας για τις διαδικασίες αξιολόγησης/πιστοποίησης των γνώσεων και δεξιοτήτων που έχουν αποκτηθεί.

Η ενίσχυση της συνεργασίας και της δημιουργίας μιας κοινότητας μάθησης είναι δυνατόν να επηρεάσει και τους τρόπους μελέτης των ίδιων των εκπαιδευομένων προς μια πιο ενεργητική, δημιουργική στάση διερεύνησης και εξοικείωσης με τη γνώση. Η τήρηση προσωπικών σημειώσεων και η ανταλλαγή τους μεταξύ των εκπαιδευομένων, η μελέτη πρόσθετων βιβλιογραφικών –ελληνόφωνων και ξενόγλωσσων– πηγών, η αναζήτηση συμπληρωματικών ερεθισμάτων στο ευρύτερο κοινωνικό και επαγγελματικό περιβάλλον (παρακολούθηση οπτικοακουστικού υλικού, επαγγελματικών περιοδικών, εξέλιξης θεσμικού πλαισίου κ.λπ.), σχετικών με την επαγγελματική ειδικότητα, είναι δυνατόν να αποτελέσουν σημαντικά συστατικά για τη διαμόρφωση μιας ισχυρής σχέσης με την εξειδικευμένη επαγγελματική γνώση και τις δεξιότητες που απαιτεί το επάγγελμα.

Η παρακολούθηση ενός εντατικού προγράμματος κατάρτισης μεγάλης διάρκειας αναγκαστικά επιφέρει αλλαγές στην καθημερινότητα των εκπαιδευομένων. Η αποδοτική διαχείριση του καθημερινού χρόνου απαιτεί προσωπικό προγραμματισμό και φροντίδα. Η συστηματική πρόβλεψη ενός χρονικού διαστήματος μελέτης, καθημερινά, καθώς και η τήρηση ενός ημερολογίου μάθησης όπου οι εκπαιδευόμενοι/-ες θα καταγράφουν την καθημερινή τους πρόοδο, αλλά ενδεχομένως θα σημειώνουν και ερωτήματα ή παρατηρήσεις σχετικές με την κάθε μαθησιακή ενότητα, μπορούν να αποτελέσουν χρήσιμες και πολύ δημιουργικές πρακτικές ενίσχυσης των διαδικασιών μάθησης.

Η φοίτηση στο Ινστιτούτο Επαγγελματικής Κατάρτισης είναι περίοδος διαμόρφωσης μιας μελλοντικής επαγγελματικής ταυτότητας. Είναι η αρχή μιας νέας προσωπικής ιστορίας και μιας διαρκούς εξέλιξης μπροστά σε απαιτήσεις και αναγκαιότητες

που χαρακτηρίζουν την κάθε επαγγελματική ειδικότητα. Είναι η ευκαιρία διασύνδεσης του/της κάθε εκπαιδευόμενου/-ης με τις γνώσεις, τις δεξιότητες και τις στάσεις που είναι κοινές στην επαγγελματική κοινότητα στην οποία σταδιακά θα ενταχθεί.

4 | Υγεία και ασφάλεια κατά τη διάρκεια της κατάρτισης

Για την προστασία των καταρτιζομένων, τόσο στην αίθουσα διδασκαλίας και στους εργαστηριακούς χώρους του ΙΕΚ όσο και στις επιχειρήσεις που επισκέπτονται για την υλοποίηση της πρακτικής άσκησης/μαθητείας, τηρούνται όλες οι προβλεπόμενες διατάξεις για τους κανόνες υγείας και ασφάλειας στην ειδικότητα και το επάγγελμα, αλλά και ευρύτερα, όπως προβλέπονται ιδίως από:

- Τον κώδικα νόμων για την υγεία και την ασφάλεια των εργαζομένων (βλ. Ν. 3850/2010), όπως ισχύει.
- Τις διατάξεις του κτιριοδομικού κανονισμού (βλ. 3046/304/89-ΦΕΚ 59/Δ/3-02-89), όπως ισχύει.
- Τον κανονισμό λειτουργίας των εργαστηριακών κέντρων (ΦΕΚ 1318 Β'/2015), όπως ισχύει.
- Το ΦΕΚ 3938/Β/26-8-2021, κοινή Υπουργική Απόφαση υπ' αριθμ. Κ5/97484 με θέμα την «Πρακτική άσκηση σπουδαστών Ινστιτούτων Επαγγελματικής Κατάρτισης αρμοδιότητας Υπουργείου Παιδείας και Θρησκευμάτων», όπως ισχύει.
- Το ΦΕΚ 4146/Β/9-9-2021, κοινή Υπουργική Απόφαση υπ' αριθμ. ΦΒ7/108652/Κ3, με θέμα το «Πλαίσιο Ποιότητας Μαθητείας», όπως ισχύει.

Παρακάτω παρατίθενται οι βασικοί κανόνες υγείας και ασφάλειας, καθώς και ο σχετικός αναγκαίος ατομικός εξοπλισμός για τις συνθήκες άσκησης της ειδικότητας:

4.1 • Βασικοί κανόνες υγείας και ασφάλειας

Οι χώροι κατάρτισης (αίθουσες διδασκαλίας και εργαστηριακοί χώροι) πρέπει να διαθέτουν όλες τις απαραίτητες προδιαγραφές που εξασφαλίζουν την ασφάλεια και την υγεία των καταρτιζομένων και των εκπαιδευτών/-τριών.

Οι αίθουσες πρέπει να πληρούν τις παρακάτω προδιαγραφές:

- Παροχή επαρκούς φυσικού φωτισμού και πρόβλεψη για επαρκή παροχή μεθόδων τεχνητού φωτισμού, ο οποίος θα οδηγεί έως την έξοδο των κτιριακών εγκαταστάσεων σε περιπτώσεις διακοπής ρεύματος,
- Παροχή επαρκούς αερισμού με σύγχρονες μονάδες εξαερισμού,
- Αδιάλειπτη παροχή θέρμανσης/ψύξης κατά την πλήρη διάρκεια του ακαδημαϊκού έτους, ώστε να διασφαλίζεται ένα ελάχιστο επίπεδο επαρκούς προστασίας σε καιρικά φαινόμενα και περιόδους ψύχους ή/και καύσωνα,
- Πρόβλεψη επαρκούς αριθμού εξόδων κινδύνου, βάσει της σχετικής νομοθεσίας περί κανονισμών πυροπροστασίας,
- Παροχή συστήματος πυρασφάλειας-πυρανίχνευσης-πυρόσβεσης,

- Πρόβλεψη για χώρο απαραίτητου εξοπλισμού για την παροχή πρώτων βοηθειών σε περίπτωση ατυχήματος,
- Πρόβλεψη για παροχή αναρτημένου πίνακα χρήσιμων τηλεφώνων σε περίπτωση ατυχήματος (τηλεφωνικών αριθμών Πρώτων Βοηθειών, Πυροσβεστικής Υπηρεσίας, Άμεσης Δράσης κ.λπ.),
- Δεν επιτρέπεται η χρήση φαγητού και ποτού εντός του χώρου των εργαστηρίων, ώστε να αποφευχθεί βραχυκύκλωμα των υλικών μερών από υπολείμματα φαγητού ή σταγόνων ροφήματος, με απώτερο σκοπό την προστασία της ανθρώπινης ζωής,
- Πρόβλεψη για παροχή ράμπας σε όλες τις περιπτώσεις που η μετακίνηση εντός της αίθουσας μέσω δαπέδου δεν είναι εφικτή για άτομα με κινητικά προβλήματα, με σκοπό την προστασία της ανθρώπινης ζωής και υγείας και την αποφυγή ατυχήματος,
- Παροχή κάδου απορριμμάτων έξω από τον χώρο διδασκαλίας, ο οποίος ιδανικά σε κάθε διάλειμμα θα πρέπει να καθαρίζεται,
- Πρόβλεψη για προειδοποιητικά μηνύματα εντός του εργαστηρίου περί μη τοποθέτησης συσκευών GSM/GPRS κινητού τηλεφώνου κατά την ώρα της παρουσίας καταρτιζομένων εντός των εργαστηρίων, καθώς υπάρχει κίνδυνος, λόγω του φαινομένου του συντονισμού και της ηλεκτρομαγνητικής ακτινοβολίας που εκπέμπεται από τις GSM συσκευές, να υποστούν βλάβη τα υλικά μέρη του ηλεκτρονικού υπολογιστή, με κίνδυνο την απώλεια δεδομένων των πραγματειών των καταρτιζομένων.

4.2 • Μέσα ατομικής προστασίας

Δεν απαιτούνται συγκεκριμένα μέσα προστασίας για τους/τις καταρτιζόμενους/-ες της ειδικότητας «Τεχνικός Λογισμικού Η/Υ».



Προδιαγραφές υλοποίησης πρακτικής άσκησης

1 | Ο θεσμός της πρακτικής άσκησης

Η πρακτική άσκηση συνδέεται άρρηκτα με τη θεωρητική κατάρτιση, αφού, κατά της διάρκειάς της, οι πρακτικά ασκούμενοι/-ες ανακαλούν τη θεωρητική και εργαστηριακή γνώση για να την εφαρμόσουν στην πράξη και να αντεπεξέλθουν στις εργασίες που τους ανατίθενται. Καλούνται να αναλάβουν συγκεκριμένα καθήκοντα και να δώσουν λύση σε πρακτικά προβλήματα που ανακύπτουν, υπό την εποπτεία των εκπαιδευτών/-τριών. Έτσι, ο θεσμός της πρακτικής άσκησης στοχεύει στην ανάπτυξη επαγγελματικών ικανοτήτων/δεξιοτήτων σχετικών με την ειδικότητα, στην ενίσχυση της επαφής με τον εργασιακό χώρο και την προετοιμασία των εκπαιδευομένων για την παραγωγική διαδικασία, μέσω της απόκτησης εμπειριών, ιδιαίτερα χρήσιμων για τη μετέπειτα επαγγελματική τους πορεία.

Αναλυτικότερα, η πρακτική άσκηση είναι υποχρεωτική για τους/τις εκπαιδευόμενους/-ες των Σχολών Ανώτερης Επαγγελματικής Κατάρτισης (ΣΑΕΚ) και θεωρείται απαραίτητη προϋπόθεση για την απόκτηση Βεβαίωσης Επαγγελματικής Κατάρτισης (Άρθρο 27 του Ν. 4763/2020 για το Εθνικό Σύστημα Επαγγελματικής Εκπαίδευσης, Κατάρτισης και Διά Βίου Μάθησης).

Στη συνέχεια, αναφέρονται χρήσιμες πληροφορίες για τον θεσμό της πρακτικής άσκησης, όπως περιγράφονται στη σχετική νομοθεσία⁴, και που αφορούν τις βασικές προϋποθέσεις, τον τρόπο και τους όρους υλοποίησής της.

Διάρκεια πρακτικής άσκησης

Η συνολική διάρκεια της περιόδου πρακτικής άσκησης είναι εννιακόσιες εξήντα (960) ώρες. Οι ώρες πρακτικής ανά ημέρα καθορίζονται σε τέσσερις (4) έως οκτώ (8) ανάλογα με τη φύση και το αντικείμενο της ειδικότητας κατάρτισης του/της ασκούμενου/-ης. Δεν επιτρέπεται η υπέρβαση του ημερήσιου ωραρίου πέραν των ωρών που ορίζονται στην ειδική σύμβαση πρακτικής άσκησης.

Όροι υλοποίησης πρακτικής άσκησης

Η πρακτική άσκηση δύναται να πραγματοποιείται σε θέσεις που προσφέρονται από φυσικά πρόσωπα, ΝΠΔΔ, ΝΠΙΔ, δημόσιες υπηρεσίες, ΟΤΑ α' και β' βαθμού και επιχειρήσεις. Εξαιρούνται οι φορείς:

- α) Προσωρινής απασχόλησης
- β) Τα νυχτερινά κέντρα
- γ) Παροχής καθαριότητας και φύλαξης

4 Οι πληροφορίες που περιλαμβάνονται στο Μέρος Δ' του εκπαιδευτικού εγχειριδίου και που αφορούν τη διάρκεια της πρακτικής άσκησης, τους όρους υλοποίησης, τα δικαιώματα και τις υποχρεώσεις των πρακτικά ασκούμενων προέρχονται από το ΦΕΚ 3938/Β/26-8-2021. Κοινή Υπουργική Απόφαση υπ' αριθμ. Κ5/97484. Πρακτική άσκηση σπουδαστών Ινστιτούτων Επαγγελματικής Κατάρτισης αρμοδιότητας Υπουργείου Παιδείας και Θρησκευμάτων.

- δ) Τα πρακτορεία τυχερών παιχνιδιών
- ε) Κάθε επιχείρηση στην οποία δεν είναι εφικτός ο έλεγχος της εκπαίδευσης από τον αρμόδιο φορέα.

Ο/Η εκπαιδευόμενος/-η ΣΑΕΚ, προκειμένου να πραγματοποιήσει πρακτική άσκηση, υπογράφει ειδική σύμβαση πρακτικής άσκησης με τον/την εργοδότη/-τρια, η οποία θεωρείται από το ΣΑΕΚ φοίτησης. Η ειδική σύμβαση πρακτικής άσκησης δεν συνιστά σύμβαση εξαρτημένης εργασίας.

Βασικός συντελεστής για την επιτυχή υλοποίηση της πρακτικής άσκησης είναι και ο/η εκπαιδευτής/-τρια της επιχείρησης ή υπηρεσίας ο/η οποίος/-α αναλαμβάνει την παρακολούθηση και υποστήριξη των ασκουμένων. Σε αυτή την κατεύθυνση, ο/η εργοδότης/-τρια ορίζει έμπειρο στέλεχος συναφούς επαγγελματικής ειδικότητας με τον/την πρακτικά ασκούμενο/-η ως «εκπαιδευτή/-τρια στον χώρο εργασίας», ο/η οποίος/-α αναλαμβάνει την αποτελεσματική υλοποίηση των εκπαιδευτικών δραστηριοτήτων στον χώρο εργασίας και την παρακολούθηση της προόδου του/της πρακτικά ασκουμένου/-ης.

Η παρακολούθηση της προόδου του/της πρακτικά ασκουμένου/-ης γίνεται μέσω του βιβλίου πρακτικής άσκησης. Αναλυτικότερα, σε αυτό καταγράφει ο/η ίδιος/-α ο/η ασκούμενος/-η κατά εβδομάδα τις εργασίες με τις οποίες ασχολήθηκε, αλλά και τα καθήκοντα που του/της ανατέθηκαν στον χώρο πραγματοποίησης της πρακτικής άσκησης. Κάθε εβδομαδιαία καταχώρηση ελέγχεται και υπογράφεται από τον/την εκπαιδευτή/-τρια στον χώρο εργασίας.

2 | Οδηγίες για τον/την πρακτικά ασκούμενο/-η

2.1 • Προϋποθέσεις εγγραφής στο πρόγραμμα πρακτικής άσκησης

Η πρακτική άσκηση είναι υποχρεωτική για τους/τις εκπαιδευόμενους/-ες των Σχολών Ανώτερης Επαγγελματικής Κατάρτισης και θεωρείται απαραίτητη προϋπόθεση για την απόκτηση Βεβαίωσης Επαγγελματικής Κατάρτισης.

Για την έναρξη της πρακτικής άσκησης στην ειδικότητα «Τεχνικός Λογισμικού Η/Υ», οι εκπαιδευόμενοι/-ες πρέπει να έχουν συμπληρώσει το εξάμηνο Β΄ της φοίτησης στις ΣΑΕΚ. Στην περίπτωση αυτή, μπορούν πια να τοποθετηθούν σε θέση πρακτικής της ειδικότητας τους.

2.2 • Δικαιώματα και υποχρεώσεις του/της πρακτικά ασκουμένου/-ης

Βασική προϋπόθεση για την επιτυχή υλοποίηση ενός προγράμματος πρακτικής άσκησης είναι η γνώση και η εφαρμογή των δικαιωμάτων και των υποχρεώσεων κάθε εμπλεκόμενου μέλους όπως ορίζονται στην εκάστοτε ισχύουσα νομοθεσία⁵. Στη συνέχεια παρατίθενται κάποια δικαιώματα και υποχρεώσεις των πρακτικά ασκούμενων.

Δικαιώματα πρακτικά ασκούμενων

1. Τμηματική ή συνεχόμενη υλοποίηση της πρακτικής άσκησης.
2. Δυνατότητα αποζημίωσης, η οποία ορίζεται στο 80% του νόμιμου, νομοθετημένου, κατώτατου ορίου του ημερομισθίου του/της ανειδίκευτου/-ης εργάτη/-τριας, ή όπως αυτό διαμορφώνεται από το Υπουργείο Εργασίας και Κοινωνικών Υποθέσεων ή αναλογικά εάν η ημερήσια διάρκεια της πρακτικής είναι μικρότερη των οκτώ (8) ωρών. Η αποζημίωση καταβάλλεται στον/στην πρακτικά ασκούμενο/-η μετά την ολοκλήρωση της πρακτικής άσκησης. Σε περίπτωση μη δυνατότητας χρηματοδότησης της αποζημίωσης της πρακτικής άσκησης, δεν υφίσταται η υποχρέωση αποζημίωσής της, παρά μόνο η υποχρέωση του/της εργοδότη/-τριας να αποδίδει τις προβλεπόμενες ασφαλιστικές εισφορές.

5 Οι πληροφορίες που περιλαμβάνονται στο μέρος Δ΄ του εκπαιδευτικού εγχειριδίου και που αφορούν τη διάρκεια της πρακτικής άσκησης, τους όρους υλοποίησης, τα δικαιώματα και τις υποχρεώσεις των πρακτικά ασκούμενων προέρχονται από το ΦΕΚ 3938/Β/26-8-2021. Κοινή Υπουργική Απόφαση υπ' αριθμ. Κ5/97484. Πρακτική άσκηση σπουδαστών Ινστιτούτων Επαγγελματικής Κατάρτισης αρμοδιότητας Υπουργείου Παιδείας και Θρησκευμάτων.

3. Υπαγωγή στην ασφάλιση του e-ΕΦΚΑ (πρώην ΙΚΑ-ΕΤΑΜ) για τον κλάδο του ατυχήματος. Για την ασφάλισή του/της καταβάλλονται οι προβλεπόμενες από την παρ. 1 του άρθρου 10 του Ν. 2217/1994 (Α' 83) ασφαλιστικές εισφορές, οι οποίες βαρύνουν το φυσικό ή νομικό πρόσωπο (εργοδότης/-τρια) στο οποίο υλοποιείται η πρακτική άσκηση.
4. Δικαίωμα αναφοράς στη ΣΑΕΚ σε περίπτωση μη τήρησης των όρων πρακτικής άσκησης.
5. Δικαίωμα διακοπής πρακτικής άσκησης βάσει τεκμηρίωσης και έπειτα από σχετική δήλωση στη ΣΑΕΚ εποπτείας.
6. Αλλαγή εργοδότη/-τριας, εφόσον συντρέχει τεκμηριωμένος σοβαρότατος λόγος.
7. Οι πρακτικά ασκούμενοι/-ες δεν απασχολούνται την Κυριακή και τις επίσημες αργίες.

Υποχρεώσεις πρακτικά ασκουμένων

1. Τήρηση του ημερήσιου ωραρίου πρακτικής άσκησης, όπως ορίζεται στην ειδική σύμβαση.
2. Τήρηση των όρων υγείας και ασφάλειας του/της εργοδότη/-τριας.
3. Σεβασμός της κινητής και ακίνητης περιουσίας του/της εργοδότη/-τριας.
4. Αρμονική συνεργασία με τα στελέχη του/της εργοδότη/-τριας.
5. Προσκόμιση –όπου απαιτείται– όλων των απαραίτητων ιατρικών βεβαιώσεων για την εξάσκηση του επαγγέλματος.
6. Προσκόμιση στη ΣΑΕΚ των απαραίτητων δικαιολογητικών, πριν από την έναρξη και μετά τη λήξη της πρακτικής άσκησης, αλλά και σε περίπτωση διακοπής της.
7. Ενημέρωση σε περίπτωση απουσίας του/της ασκουμένου/-ης της επιχείρησης και της ΣΑΕΚ εποπτείας.
8. Τήρηση βιβλίου πρακτικής άσκησης, το οποίο διατίθεται από τη ΣΑΕΚ και στο οποίο αναγράφονται από τους/τις ασκούμενους/-ες κατά εβδομάδα οι εργασίες με τις οποίες ασχολήθηκαν και τα καθήκοντα που τους ανατέθηκαν στον χώρο πραγματοποίησης της πρακτικής άσκησης.
9. Προσκόμιση στο τέλος κάθε μήνα στη ΣΑΕΚ φοίτησης ή εποπτείας της πρακτικής άσκησης του βιβλίου πρακτικής άσκησης για έλεγχο.
10. Υποβολή μετά την ολοκλήρωση της πρακτικής άσκησης του βιβλίου πρακτικής άσκησης στη ΣΑΕΚ φοίτησης, συμπληρωμένου με τις εβδομαδιαίες εκθέσεις, τον χρόνο και το αντικείμενο απασχόλησης, τις ημέρες απουσίας, και την επίδοση του/της πρακτικά ασκουμένου/-ης. Υποβολή του εντύπου λήξης (Βεβαίωσης Παρουσίας) της πρακτικής άσκησης, συμπληρωμένου, υπογεγραμμένου και σφραγισμένου από τον/την εργοδότη/-τρια-νόμιμο/-η εκπρόσωπο του φορέα απασχόλησης, στο οποίο βεβαιώνεται ότι ο/η εκπαιδευόμενος/-η πραγματοποίησε την πρακτική άσκηση στην επιχείρηση/οργανισμό, καθώς και το χρονικό διάστημα αυτής.

11. Άμεση ενημέρωση της ΣΑΕΚ φοίτησης από τον/την πρακτικά ασκούμενο/-η σε περίπτωση διακοπής της πρακτικής άσκησης και προσκόμιση του βιβλίου πρακτικής και του εντύπου της λήξης (Βεβαίωσης Παρουσίας) με τις ημέρες πρακτικής άσκησης που έχουν πραγματοποιηθεί. Για να συνεχίσει ο/η εκπαιδευόμενος/-η την πρακτική άσκηση, για το υπόλοιπο του προβλεπόμενου διαστήματος, στον ίδιο ή σε άλλο φορέα απασχόλησης (εργοδότη/-τρια), θα πρέπει να ακολουθηθεί εκ νέου η διαδικασία έναρξης πρακτικής. Αν η διακοπή της πρακτικής άσκησης γίνει από τον/την εργοδότη/-τρια, τότε οφείλει ο/η τελευταίος/-α να ενημερώσει άμεσα τη ΣΑΕΚ φοίτησης του/της πρακτικά ασκούμενου/-ης.

2.3 • Φορείς υλοποίησης πρακτικής άσκησης

Κάθε πρακτικά ασκούμενος/-η πραγματοποιεί την πρακτική άσκηση σε τμήματα των φορέων απασχόλησης αντίστοιχα με την ειδικότητα του/της, με την εποπτεία του υπεύθυνου φορέα, ειδικότητας αντίστοιχης με το αντικείμενο κατάρτισής του/της.

Συγκεκριμένα, στην ειδικότητα «Τεχνικός Λογισμικού Η/Υ» οι εκπαιδευόμενοι/-ες πραγματοποιούν πρακτική άσκηση σε τομείς που σχετίζονται με τις Τεχνολογίες της Πληροφορικής και των Επικοινωνιών (ΤΠΕ), σε φορείς/επιχειρήσεις, όπως φορείς του δημοσίου και του ιδιωτικού τομέα που διαθέτουν τμήματα ή/και διευθύνσεις Πληροφορικής, και σε θέσεις εργασίας όπως: Προγραμματιστής (C Developer), Προγραμματιστής JAVA (JAVA Developer), Προγραμματιστής C++ (C++ Developer), Προγραμματιστής C# (C# Developer), Προγραμματιστής Python (Python Developer), Προγραμματιστής .NET (.NET Developer), Προγραμματιστής Front-End (Front-End Developer), Προγραμματιστής Back-End (Back-End Developer), Προγραμματιστής PHP (PHP Developer), Προγραμματιστής Εφαρμογών Κινητών Συσκευών (Mobile App Developer), Προγραμματιστής Διαδικτυακών Εφαρμογών (Web Developer), Προγραμματιστής JavaScript (JavaScript Developer), Σχεδιαστής Διαδικτυακών Εφαρμογών (Web Designer), Σχεδιαστής UX/UI (UX/UI Designer), Διαχειριστής Βάσεων Δεδομένων (Database Administrator), Προγραμματιστής Βάσεων Δεδομένων (Database Developer), Αναλυτής Συστημάτων (System Analyst), Διαχειριστής Ηλεκτρονικού Καταστήματος (E-shop Administrator), Μηχανικός Δοκιμών και Διασφάλισης Ποιότητας (QA/Test Engineer) κ.λπ.

3 | Ο ρόλος του/της εκπαιδευτή/-τριας της πρακτικής άσκησης

Ο/Η εργοδότης/-τρια της επιχείρησης που προσφέρει θέση πρακτικής άσκησης ορίζει ένα έμπειρο στέλεχος συναφούς επαγγελματικής ειδικότητας με τον/την πρακτικά ασκούμενο/-η ως «εκπαιδευτή/-τρια στον χώρο εργασίας», ο/η οποίος/-α αναλαμβάνει την αποτελεσματική υλοποίηση των εκπαιδευτικών δραστηριοτήτων στον χώρο εργασίας, την παρακολούθηση της προόδου των εκπαιδευομένων και την ανατροφοδότηση των υπεύθυνων εκπαιδευτών στην εκπαιδευτική δομή.

Αναλυτικότερα, ο/η εκπαιδευτής/-τρια είναι το συνδετικό πρόσωπο του/της εργοδότη/-τριας της επιχείρησης με την εκπαιδευτική δομή (ΣΑΕΚ) και, κατά συνέπεια, έχει συνεχή συνεργασία με αυτήν. Επιπλέον, ο ρόλος αφορά την παροχή συμβουλών, πληροφοριών ή καθοδήγησης, καθώς πρόκειται για ένα άτομο με χρήσιμη εμπειρία, δεξιότητες και εξειδίκευση, το οποίο υποστηρίζει την προσωπική και επαγγελματική ανάπτυξη των πρακτικά ασκουμένων.

4 | Προσδοκώμενα μαθησιακά αποτελέσματα της πρακτικής άσκησης

Κατά τη διάρκεια της πρακτικής άσκησης, επιδιώκεται η αναβάθμιση των γνώσεων, επαγγελματικών δεξιοτήτων και ικανοτήτων των σπουδαστών/-τριών ΣΑΕΚ, με αποτέλεσμα την ομαλή μετάβασή τους από την αίθουσα κατάρτισης στον χώρο εργασίας, και μάλιστα κάτω από πραγματικές εργασιακές συνθήκες. Στο πλαίσιο αυτής της μετάβασης και της ομαλής ένταξης, οι πρακτικά ασκούμενοι/-ες καλούνται να καλλιεργήσουν όχι μόνο επαγγελματικές δεξιότητες που αφορούν την ειδικότητα και οι οποίες δεν εξαντλούνται στο πλαίσιο της αίθουσας κατάρτισης, αλλά και οριζόντιες δεξιότητες που ενισχύουν την επαγγελματική τους συμπεριφορά και καλλιεργούν την περιβαλλοντική αλλά και επιχειρηματική κουλτούρα. Έτσι, η πρακτική άσκηση αποτελεί ένα προπαρασκευαστικό στάδιο κατά το οποίο αναβαθμίζονται οι γενικές και ειδικές γνώσεις, συντελούνται σημαντικές διεργασίες επαγγελματικού προσανατολισμού και διευκολύνεται η επαγγελματική ανάπτυξη του ατόμου.

Αναλυτικότερα, κατά τη διάρκεια της πρακτικής άσκησης δίνεται η δυνατότητα στον/στην πρακτικά ασκούμενο/-η να ασκηθεί στις εργασίες που απορρέουν από τα επιμέρους μαθησιακά αποτελέσματα του προγράμματος κατάρτισης στην ειδικότητα. Η άσκηση μπορεί να επιτευχθεί μέσω της παρατήρησης της εργασίας, της συμμετοχής σε ομάδα εκτέλεσης της εργασίας, της καθοδηγούμενης εργασίας ή της δοκιμής/αυτόνομης εκτέλεσης της εργασίας από τον/την πρακτικά ασκούμενο/-η.

Στον πίνακα που ακολουθεί, αποτυπώνονται οι ενότητες προσδοκώμενων μαθησιακών αποτελεσμάτων της πρακτικής άσκησης για την ειδικότητα «Τεχνικός Λογισμικού Η/Υ» και οι αντίστοιχες ενδεικτικές εργασίες και ειδικές προδιαγραφές ανά ενότητα κατά τη διάρκεια της πρακτικής άσκησης. Επισημαίνεται ότι οι εν λόγω εργασίες μπορούν να χρησιμοποιηθούν από τους/τις πρακτικά ασκούμενους/-ες για τη συμπλήρωση του βιβλίου πρακτικής άσκησης.

Πίνακας 3 /// Ενότητες προσδοκώμενων μαθησιακών αποτελεσμάτων

ΕΝΟΤΗΤΕΣ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ		
ΕΝΟΤΗΤΕΣ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	ΕΝΔΕΙΚΤΙΚΕΣ ΕΡΓΑΣΙΕΣ ΚΑΤΑ ΤΗ ΔΙΑΡΚΕΙΑ ΤΗΣ ΠΡΑΚΤΙΚΗΣ ΑΣΚΗΣΗΣ	ΕΙΔΙΚΕΣ ΠΡΟΔΙΑΓΡΑΦΕΣ
Α. «Χρήση συστημάτων λογισμικού και διαχείριση ψηφιακών τεχνολογιών»	- Αναγνώριση των διαφόρων κατηγοριών υλικού υπολογιστών, - Αναγνώριση της δομής του υπολογιστή και της λειτουργικότητας των διαφόρων υποσυστημάτων του, - Αξιολόγηση διαφόρων συνθέσεων υλικού και λογισμικού σύμφωνα με τις κατάλληλες προδιαγραφές, - Εξήγηση των δυνατοτήτων του κοινού λογισμικού για εφαρμογές γραφείου και βοηθητικών εργαλείων λογισμικού, - Διαχείριση διαφορετικών λειτουργικών συστημάτων και λογισμικού εφαρμογών γραφείου, - Ανάπτυξη τρόπων βέλτιστης παραμετροποίησης των λειτουργικών συστημάτων, - Εγκατάσταση και αναβάθμιση λογισμικού συστημάτων (system Software) και περιφερειακών συσκευών, - Εγκατάσταση και συντήρηση λογισμικού εφαρμογών σύμφωνα με τις οδηγίες των κατασκευαστών του, - Διατήρηση αντιγράφων ασφάλειας (backup) και επαναφορά (restore) δεδομένων από αντίγραφα ασφάλειας, θέτοντας σε κανονική κατάσταση λειτουργίας εγκαταστάσεις εφαρμογών, - Εκτέλεση διαγνωστικών ελέγχων σε δικτυακές εγκαταστάσεις υπολογιστών και αναγνώριση των αιτιών των βλαβών, - Χρήση υπηρεσιών νεφούπολογιστικής, - Παραμετροποίηση σύγχρονων πλατφορμών τηλεδιασκέψεων, - Επιλογή κατάλληλων μέτρων για τη φυσική και λογική προστασία των δεδομένων, - Αναγνώριση των ενδεχόμενων απειλών (ιοί, hackers) και λήψη των απαραίτητων μέτρων για την προστασία των συστημάτων λογισμικού, - Σύνταξη μελετών, εκθέσεων, οδηγιών υλοποίησης και τεκμηρίωσης, σχετικών με τις ΤΠΕ.	- Σύγχρονος Η/Υ με σύνδεση στο Διαδίκτυο, - Λογισμικό κειμενογράφου, - Λογισμικό υπολογιστικών φύλλων, - Λογισμικό παρουσιάσεων, - Λογισμικό τηλεδιασκέψεων, - Υπηρεσίες νεφούπολογιστικής.

συνέχεια >>

Πίνακας 3 /// Ενότητες προσδοκώμενων μαθησιακών αποτελεσμάτων

ΕΝΟΤΗΤΕΣ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ		
ΕΝΟΤΗΤΕΣ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	ΕΝΔΕΙΚΤΙΚΕΣ ΕΡΓΑΣΙΕΣ ΚΑΤΑ ΤΗ ΔΙΑΡΚΕΙΑ ΤΗΣ ΠΡΑΚΤΙΚΗΣ ΑΣΚΗΣΗΣ	ΕΙΔΙΚΕΣ ΠΡΟΔΙΑΓΡΑΦΕΣ
Β. «Προγραμματισμός και διαχείριση δεδομένων»	- Απόκτηση αντίληψης των διαφορετικών στυλ προγραμματισμού (διαδικαστικός, αντικειμενοστρεφής, συναρτησιακός), - Περιγραφή λύσεων σε υπολογιστικά βήματα για την αλγοριθμική αντιμετώπιση ενός προβλήματος, - Επιλογή των κατάλληλων προγραμματιστικών δομών για τις βασικές λειτουργίες των προγραμμάτων, - Χρήση των κυριότερων γλωσσών προγραμματισμού, - Σχεδίαση προγραμματιστικών λύσεων για επίλυση πραγματικών προβλημάτων, - Εκσφαλμάτωση προγραμμάτων που εμφανίζουν συντακτικά ή λογικά σφάλματα, - Βελτιστοποίηση των προγραμμάτων για την καλύτερη απόδοση, - Εφαρμογή τεχνικών αμυντικού προγραμματισμού, - Χρήση προγραμματιστικών πλαισίων (frameworks) και βιβλιοθηκών για τη διαχείριση και ανάλυση των δεδομένων, - Επεξεργασία δεδομένων με χρήση βάσεων δεδομένων, - Χρήση εργαλείων για την αναπαράσταση και διαχείριση συνδεδεμένων δεδομένων, - Έλεγχος της εισαγωγής δεδομένων από τους/τις χρήστες/-τριες, μεριμνώντας για την ασφάλεια και τη διατήρηση αυτών, - Εξαγωγή χρήσιμων συμπερασμάτων από τη διαχείριση των δεδομένων.	- Σύγχρονος Η/Υ με σύνδεση στο Διαδίκτυο, - Λογισμικό ολοκληρωμένου προγραμματιστικού περιβάλλοντος (IDE), - Λογισμικό διαχείρισης βάσεων δεδομένων, - Εγκατεστημένες βιβλιοθήκες γλωσσών προγραμματισμού και βάσεων δεδομένων.

συνέχεια >>

Πίνακας 3 /// Ενότητες προσδοκώμενων μαθησιακών αποτελεσμάτων

ΕΝΟΤΗΤΕΣ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ		
ΕΝΟΤΗΤΕΣ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	ΕΝΔΕΙΚΤΙΚΕΣ ΕΡΓΑΣΙΕΣ ΚΑΤΑ ΤΗ ΔΙΑΡΚΕΙΑ ΤΗΣ ΠΡΑΚΤΙΚΗΣ ΑΣΚΗΣΗΣ	ΕΙΔΙΚΕΣ ΠΡΟΔΙΑΓΡΑΦΕΣ
Γ. «Ανάπτυξη λογισμικού και διαδικτυακών εφαρμογών»	- Κατηγοριοποίηση των τεχνολογιών ανάπτυξης λογισμικού συστήματος, - Αποτίμηση της καταλληλότητας και αποδοτικότητας των προγραμμάτων, - Πρόβλεψη των κινδύνων που ενδέχεται να προκύψουν σε ένα έργο ανάπτυξης λογισμικού, - Ανάπτυξη διαδικτυακών εφαρμογών με χρήση HTML5 και CSS3, - Υλοποίηση διαδικτυακών εφαρμογών, χρησιμοποιώντας τεχνολογίες από την πλευρά του/της πελάτη/-ισσας (client-side), - Ανάπτυξη δυναμικών εφαρμογών του Παγκόσμιου Ιστού με τεχνολογίες από την πλευρά του διακομιστή (server-side) και βάσεις δεδομένων, - Σχεδίαση και υλοποίηση γραφικών περιβαλλόντων με βάση την προστασία των προσωπικών δεδομένων και την ασφάλεια της πληροφορίας (information security), - Ανάπτυξη διαδικτυακών εφαρμογών με εργαλεία διαχείρισης περιεχομένου (CMS), - Εφαρμογή τεχνικών ασφάλειας και παρακολούθησης της λειτουργίας και των στατιστικών χρήσης των διαδικτυακών εφαρμογών, - Παραγωγή τεκμηρίωσης της λειτουργικότητας μιας εφαρμογής, - Χρήση απομακρυσμένων συστημάτων ελέγχου εκδόσεων για ανάπτυξη λογισμικού.	- Σύγχρονος Η/Υ με σύνδεση στο Διαδίκτυο, - Λογισμικό σχεδίασης διαδικτυακών εφαρμογών, - Λογισμικό ολοκληρωμένου προγραμματιστικού περιβάλλοντος διαδικτυακών εφαρμογών (IDE), - Τοπικός ή/και απομακρυσμένος διαδικτυακός διακομιστής (web server), - Λογισμικό διαχείρισης βάσεων δεδομένων, - Λογισμικό μεταφοράς αρχείων, - Εγκατεστημένες βιβλιοθήκες γλωσσών προγραμματισμού και βάσεων δεδομένων, - Εργαλεία δοκιμών αποδοτικότητας και ανταποκρισιμότητας (web performance tools), - Εργαλεία ελέγχου εκδόσεων για ανάπτυξη λογισμικού.

Πίνακας 3 /// Ενότητες προσδοκώμενων μαθησιακών αποτελεσμάτων

ΕΝΟΤΗΤΕΣ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ		
ΕΝΟΤΗΤΕΣ ΠΡΟΣΔΟΚΩΜΕΝΩΝ ΜΑΘΗΣΙΑΚΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	ΕΝΔΕΙΚΤΙΚΕΣ ΕΡΓΑΣΙΕΣ ΚΑΤΑ ΤΗ ΔΙΑΡΚΕΙΑ ΤΗΣ ΠΡΑΚΤΙΚΗΣ ΑΣΚΗΣΗΣ	ΕΙΔΙΚΕΣ ΠΡΟΔΙΑΓΡΑΦΕΣ
Δ. «Ανάλυση συστημάτων και τεχνολογία λογισμικού»	• Αναγνώριση των βασικών εννοιών της τεχνολογίας λογισμικού (λογισμικό, εργαλεία, μεθοδολογίες, κύκλος ζωής κ.λπ.), • Περιγραφή των σταδίων του κύκλου ανάπτυξης και ζωής μιας εφαρμογής λογισμικού, • Αναγνώριση των απαιτήσεων μιας εφαρμογής λογισμικού, • Περιγραφή των απαιτήσεων μιας εφαρμογής λογισμικού, χρησιμοποιώντας τις αρχές της δομημένης ανάλυσης, • Αναπαράσταση των απαιτήσεων πελατών, χρησιμοποιώντας εργαλεία και μεθοδολογίες ανάλυσης και σχεδίασης, • Αναγνώριση των ελέγχων που διενεργούνται σε κάθε επίπεδο ελέγχου κώδικα λογισμικού (levels of testing), • Εξήγηση των διαφορετικών τεχνικών που χρησιμοποιούνται για τον έλεγχο ενός συστήματος λογισμικού, • Χρήσης εργαλείων ελέγχου ενός συστήματος λογισμικού, • Εφαρμογή των κατάλληλων μεθόδων εκσφαλμάτωσης (debugging), με σκοπό τη διόρθωση σφαλμάτων, • Εξήγηση των βασικών αρχών της ποιότητας ενός λογισμικού και πώς αυτή διασφαλίζεται, • Έλεγχος της ποιότητας ενός συστήματος λογισμικού, • Σχεδίαση βάσεων δεδομένων, χρησιμοποιώντας αποτελεσματικά τις τεχνικές του Σημαιολογικού, Λογικού και Φυσικού Σχεδιασμού, • Υλοποίηση μιας βάσης δεδομένων σε ένα σύγχρονο σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων, • Προσαρμογή του κώδικα και της δομής των βάσεων δεδομένων, ανάλογα με τις προδιαγραφές, • Ενσωμάτωση βάσεων δεδομένων σε εφαρμογές.	• Σύγχρονος Η/Υ με σύνδεση στο Διαδίκτυο, • Εργαλεία σχεδίασης βάσεων δεδομένων, • Λογισμικό διαχείρισης βάσεων δεδομένων, • Λογισμικό μεταφοράς αρχείων, • Εγκατεστημένες βιβλιοθήκες γλωσσών προγραμματισμού και βάσεων δεδομένων, • Εργαλεία υποβοήθησης της σχεδίασης και ανάπτυξης λογισμικού (Case Tools), • Εργαλεία ελέγχου λογισμικού.

5 | Αναλυτικό περιεχόμενο προγράμματος πρακτικής άσκησης

Κατά τη διάρκεια της πρακτικής άσκησης, ο/η πρακτικά ασκούμενος/-η θα έρθει σε επαφή με τις τεχνολογίες Πληροφορικής, τη σχεδίαση-ανάπτυξη και τη χρήση συστημάτων λογισμικού και διαδικτυακών εφαρμογών. Επίσης, θα εξοικειωθεί με τα θέματα ασφάλειας και τους κινδύνους που ενδέχεται να προκύψουν σε ένα έργο ανάπτυξης λογισμικού ή σε μια βάση δεδομένων. Αυτά είναι δυνατόν να τα επιτύχει μέσα από ενδεικτικές εργασίες εντός του εργασιακού περιβάλλοντος, οι οποίες μπορούν να εκτελεστούν από τον/την πρακτικά ασκούμενο/-η είτε μέσω της εξωτερικής παρατήρησης είτε μέσω της συμμετοχής ως μέλους εντός συγκεκριμένης ομάδας υλοποίησης έργου.

5.1.A • ΧΡΗΣΗ ΣΥΣΤΗΜΑΤΩΝ ΛΟΓΙΣΜΙΚΟΥ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗ ΨΗΦΙΑΚΩΝ ΤΕΧΝΟΛΟΓΙΩΝ

Μια ενδεικτική εργασία, κατά τη διάρκεια της πρακτικής άσκησης του/της πρακτικά ασκούμενου/-ης, είναι η εξοικείωση με το λογισμικό εφαρμογών γραφείου της επιχείρησης ή του οργανισμού, καθώς και με τα βοηθητικά εργαλεία λογισμικού που θα είναι ήδη εγκατεστημένα ή θα εγκατασταθούν στα συγκεκριμένα πληροφοριακά συστήματα (λογισμικό κειμενογράφου, υπολογιστικών φύλλων, παρουσιάσεων, βάσεων δεδομένων, τηλεδιασκέψεων κ.λπ.). Η αλληλεπίδραση του/της πρακτικά ασκούμενου/-ης με το σύνολο/μέρος των προγραμμάτων λογισμικού μπορεί να πραγματοποιηθεί μέσω σύγχρονου ηλεκτρονικού υπολογιστή, ο οποίος θα διαθέτει σύνδεση στο Διαδίκτυο και λογισμικό εφαρμογών γραφείου, το οποίο θα πρέπει να είναι εγκατεστημένο είτε τοπικά είτε διαδικτυακά σε υπηρεσίες νεφοϋπολογιστικής.

Μια επιπλέον ενδεικτική εργασία είναι η αναγνώριση από τον/την πρακτικά ασκούμενο/-η των εξαρτημάτων και των κατηγοριών υλικού υπολογιστών με τους οποίους δραστηριοποιείται η εταιρεία/οργανισμός, η επιλογή υλικού (Hardware) συγκεκριμένων προδιαγραφών σύμφωνα με τις ανάγκες των πελατών/-ισών ή του εργασιακού περιβάλλοντος, και η αναγνώριση των πιο συνηθισμένων βλαβών και δυσλειτουργιών σε υπολογιστικά συστήματα, μέσω της χρήσης εργαλείων(π.χ. πολύμετρο και δοκιμαστικό κατσαβίδι).

Τέλος, μια ακόμη ενδεικτική εργασία ενός/μίας πρακτικά ασκούμενου/-ης είναι η απарίθμηση των πιο διαδεδομένων λειτουργικών συστημάτων και των χαρακτηριστικών τους, και η επιλογή του καταλληλότερου λειτουργικού συστήματος που θα πρέπει να εγκατασταθεί ή/και να παραμετροποιηθεί, ανάλογα με την περίπτωση χρήσης του και τις ανάγκες που εκφράζονται από τους/τις τελικούς/-ές χρήστες/-τριες και πελάτες/-ισσες. Η εγκατάσταση των διαφόρων λειτουργικών συστημάτων θα πραγματοποιείται με τη χρήση μνημών αποθήκευσης (flash memories, DVD κ.ά.) iso αρχείων λειτουργικών συστημάτων.

Ιδιαίτερη προσοχή πρέπει να δοθεί στη λήψη των απαραίτητων μέτρων για την προστασία των δεδομένων των συστημάτων λογισμικού, διατηρώντας αντίγραφα ασφαλείας (backup) τα οποία θα χρησιμοποιηθούν για την επαναφορά (restore) των εφαρμογών και των δεδομένων τους σε κανονική κατάσταση λειτουργίας. Οι παραπάνω εργασίες θα πραγματοποιούνται με τη χρήση κατάλληλου λογισμικού και εξωτερικών μνημών αποθήκευσης (flash memories, DVD κ.ά.).

5.1.B • ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗ ΔΕΔΟΜΕΝΩΝ

Μια ενδεικτική εργασία, κατά τη διάρκεια της πρακτικής άσκησης του/της πρακτικά ασκουμένου/-ης, είναι η χρήση της κατάλληλης αλγοριθμικής τεχνικής για την επίλυση ενός πραγματικού προβλήματος. Με αυτόν τον τρόπο, ο/η πρακτικά ασκούμενος/-η θα σχεδιάσει μια προγραμματιστική λύση και θα επιλέξει την προσφορότερη γλώσσα προγραμματισμού με την οποία θα υλοποιήσει μια εφαρμογή γραφικού περιβάλλοντος και η οποία θα έχει τη δυνατότητα να αλληλοεπιδρά με το Διαδίκτυο.

Μια επιπλέον ενδεικτική εργασία, κατά τη διάρκεια της πρακτικής άσκησης του/της πρακτικά ασκουμένου/-ης, είναι η εφαρμογή της κατάλληλης τεχνικής αμυντικού προγραμματισμού, δημιουργώντας έτσι πιο ανθεκτικές και ασφαλείς εφαρμογές. Στο παραπάνω πλαίσιο, υπάγεται η τροποποίηση και βελτιστοποίηση του υπάρχοντα κώδικα των προγραμμάτων για την καλύτερη απόδοσή τους. Επίσης, επειδή η πολυπλοκότητα ενός προγράμματος είναι συνδεδεμένη με την ύπαρξη σφαλμάτων μέσα σε αυτό, ο/η πρακτικά ασκούμενος/-η θα επιλέξει την κατάλληλη τεχνική για την εκσφαλμάτωση (debugging) των συντακτικών και λογικών σφαλμάτων.

Επίσης, μια ενδεικτική εργασία είναι η επεξεργασία από τον/την πρακτικά ασκούμενο/-η των δεδομένων με τη χρήση των κατάλληλων βάσεων δεδομένων, που θα χρησιμοποιηθούν σε συστήματα διαχείρισης βάσεων δεδομένων αλλά και σε κάθε εξατομικευμένη διαδικτυακή εφαρμογή, μεριμνώντας για την ασφάλειά τους και την εξαγωγή χρήσιμων συμπερασμάτων. Η αναγνώριση των προβλημάτων ασφαλείας σε πληροφοριακά συστήματα καθώς και η διάκριση των διαφορών ανάμεσα στην εμπιστευτικότητα, την ακεραιότητα και τη διαθεσιμότητα της πληροφορίας κρίνεται ιδιαίτερα σημαντική.

Τέλος, μια ακόμη ενδεικτική εργασία ενός/μίας πρακτικά ασκουμένου/-ης είναι η εξοικείωση με την εισαγωγή περιεχομένου μεταξύ διαφορετικών συστημάτων λογισμικού, εξασφαλίζοντας την ομαλή μετάπτωση των δεδομένων. Υπό το πρίσμα των παραπάνω, θα ενσωματώσουν βάσεις δεδομένων σε ιστότοπους. Για την ολοκλήρωση των εν λόγω εργασιών, κρίνεται σημαντική η καταγραφή των σημείων ευπάθειας των πληροφοριακών συστημάτων και η εφαρμογή του Γενικού Κανονισμού για την προστασία των δεδομένων, τη διασφάλιση ζητημάτων ιδιωτικότητας και του απόρρητου των επικοινωνιών στο Διαδίκτυο και στο IoT, καθώς και η σύμπληυση (compliance) με τις προδιαγραφές του μοντέλου GDPR.

Μια ενδεικτική εργασία, κατά τη διάρκεια της πρακτικής άσκησης του/της πρακτικά ασκουμένου/-ης, είναι η χρήση της κατάλληλης γλώσσας για την ανάπτυξη μιας διαδικτυακής εφαρμογής, η οποία θα χρησιμοποιηθεί για την επίλυση ενός πραγματικού προβλήματος. Στο πλαίσιο της επιλογής των κατάλληλων στοιχείων για τη δημιουργία περιεχομένου, ο/η πρακτικά ασκούμενος/-η εξοικειώνεται με τις διαδικτυακές τεχνολογίες και τα εργαλεία που χρησιμοποιούνται από την επιχείρηση ή τον οργανισμό για την ανάπτυξη και διαχείριση του ιστοτόπου. Στη συνέχεια, θα μελετήσει τις υπάρχουσες ιστοσελίδες, τη δομή του ιστοτόπου, τον τρόπο διαχείρισης, τη μέθοδο φιλοξενίας, και τις τεχνολογίες που χρησιμοποιούνται (π.χ. HTML, CSS, JavaScript, PHP ή κάποιο πλαίσιο, όπως το Bootstrap, ή κάποιο εξειδικευμένο εργαλείο για e-shops). Αφού μελετήσει τον τρόπο που σχεδιάζεται και αναπτύσσεται ο ιστότοπος της επιχείρησης ή του οργανισμού, θα επιλέξει τα κατάλληλα στοιχεία και χαρακτηριστικά για τη δημιουργία περιεχομένου που θα χρησιμοποιηθεί στον Παγκόσμιο Ιστό.

Μια επιπλέον ενδεικτική εργασία, κατά τη διάρκεια της πρακτικής άσκησης του/της πρακτικά ασκουμένου/-ης, είναι η ανάπτυξη νέων ιστοσελίδων ή η βελτίωση/συντήρηση μιας υπάρχουσας ιστοσελίδας επιχείρησης/οργανισμού, μέσω των εργαλείων που χρησιμοποιούνται ήδη στην επιχείρηση ή στον οργανισμό. Κατά τις εργασίες ανάπτυξης συντήρησης, ο/η πρακτικά ασκούμενος/-η μελετά θέματα ασφάλειας των πληροφοριών, λαμβάνει υπόψη το ύψος του κειμένου και η εμφάνιση της ιστοσελίδας να ταιριάζει με τις υπόλοιπες ιστοσελίδες του οργανισμού, αλλά και θέματα συμβατότητας της ιστοσελίδας και ορθής προβολής σε διαφορετικά υπολογιστικά περιβάλλοντα.

Τέλος, μια ακόμη ενδεικτική εργασία από τον/την πρακτικά ασκούμενο/-η είναι η εξοικειωσή του/της σε θέματα που αφορούν θέματα συμπεριφοράς στο Διαδίκτυο, τόσο κατά την επικοινωνία όσο και κατά τον διαμοιρασμό πληροφοριών. Οι πρακτικά ασκούμενοι/-ες, έχοντας αντιληφθεί τους κινδύνους και τα προβλήματα που μπορούν να δημιουργηθούν, θα χρησιμοποιήσουν τα κατάλληλα εργαλεία ελέγχου ασφάλειας και παρακολούθησης του ιστοτόπου για ζητήματα πρόσβασης και απόδοσης. Μέσα από τα εργαλεία που διαθέτει η επιχείρηση ή ο οργανισμός, αλλά και άλλα εργαλεία που θα προτείνει ο/η πρακτικά ασκούμενος/-η, μπορεί να γίνεται έλεγχος και παραγωγή στατιστικών σχετικά με τη χρήση των ιστοσελίδων, την αναφορά των λαθών, την αναγνώριση προβλημάτων ασφάλειας κ.λπ.

Μια ενδεικτική εργασία, κατά τη διάρκεια της πρακτικής άσκησης του/της πρακτικά ασκουμένου/-ης, είναι η αναγνώριση των βασικών εννοιών της τεχνολογίας λογισμικού (λογισμικό, εργαλεία, μεθοδολογίες κ.λπ.), και η περιγραφή των σταδίων του κύκλου ανάπτυξης και ζωής μιας εφαρμογής λογισμικού. Με χρήση των αρχών της δομημένης ανάλυσης, καταγράφονται οι απαιτήσεις (λειτουργικές και μη) και οι περιορισμοί μιας εφαρμογής λογισμικού και συγκεντρώνονται στο «Έγγραφο Προδιαγραφών των Απαιτήσεων».

Μια επιπλέον ενδεικτική εργασία, κατά τη διάρκεια της πρακτικής άσκησης του/της πρακτικά ασκουμένου/-ης, είναι η επιλογή της κατάλληλης προσέγγισης (μοντέλου) για την ανάπτυξη μιας εφαρμογής λογισμικού. Με χρήση κατάλληλων μεθοδολογιών ανάλυσης και σχεδίασης, αλλά και εργαλείων λογισμικού, θα γίνει αναπαράσταση των απαιτήσεων των πελατών/-ισμών σε γλώσσα μοντελοποίησης UML και σχεδιασμός των διαγραμμάτων που περιγράφουν τη φυσική/λογική δομή, αλλά και τη δυναμική συμπεριφορά του συστήματος.

Επίσης, μια ενδεικτική εργασία είναι η εκτέλεση από τον/την πρακτικά ασκούμενο/-η των απαραίτητων ελέγχων λογισμικού, προκειμένου να βεβαιωθεί ότι το λογισμικό είναι υψηλής ποιότητας. Με τη χρήση κατάλληλων τεχνικών (επαλήθευσης και επικύρωσης λογισμικού), μεθόδων (μετρικές λογισμικού και δένδρα απόφασης) και εργαλείων testing, η εν λόγω εργασία σκοπεύει στην εξοικείωση του/της πρακτικά ασκουμένου/-ης με τα διαφορετικά επίπεδα ελέγχου λογισμικού, αναλύοντας το είδος των δοκιμών που περιλαμβάνονται σε κάθε ένα από αυτά.

Τέλος, μια ακόμη ενδεικτική εργασία ενός/μιας πρακτικά ασκουμένου/-ης είναι η εξοικείωση με τη διαδικασία σχεδίασης μιας βάσης δεδομένων, χρησιμοποιώντας αποτελεσματικά τις τεχνικές του Σημασιολογικού, Λογικού και Φυσικού Σχεδιασμού. Με τη χρήση κατάλληλου εργαλείου λογισμικού, ο/η πρακτικά ασκούμενος/-η θα απεικονίσει το σχήμα της βάσης δεδομένων μιας επιχείρησης ή ενός οργανισμού, προσαρμόζοντας τη δομή της ανάλογα με τις δοσμένες προδιαγραφές. Τέλος, θα πρέπει να δοθεί ιδιαίτερη σημασία στην ενσωμάτωση των βάσεων δεδομένων σε εφαρμογές για την άντληση στοιχείων και τον μετασχηματισμό/εμπλουτισμό των πληροφοριών, για να παραχθούν νέα δεδομένα και να δημιουργηθεί ένας νέος κύκλος πληροφοριών.

5.2 • Μεθοδολογία υλοποίησης πρακτικής άσκησης

Για την επιτυχή υλοποίηση της πρακτικής άσκησης, απαιτείται συνέργεια των συστημάτων εκπαίδευσης, κατάρτισης και αγοράς εργασίας, και συνεργασία του συνόλου του ανθρώπινου δυναμικού που εμπλέκεται στη μαθησιακή διαδικασία και περιλαμβάνει: τον/την εργοδότη/-τρια, που προσφέρει θέση πρακτικής άσκησης, τον/την εκπαιδευτή/-τρια στον χώρο εργασίας, τους/τις εργαζομένους/-ες στην επιχείρηση, τον/την εκπαιδευτή/-τρια και τον/την επόπτη/-τρια από την πλευρά της εκπαιδευτικής δομής. Η εν λόγω συνεργασία αποτελεί απαραίτητη προϋπόθεση για την αναβάθμιση της ποιότητας του τρόπου εφαρμογής της πρακτικής άσκησης και για την πραγμάτωση των μαθησιακών αποτελεσμάτων της ειδικότητας.

Στο πλαίσιο της μάθησης στον χώρο εργασίας, ο/η ασκούμενος/-η έρχεται σε επαφή με πραγματικά εργασιακά περιβάλλοντα, ασκείται σε πραγματικές εργασιακές συνθήκες και συνεργάζεται με επαγγελματίες της ειδικότητας ή άλλων ειδικοτήτων που εργάζονται στην ίδια επιχείρηση. Για τον λόγο αυτό, η ύπαρξη κουλτούρας μάθησης αποτελεί προαπαιτούμενο για όλα τα εμπλεκόμενα μέρη.

Αναλυτικότερα, κομβικός είναι ο ρόλος του/της εκπαιδευτή/-τριας της επιχείρησης, ο/η οποίος/-α καλείται να λειτουργήσει ως μέντορας, να υποστηρίζει τον/την εκπαιδευόμενο/-η στη διερεύνηση των επαγγελματικών του/της στόχων, να ενθαρρύνει την ανάπτυξη δεξιοτήτων, να εμπλέκει σε νέες εμπειρίες και να εισάγει στον κόσμο της πραγματικής εργασίας, αποτελώντας επαγγελματικό πρότυπο. Επομένως, καλείται να δημιουργήσει ένα περιβάλλον εμπιστοσύνης και ασφάλειας, που παρέχει στον/στην εκπαιδευόμενο/-η τη δυνατότητα δοκιμής και λάθους μέσω της ενθάρρυνσης της ατομικής πρωτοβουλίας. Απαραίτητα μεθοδολογικά βήματα από την πλευρά του/της εκπαιδευτή/-τριας της επιχείρησης για την ομαλή ένταξη των εκπαιδευομένων στο εργασιακό περιβάλλον και την προώθηση της συνεργασίας με το υφιστάμενο προσωπικό της επιχείρησης είναι η περιγραφή της θέσης πρακτικής άσκησης, ο προσανατολισμός στο έργο και η εναλλαγή εργασιακών θέσεων.

Ειδικότερα, ως προς την περιγραφή της θέσης πρακτικής άσκησης, ο/η εκπαιδευτής/-τρια καλείται να λάβει υπόψη τη συνοπτική περιγραφή της ειδικότητας και τις αρμοδιότητες/καθήκοντα, όπως αποτυπώνονται στον οδηγό κατάρτισης, γεγονός που προϋποθέτει τη συνεργασία με την εκπαιδευτική δομή. Καλείται να τροφοδοτήσει τους/τις ασκούμενους/-ες με τις απαραίτητες πληροφορίες που αφορούν την εκτελούμενη εργασία, τις ευθύνες, τις δεξιότητες και την εκπαίδευση που απαιτείται, τις συνθήκες κάτω από τις οποίες γίνεται η εργασία και τις σχέσεις της με άλλες εργασίες.

Παράλληλα, απαραίτητος κρίνεται και ο προσανατολισμός στο έργο, με στόχο να κατανοήσουν οι εκπαιδευόμενοι/-ες το εργασιακό περιβάλλον στο οποίο εντάσσονται, να λάβουν υπόψη τα χαρακτηριστικά και τον τρόπο λειτουργίας της επιχείρησης, να γνωρίσουν το προσωπικό της και να αναπτύξουν το αίσθημα του ανήκειν. Στο πλαί-

σιο του προσανατολισμού, οι εκπαιδευόμενοι/-ες ενημερώνονται για την ιστορία της επιχείρησης, την πολιτική της, τα προϊόντα ή τις υπηρεσίες, την κτιριακή υποδομή, τη γενική δομή, τους κανόνες συμπεριφοράς, την πολιτική προαγωγών, τα συστήματα αξιολόγησης κ.λπ.

Τέλος, προτείνεται η εναλλαγή των εργασιακών θέσεων (job rotation), κατά την οποία ο/η εκπαιδευόμενος/-η περνάει διαδοχικά από όλες τις συναφείς ή συσχετιζόμενες θέσεις εργασίας μέχρι να ασκηθεί σε όλο τον κύκλο της εργασίας. Η συγκεκριμένη πρακτική μειώνει τη μονοτονία, την κούραση και τα σφάλματα, ενώ αυξάνει την απόδοση και την ικανοποίηση. Ωστόσο, δε συνιστάται για τις ειδικότητες που απαιτούν υψηλή ειδικευση, καθώς απαιτείται χρόνος, άρα ο/η εκπαιδευτής/-τρια πρέπει να λάβει υπόψη του το εξάμηνο υλοποίησης της πρακτικής άσκησης. Τέλος, το μικρό μέγεθος της επιχείρησης μπορεί να λειτουργήσει ανασταλτικά στην εφαρμογή της εν λόγω πρακτικής.

Βιβλιογραφία

A | Βιβλιογραφικές αναφορές σχετικές με την ειδικότητα

ΕΛΛΗΝΟΓΛΩΣΣΗ

- Αβούρης, Ν. & Σιντόρης, Χ. (2023). *Προγραμματισμός Διαδικτύου*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://dx.doi.org/10.57713/kallipos-337>
- Behrouz, A. F. & Richard, F. G. (2022) *Προγραμματισμός με C++. Αντικειμενοστρεφής Προσέγγιση*. Αθήνα: Εκδόσεις Επίκεντρο.
- Βενιέρης, Ι. Σ. (2003). *Σημειώσεις για τη γλώσσα HTML (HyperText Markup Language)* [Ηλεκτρονική έκδοση]. Αθήνα: Εθνικό Μετσόβιο Πολυτεχνείο, Εργαστήριο Ευφύων Επικοινωνιών και Δικτύων Ευρείας Ζώνης. <http://www.icbnet.ntua.gr/Mathimata/IntroInternetTech/HTML%20Notes.pdf>
- Δαβγιώτης, Ι., Λαζαρίνης, Φ., Παπαδάκης, Κ., Μόκα, Ε. & Φασνάκης, Κ. (2022). *Οδηγός κατάρτισης «Τεχνικός λογισμικού Η/Υ»*. Αθήνα: Γενική Γραμματεία Επαγγελματικής Εκπαίδευσης, κατάρτισης, Δια Βίου Μάθησης και Νεολαίας.
- Δαβγιώτης, Ι., Λαζαρίνης, Φ., Καλύβα, Ε., Παπαδάκης, Κ. & Φασνάκης, Κ. (2024). *Εκπαιδευτικό Εγχειρίδιο «Τεχνικός λογισμικού Η/Υ»*. Αθήνα: Γενική Γραμματεία Επαγγελματικής Εκπαίδευσης, Κατάρτισης, Διά βίου μάθησης και Νεολαίας.
- Deitel, M. H. & Deitel, J. P. (2015). *JAVA How To Program (Early Objects)*. Αθήνα: Εκδόσεις Κλειδάριθμος.
- Δρόσος, Δ., Βουγιούκας, Δ., Καλλίγερος, Ε., Κοκολάκης, Σ. & Σκιάνης, Χ. (2015). *Εισαγωγή στην επιστήμη των υπολογιστών & επικοινωνιών*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/4582>
- Εθνική Ακαδημία Ψηφιακών Ικανοτήτων: <https://nationaldigitalacademy.gov.gr/mathimata>

eirinika.gr | Ειδήσεις από την Ελλάδα και τον Κόσμο (χ.χ.). Αυτά είναι τα επαγγέλματα που θα έχουν ζήτηση στο μέλλον: η μεγάλη έρευνα του Παν/μίου Πειραιά.

Ανακτήθηκε στις 29 Σεπτεμβρίου 2022 από:

<https://www.eirinika.gr/article/149780/ayta-einai-ta-epaggelmata-poy-tha-ehoy-n-zitisi-sto-mellon-i-megali-ereyna-toy-panmioy>

Εισαγωγή στην Python, Mathesis (χ.χ.). Ηράκλειο: Πανεπιστημιακές Εκδόσεις Κρήτης, ΙΤΕ:

<https://mathesis.cup.gr/courses/course-v1:ComputerScience+CS1.1+20C/course/>

Halvorson, M. (2016). *Microsoft Visual Basic 2013 Βήμα-Βήμα*. Αθήνα: Εκδόσεις Κλειδάριθμος.

Hanly, J. R. & Koffman, E. B. (2021). *Αρχές και τεχνικές προγραμματισμού με τη γλώσσα C*. Αθήνα: Εκδόσεις Κριτική.

Harvey, M. & Deitel, P. (2015). *C++ προγραμματισμός*. Αθήνα: Εκδόσεις Μ. Γκιούρδα.

Hennessy, L. J. & Patterson, A. D. (2020). *Αρχιτεκτονική υπολογιστών: μια ποσοτική προσέγγιση*. Αθήνα: Εκδόσεις Κλειδάριθμος.

Hoffer, J., George, J. & Valacich, J. (2011). *Πληροφοριακά συστήματα: σύγχρονη Ανάλυση & Σχεδίαση*. Θεσσαλονίκη: Εκδόσεις Τζιόλα.

Hoffer, J., Ramesh, V. & Tori, H. (2017). *Βάσεις Δεδομένων: σύγχρονη διαχείριση*. Θεσσαλονίκη: Εκδόσεις Τζιόλα.

Κανίδης, Ε., Κωνσταντοπούλου, Μ. Δ., Λέων, Π. & Μόρμορης, Ε. (χ.χ.). *Εισαγωγή στα Λειτουργικά Συστήματα*. Ανοιχτά Μαθήματα ΕΕΛΛΑΚ:

<https://elearn.ellak.gr/course/view.php?id=195>

Καριέρα Future Generation. (2022). *Τι είναι ο Μηχανικός Λογισμικού (Software engineer) | Περιγραφή-Μισθός*. Ανακτήθηκε στις 29 Σεπτεμβρίου 2022 από:

https://futuregeneration.gr/job_role/job-role-software-engineer/

Kendall, K. & Kendall, J. (2023). *Ανάλυση και Σχεδίαση Συστημάτων*. Αθήνα: Εκδόσεις Μ. Γκιούρδα.

Κεχρής, Ε. (2021). *Σχεσιακές Βάσεις Δεδομένων*. Αθήνα: Εκδόσεις Κριτική.

- Lemay, L., Colburn, R. & Kyrnin, J. (2016). *Πλήρες Εγχειρίδιο HTML5, CSS και JavaScript*. Αθήνα: Εκδόσεις Μ. Γκιούρδα.
- Λιακέας, Γ. (2021). *Η γλώσσα JavaScript*. Αθήνα: Εκδόσεις Κλειδάριθμος.
- Μαγκούτης, Κ. & Νικολάου, Χ. (2015). *Εισαγωγή στον αντικειμενοστραφή προγραμματισμό με Python*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/1708>
- Μανής, Γ. (2015). *Εισαγωγή στον Προγραμματισμό με αρωγό τη γλώσσα Python*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/2745>
- Μαστοροκώστας, Π. (2015). *Διαδικαστικός Προγραμματισμός: η Γλώσσα C*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις, ΣΕΑΒ.
- Μπάγκος, Π. (2015). *Βιοπληροφορική*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/5028>
- Νικολαΐδης, Β. (2022). *Προγραμματισμός σε R*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://dx.doi.org/10.57713/kallipos-100>
- Norton, P. & Socha, J. (2009). *Το βιβλίο της Assembly για τα PC*. Αθήνα: Εκδόσεις Κλειδάριθμος.
- Ξαρχάκος, Κ. & Καρολίδης, Δ. (2016). *Μαθαίνετε εύκολα Microsoft Office 2016*. Αθήνα: Εκδόσεις Άβακας.
- Ξένος, Μ. (2021). *Ποιότητα Λογισμικού*. Αθήνα: Εκδόσεις Gotsis.
- Ο επίσημος ιστότοπος της γλώσσας Go: <https://go.dev/>
- Ο επίσημος ιστότοπος της γλώσσας Kotlin: <https://kotlinlang.org/>
- Ο επίσημος ιστότοπος της γλώσσας Perl: <https://www.perl.org/>
- Ο επίσημος ιστότοπος της γλώσσας R: <https://www.r-project.org/>
- Ο επίσημος ιστότοπος της γλώσσας Ruby: <https://www.ruby-lang.org/>

Ο επίσημος ιστότοπος της γλώσσας Rust: <https://www.rust-lang.org/>

Ο επίσημος ιστότοπος της γλώσσας Swift: <https://www.swift.org/>

Ο επίσημος ιστότοπος της γλώσσας Typescript: <https://www.typescriptlang.org/>

Ο επίσημος ιστότοπος της γλώσσας Xml: <https://www.xml.com/>

Ο επίσημος ιστότοπος του Android studio: <https://developer.android.com/studio>

Ο επίσημος ιστότοπος του HeidiSQL: <https://www.heidisql.com/>

Ο επίσημος ιστότοπος του MariaDB: <https://mariadb.org/documentation/>

Ο επίσημος ιστότοπος του Ubuntu: <https://ubuntu.com/>

Ομάδα του Εγχειριδίου του Ubuntu. (2010). *Ξεκινώντας με το Ubuntu 10.04*
[Ηλεκτρονική έκδοση]: <https://ubuntu-manual.org/?lang=el>

Patterson, A. D. & Hennessy, L. J. (2023). *Οργάνωση και σχεδίαση υπολογιστών MIPS: η διασύνδεση υλικού και λογισμικού*. Αθήνα: Εκδόσεις Κλειδάριθμος.

Phillips, J. (2007). *Διαχείριση έργων πληροφορικής*. Αθήνα: Εκδόσεις Μ. Γκιούρδα.

Προχωρημένος προγραμματισμός με Python, Mathesis, Ηράκλειο: Πανεπιστημιακές Εκδόσεις Κρήτης, ITE:
<https://mathesis.cup.gr/courses/course-v1:ComputerScience+CS1.2+20C/course/>

Radiotileoptiki S.A (OPEN Digital Group). (2022, Ιανουάριος 30). «Πανελλήνιες 2022: Σπουδές στην πληροφορική. Πτυχίο που προσφέρει σίγουρη καριέρα», ΕΘΝΟΣ:
<https://www.ethnos.gr/greece/article/193406/panellhnies2022spoydessthnlhrofrikhptyxiopoyprofereisigoyrhkaria>

Σαμψών, Δ. (2003). *Η Γλώσσα Σήμανσης XML* [Πανεπιστημιακές σημειώσεις]:
http://www.fme.aegean.gr/sites/default/files/dsamps_on_xml_lectures-notes-dec2003.pdf

Sharp, J. (2008). *Microsoft Visual C# 2008*. Αθήνα: Εκδόσεις Κλειδάριθμος.

- Σιδηρόπουλος, Α. (2015). *Εισαγωγή στα λειτουργικά συστήματα με χρήση και προγραμματισμό του Κελύφους*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/2474>
- Silberschatz, A., Galvin, P. B. & Gagne, G. (2009). *Λειτουργικά Συστήματα*. Αθήνα: Εκδόσεις Ίων.
- Stallings, W. (2017). *Λειτουργικά Συστήματα*. Θεσσαλονίκη: Εκδόσεις Τζιόλα.
- Stroustrup, B. (2018). *Προγραμματισμός με τη C++*. Αθήνα: Εκδόσεις Παπασωτηρίου.
- Συλλογικό Έργο. (2021). *7+1 Ελληνικά Microsoft Windows 10-Office 2019/365*. Αθήνα: Εκδόσεις Κλειδάριθμος.
- Tanenbaum, A. S. & Herbert, B. (2018). *Σύγχρονα Λειτουργικά Συστήματα*. Αθήνα: Εκδόσεις Κλειδάριθμος.
- Τσιρίκας, Α. (2021). *Οργάνωση & Διοίκηση Έργων*. Αθήνα: Εκδόσεις Rosili.
- Ullman, L. (2009). *Εισαγωγή στις PHP 6 & MYSQL 5 με εικόνες*. Αθήνα: Εκδόσεις Κλειδάριθμος.
- Φιτσιλής, Π. (χ.χ.). *Σχεδιασμός και Διαχείριση Έργων*. [Ανοιχτά μαθήματα του Οικονομικού Πανεπιστημίου Αθηνών]: <https://eclass.aueb.gr/courses/INF194/>
- Χατζηγιαννάκης, Ν. Μ. (2014). *Η γλώσσα C++ σε βάθος*. Αθήνα: Εκδόσεις Κλειδάριθμος.
- Χατζηγιαννάκης, Ν. Μ. (2017). *Η γλώσσα C σε βάθος*. Αθήνα: Εκδόσεις Κλειδάριθμος.
- Χειλάς, Κ., Βακαλούδης, Α. & Πολίτης, Α. (2015). *Η γλώσσα HTML. Πίνακες, Λίστες, Φόρμες*. Αθήνα: ΚΑΛΛΙΠΟΣ. Ανοικτές Ακαδημαϊκές Εκδόσεις: <https://hdl.handle.net/11419/1774>

ΞΕΝΟΓΛΩΣΣΗ

- Bennett, S., Farmer, R. & McRobb, S. (2010). *Object-oriented systems analysis and design using UML*. UK Higher Education Computing Computer Science.

Blokdyk, G. (2019). *Software development life cycle SDLC a complete guide-2020 edition*. 5starcooks.

C tutorial. (n.d.). Retrieved from: <https://www.w3schools.com/c/index.php>

Dawson, C. & Straub, B. (2016). *Building Tools with GitHub-Customize Your Workflow*. California: O'Reilly.

Duckett, J. (2014). *JavaScript and jQuery: Interactive front-end web development*. . New Jersey: Wiley & Sons.

Duckett, J. (2023). *Front-End Back-End Development with HTML, CSS, JavaScript, jQuery, PHP, and MySQL*. New Jersey: Wiley & Sons.

Freeman, A. (2017). *Pro ASP.NET Core MVC 2*. New York: Apress.

Gajda, W. (2013). *Git Recipes-A Problem-Solution Approach*. New York: Apress.

Guleria, P. (2022). *ASP.NET and VB.NET in 30 Days: Acquire a Solid Foundation in the Fundamentals of Windows and Web Application Development*. New Delhi: BPB Publications.

Hussain, K. & Hussain, F. (2023). *Mastering VB.NET: A Comprehensive Guide to Visual Basic .NET Programming*. Singapore: Sonar Publishing.

Hutten, D. (2017). *Git-Learn Version Control with Git-A step-by-step Ultimate beginners Guide*. CreateSpace Independent Publishing Platform.

Information system-Definition, examples, & facts. (2001, October 12). Retrieved from: <https://www.britannica.com/topic/information-system>

JavaScript tutorial. (n.d.). Retrieved from: <https://www.w3schools.com/js/default.asp>

Kaner, C., Bach, J. & Pettichord, B. (2001). *Lessons learned in software testing: A context-driven approach*. New Jersey: Wiley & Sons.

Langer, A. M. (2012). *Guide to software development: Designing and managing the life cycle*. New York: Springer Science & Business Media.

Loy, M., Niemeyer, P. & Leuck, D. (2023). *Learning Java: An Introduction to Real-World Programming with JAVA*. California: O'Reilly.

Mann, A. E. (2023). *PHP Cookbook*. California: O'Reilly.

Negus, C. (2020). *Linux Bible*. New Jersey: Wiley & Sons.

Omos, U. V. (2020). *Big Little Book on Git-Gitlab Github GitOps DevOps & CICD: Git-Gitlab Github GitOps DevOps & CICD*. Gridlockaz.

Price, J. M. (2021). *C# 10 and .NET 6-Modern Cross-Platform Development: Build apps, websites, and services with ASP.NET Core 6, Blazor, and EF Core 6 using Visual Studio 2022 and Visual Studio Code*. Birmingham: Packt Publishing.

Rajaraman, V. (2004). *Analysis and design of information systems*. PHI Learning Pvt.

Robbins, J. N. (2018). *Learning web design: A beginner's guide to HTML, CSS, JavaScript, and web graphics*. California: O'Reilly.

Sarabia, R. & Hansen, B. (2023). *Test-Driven Development with PHP 8: Build extensible, reliable, and maintainable enterprise-level applications using TDD and BDD with PHP*. Birmingham: Packt Publishing.

Schildt, H. & Coward, D. (2024). *Java: The Complete Reference*. Ohio: McGraw Hill.

Sisk, K. (2022, April 25). *A roadmap from mechanic to software engineer*. Medium: <https://engineering.udacity.com/a-roadmap-from-mechanic-to-software-engineer-13c78759073>

The Enterprisers Project | A community helping CIOs and IT leaders solve problems: (n.d.). *Software development trends: What's flourishing and what's fading*. <https://enterprisersproject.com/article/2022/8/software-development-trends>

Online Digital Skills Training Programs & Courses for Professionals-Digital Skills Global: (n.d.). *The top 10 digital skills tech companies are looking for today*. <https://digitalskillsglobal.com/blog/the-top-10-digital-skills-tech-companies-are-looking-for-today>

Troelsen, A. & Japikse, P. (2022). *Pro C# 10 with .NET 6: Foundational Principles and Practices in Programming*. New York: Apress.

Ullenboom, C. (2024). *Spring Boot 3 and Spring Framework 6*. Bonn: Rheinwerk Publishing.

Yamacli, S. (2019). *Beginner's Guide to Visual Basic .NET Programming: A Practical Approach to VB.NET*. Independently published.

B | Βιβλιογραφικές αναφορές σχετικές με τη μεθοδολογία ανάπτυξης των εκπαιδευτικών εγχειριδίων

Βαρβιτσιώτη, Ρ., Γούλας, Χρ., Θεοδωρή, Ελ. Καρατράσογλου, Ιάκ., Μαρκίδης, Κ., Μπαμπανέλου, Δ. & Νάτσης, Π. (2022). *Πρότυπος οδηγός κατάρτισης*. Αθήνα: ΚΑΝΕΠ/ΓΣΕΕ.

Βαρβιτσιώτη, Ρ., Θεοδωρή, Ελ., Μαρκίδης, Κ. & Μπαμπανέλου, Δ. (2023). *Ανάπτυξη οριζόντιας εκπαιδευτικής μεθοδολογίας/Εκπαιδευτικός σχεδιασμός στο πεδίο της αρχικής επαγγελματικής κατάρτισης*. Αθήνα: ΚΑΝΕΠ/ΓΣΕΕ.

Βαρβιτσιώτη, Ρ., Θεοδωρή, Ελ., Μαρκίδης, Κ. & Μπαμπανέλου, Δ. (2023). *Πρότυπο εκπαιδευτικό εγχειρίδιο*. Αθήνα: ΚΑΝΕΠ/ΓΣΕΕ.

Βαρβιτσιώτη, Ρ., Θεοδωρή, Ελ., Μαρκίδης, Κ. & Μπαμπανέλου, Δ. (2023). *Πρότυπη τράπεζα θεμάτων*. Αθήνα: ΚΑΝΕΠ/ΓΣΕΕ.

Γενική Γραμματεία Επαγγελματικής Εκπαίδευσης Κατάρτισης και Διά Βίου Μάθησης. Τμήμα Σπουδών Προγραμμάτων και Οργάνωσης Επαγγελματικής Κατάρτισης. (2020). *Οδηγοί Σπουδών ειδικοτήτων ΙΕΚ του Ν. 4186/2013*. Ανακτήθηκε στις 5 Μαρτίου 2020 από: <https://gsveltly.minedu.gov.gr/sxoles-anoteris-epaggelmatikis-katartisis-s-a-e-k/odigoi-katartisis-spoudon>

Γούλας, Χ. & Λιντζέρης, Π. (2017). *Διά Βίου Μάθηση, Επαγγελματική Κατάρτιση, Απασχόληση και Οικονομία: Νέα Δεδομένα, Προτεραιότητες και Προκλήσεις*. Αθήνα: ΙΜΕ/ΓΣΕΒΕΕ, ΙΝΕ/ΓΣΕΕ.

Γούλας, Χ., Μαρκίδης, Κ. & Μπαμπανέλου, Δ. (2021). *Πρότυπο ανάπτυξης εκπαιδευτικών υλικών του ΙΝΕ/ΓΣΕΕ* [Εσωτερική έκδοση]. Αθήνα: Ινστιτούτο Εργασίας της Γενικής Συνομοσπονδίας Εργατών Ελλάδας.

Δημουλάς, Κ., Βαρβιτσιώτη, Ρ. & Σπηλιώτη, Χ. (2007). *Οδηγός Ανάπτυξης Επαγγελματικών Περιγραμμάτων*. Αθήνα: ΓΣΕΕ, ΣΕΒ, ΓΣΕΒΕΕ, ΕΣΣΕ.

ΙΝΕ/ΓΣΕΕ. (2021). *Νέα βίντεο παρουσίασης επαγγελμαμάτων* [Video]. Ανακτήθηκε από: <https://oasi.inegsee.gr/>

Καραλής, Θ., Καρατράσογλου, Ι., Μαρκίδης, Κ., Βαρβιτσιώτη, Ρ., Νάτσης, Π. & Παπαευσταθίου, Κ. (2021). *Μεθοδολογικές προσεγγίσεις ανάπτυξης επαγγελματικών περιγραμμάτων και πλαισίων εκπαιδευτικών προδιαγραφών προγραμμάτων*. Αθήνα: ΙΝΕ/ΓΣΕΕ. Ανακτήθηκε στις 15 Μαρτίου 2022 από: https://www.inegsee.gr/wp-content/uploads/2021/07/Me8odologia_EP_Ebook.pdf

Λευθεριώτου, Π. (χ.χ.). *Η Εκπαιδευτική Διεργασία στην Εκπαίδευση Ενηλίκων*. Αθήνα: Υπουργείο Εθνικής Παιδείας & Θρησκευμάτων, Γενική Γραμματεία Εκπαίδευσης Ενηλίκων.

Cedefop. (2014). *Επαγγελματική Εκπαίδευση και Κατάρτιση στην Ελλάδα: Συνοπτική Περιγραφή*. Λουξεμβούργο: Υπηρεσία Εκδόσεων της Ευρωπαϊκής Ένωσης.

Cedefop. (2014). *Terminology of European Education and Training Policy: A Selection of 130 Key Terms* (2nd edition). Luxembourg: Publications Office of the European Union.

Cedefop. (2018). *Skills forecast, trends and challenges to 2030*. Luxembourg: Publications Office of the European Union. Ανακτήθηκε από: <https://doi.org/10.2801/4492>

Kopnov, V. A., Shmurygina, O., Shchipanova, D., Dremina, M. A., Papaloizou, L., Orphanidou, Y. & Morevs, P. (2018). "Functional Analysis and Functional Maps of Qualifications in ECVET Context", in: *The Education and Science Journal*, 20(6), pp. 90-117. Διαθέσιμο σε: https://www.researchgate.net/publication/326242379_Functional_Analysis_and_Functional_Maps_of_Qualifications_in_ECVET_Context

Mansfield, B. & Schmidt, H. (2001). *Linking Vocational Education and Training Standards and Employment Requirements: An International Manual*. European Training Foundation. Ανακτήθηκε στις 9 Ιουνίου 2020 από: https://www.etf.europa.eu/sites/default/files/m/C12578310056925BC12571FE00473D6B_NOTE6UAEET.pdf

Μπουραντάς, Δ. Κ. (2002). *Μάνατζμεντ, Θεωρητικό υπόβαθρο, Σύγχρονες Πρακτικές*. Αθήνα: Εκδόσεις Μπένου.

Psifidou, I. (2009). "What learning outcome based curricula imply for teachers and trainers", in: *7th International Conference "Comparative Education and Teacher Training"* (June 29-July 3, 2009), vol. 7, pp. 183-188. Sofia, Bulgaria: Bureau for Educational Services.

Γ | Σχετική εθνική νομοθεσία

Επίσημη Εφημερίδα της Ευρωπαϊκής Ένωσης, Σύσταση του Ευρωπαϊκού Συμβουλίου και του Ευρωπαϊκού Κοινοβουλίου της 18ης Δεκεμβρίου 2006 σχετικά με τις βασικές ικανότητες της διά βίου μάθησης (2006/962/ΕΚ). Ανακτήθηκε στις 15 Μαρτίου 2022, από: <https://eur-lex.europa.eu/eli/reco/2006/962/oj>

Επίσημη Εφημερίδα της Ευρωπαϊκής Ένωσης, Σύσταση του Ευρωπαϊκού Συμβουλίου και του Ευρωπαϊκού Κοινοβουλίου της 22ας Μαΐου 2018 σχετικά με τις βασικές ικανότητες της διά βίου μάθησης (2018/C189/01). Ανακτήθηκε στις 15 Μαρτίου 2022, από: [https://eur-lex.europa.eu/legal-content/EL/TXT/PDF/?uri=CELEX:32018H0604\(01\)&from=PL](https://eur-lex.europa.eu/legal-content/EL/TXT/PDF/?uri=CELEX:32018H0604(01)&from=PL)

ΦΕΚ 4146/Β/9-9-2021. Κοινή Υπουργική Απόφαση υπ' αριθμ. ΦΒ7/108652/Κ3. *Πλαίσιο Ποιότητας Μαθητείας.*

ΦΕΚ 3938/Β/26-8-2021. Κοινή Υπουργική Απόφαση υπ' αριθμ. Κ5/97484. *Πρακτική άσκηση σπουδαστών Ινστιτούτων Επαγγελματικής Κατάρτισης αρμοδιότητας Υπουργείου Παιδείας και Θρησκευμάτων.*

ΦΕΚ 254/Α/21-12-2020. Νόμος υπ' αριθμ. 4763/2020. *Εθνικό Σύστημα Επαγγελματικής Εκπαίδευσης, Κατάρτισης και Διά Βίου Μάθησης, ενσωμάτωση στην ελληνική νομοθεσία της Οδηγίας (ΕΕ) 2018/958 του Ευρωπαϊκού Κοινοβουλίου και του Συμβουλίου της 28ης Ιουνίου 2018 σχετικά με τον έλεγχο αναλογικότητας πριν από τη θέσπιση νέας νομοθετικής κατοχύρωσης των επαγγελματιών (ΕΕ L 173), κύρωση της Συμφωνίας μεταξύ της Κυβέρνησης της Ελληνικής Δημοκρατίας και της Κυβέρνησης της Ομοσπονδιακής Δημοκρατίας της Γερμανίας για το Ελληνογερμανικό Ίδρυμα Νεολαίας και άλλες διατάξεις.*

ΦΕΚ 3520/Β/19-9-2019. Υπουργική Απόφαση υπ' αριθμ. 40331/Δ1.13521/2019. *Επανακαθορισμός Όρων Ηλεκτρονικής Υποβολής Εντύπων Αρμοδιότητας Σώματος Επιθεώρησης Εργασίας (ΣΕΠΕ) και Οργανισμού Απασχολήσεως Εργατικού Δυναμικού (ΟΑΕΔ).*

ΦΕΚ 2440/Β/18-7-2017. Κοινή Υπουργική Απόφαση υπ' αριθμ. Κ1/118932/2017. *Ρύθμιση Θεμάτων Επιδότησης και Ασφάλισης της Μαθητείας των Σπουδαστών των Δημόσιων και Ιδιωτικών Ινστιτούτων Επαγγελματικής Κατάρτισης (ΙΕΚ) και Σχολών Επαγγελματικής Κατάρτισης (ΣΕΚ).*

ΦΕΚ 1245/Β/11-04-2017. Υπουργική Απόφαση υπ' αριθμ. Κ1/54877/2017.

Τροποποίηση του Κανονισμού Λειτουργίας Ινστιτούτων Επαγγελματικής Κατάρτισης (ΙΕΚ) που υπάγονται στη Γενική Γραμματεία Διά Βίου Μάθησης (ΓΓΔΒΜ).

ΦΕΚ 1807/Β/2-7-2014. Υπουργική Απόφαση υπ' αριθμ. 5954/2014. Κανονισμός

Λειτουργίας Ινστιτούτων Επαγγελματικής Κατάρτισης (ΙΕΚ) που υπάγονται στη Γενική Γραμματεία Διά Βίου Μάθησης (ΓΓΔΒΜ).

ΦΕΚ 566/Β/8-5-2006. Κοινή Υπουργική Απόφαση υπ' αριθμ. 110998/8-5-2006.

Πιστοποίηση Επαγγελματικών Περιγραμμάτων.

ΜΕΘΟΔΟΛΟΓΙΑ ΑΝΑΠΤΥΞΗΣ ΕΚΠΑΙΔΕΥΤΙΚΩΝ ΕΓΧΕΙΡΙΔΙΩΝ

Συντακτική ομάδα

Ρένα Βαρβιτσιώτη

Ελένη Θεοδωρή

Κωνσταντίνος Μαρκίδης

Δέσποινα Μπαμπανέλου

Επιμέλεια σύνταξης

Δέσποινα Μπαμπανέλου

Το εκπαιδευτικό εγχειρίδιο συντάχθηκε στο πλαίσιο της Πράξης με τίτλο «Διαμόρφωση οδηγών κατάρτισης και εκπαιδευτικών εγχειριδίων Ινστιτούτων Επαγγελματικής Κατάρτισης (IEK)» με κωδικό ΟΠΣ (MIS) 5069281 μέσω του Επιχειρησιακού Προγράμματος «Ανάπτυξη Ανθρώπινου Δυναμικού, Εκπαίδευση και Διά Βίου Μάθηση 2014-2020» που συγχρηματοδοτείται από την Ελλάδα και την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο-Ε.Κ.Τ.).



**Επιχειρησιακό Πρόγραμμα
Ανάπτυξη Ανθρώπινου Δυναμικού,
Εκπαίδευση και Διά Βίου Μάθηση**

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης





Επιχειρησιακό Πρόγραμμα
Ανάπτυξη Ανθρώπινου Δυναμικού,
Εκπαίδευση και Διά Βίου Μάθηση

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

