

ΠΡΟΗΓΜΕΝΑ ΘΕΜΑΤΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

ΠΕΡΙΓΡΑΜΜΑ ΜΑΘΗΜΑΤΟΣ

1. ΓΕΝΙΚΑ

ΣΧΟΛΗ	ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ		
ΤΜΗΜΑ	ΠΛΗΡΟΦΟΡΙΚΗΣ		
ΕΠΙΠΕΔΟ ΣΠΟΥΔΩΝ	ΠΡΟΠΤΥΧΙΑΚΟ		
ΚΩΔΙΚΟΣ ΜΑΘΗΜΑΤΟΣ	403ΕΥΥΚ	ΕΞΑΜΗΝΟ ΣΠΟΥΔΩΝ	4 ^ο
ΤΙΤΛΟΣ ΜΑΘΗΜΑΤΟΣ	ΠΡΟΗΓΜΕΝΑ ΘΕΜΑΤΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ		
ΑΥΤΟΤΕΛΕΙΣ ΔΙΔΑΚΤΙΚΕΣ ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ <i>σε περίπτωση που οι πιστωτικές μονάδες απονέμονται σε διακριτά μέρη του μαθήματος π.χ. Διαλέξεις, Εργαστηριακές Ασκήσεις κ.λπ. Αν οι πιστωτικές μονάδες απονέμονται ενιαία για το σύνολο του μαθήματος αναγράψτε τις εβδομαδιαίες ώρες διδασκαλίας και το σύνολο των πιστωτικών μονάδων</i>		ΕΒΔΟΜΑΔΙΑΙΕΣ ΩΡΕΣ ΔΙΔΑΣΚΑΛΙΑΣ	ΠΙΣΤΩΤΙΚΕΣ ΜΟΝΑΔΕΣ
Διαλέξεις		2	6
Φροντιστηριακές Ασκήσεις		2	
<i>Προσθέστε σειρές αν χρειαστεί. Η οργάνωση διδασκαλίας και οι διδακτικές μέθοδοι που χρησιμοποιούνται περιγράφονται αναλυτικά στο 4.</i>			
ΤΥΠΟΣ ΜΑΘΗΜΑΤΟΣ <i>Υποβάθρου, Γενικών Γνώσεων, Επιστημονικής Περιοχής, Ανάπτυξης Δεξιοτήτων</i>	Υποβάθρου		
ΠΡΟΑΠΑΙΤΟΥΜΕΝΑ ΜΑΘΗΜΑΤΑ:	-		
ΓΛΩΣΣΑ ΔΙΔΑΣΚΑΛΙΑΣ και ΕΞΕΤΑΣΕΩΝ:	Ελληνική		
ΤΟ ΜΑΘΗΜΑ ΠΡΟΣΦΕΡΕΤΑΙ ΣΕ ΦΟΙΤΗΤΕΣ ERASMUS	-		
ΗΛΕΚΤΡΟΝΙΚΗ ΣΕΛΙΔΑ ΜΑΘΗΜΑΤΟΣ (URL)	-		

2. ΜΑΘΗΣΙΑΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ

Μαθησιακά Αποτελέσματα
<i>Περιγράφονται τα μαθησιακά αποτελέσματα του μαθήματος οι συγκεκριμένες γνώσεις, δεξιότητες και ικανότητες καταλλήλου επιπέδου που θα αποκτήσουν οι φοιτητές μετά την επιτυχή ολοκλήρωση του μαθήματος.</i>
<i>Συμβουλευτείτε το Παράρτημα Α</i>
<ul style="list-style-type: none"> • Περιγραφή του Επιπέδου των Μαθησιακών Αποτελεσμάτων για κάθε ένα κύκλο σπουδών σύμφωνα με Πλαίσιο Προσόντων του Ευρωπαϊκού Χώρου Ανώτατης Εκπαίδευσης • Περιγραφικοί Δείκτες Επιπέδων 6, 7 & 8 του Ευρωπαϊκού Πλαισίου Προσόντων Διά Βίου Μάθησης
<i>και Παράρτημα Β</i>
<ul style="list-style-type: none"> • Περιληπτικός Οδηγός συγγραφής Μαθησιακών Αποτελεσμάτων
<p>Σκοπός του μαθήματος είναι η περαιτέρω ανάπτυξη και επέκταση των ικανοτήτων και γνώσεων προγραμματισμού. Θα αναπτυχθούν προηγμένα θέματα σχετικά με την σχεδίαση και υλοποίησης γλωσσών προγραμματισμού, όπως συστήματα τύπων (type systems), πέρασμα παραμέτρων (parameter passing), coroutines, νήματα (threads), ταυτοχρονισμός (concurrency), παράλληλισμός (parallelism), δέσμευση μεθόδων (method binding), συναρτησιακός προγραμματισμός (functional programming).</p> <p>Ο φοιτητής μελετώντας θέματα συσχετιζόμενα με την τεχνολογία των γλωσσών προγραμματισμού βελτιώνει τις προγραμματιστικές του ικανότητες με την βαθύτερη κατανόηση του τρόπου δημιουργίας και εκτέλεσης εκτελέσιμου κώδικα. Ο παράλληλος προγραμματισμός που υποστηρίζεται από τις σύγχρονες γλώσσες είναι ουσιαστικότερο στοιχείο για την εκμετάλλευση των δυνατοτήτων των σημερινών πολυπύρηνων επεξεργαστών. Επίσης, ο συναρτησιακός προγραμματισμός επανήλθε στο προσκήνιο, γιατί είναι ιδιαίτερα κατάλληλος για τις σημερινές απαιτητικές εφαρμογές που αφορούν την αποτελεσματική επεξεργασία τεράστιου όγκου</p>

δεδομένων (π.χ. big data, next generation sequencing data στην Βιοπληροφορική). Στο μάθημα θα καλλιεργηθούν οι γνωσιακές βάσεις για να μπορεί ο σπουδαστής να ασχοληθεί αποτελεσματικά με αυτά τα σύγχρονα πεδία εφαρμογών.

Γενικές Ικανότητες

Λαμβάνοντας υπόψη τις γενικές ικανότητες που πρέπει να έχει αποκτήσει ο πτυχιούχος (όπως αυτές αναγράφονται στο Παράρτημα Διπλώματος και παρατίθενται ακολούθως) σε ποια / ποιες από αυτές αποσκοπεί το μάθημα:

Αναζήτηση, ανάλυση και σύνθεση δεδομένων και πληροφοριών, με τη χρήση και των απαραίτητων τεχνολογιών

Προσαρμογή σε νέες καταστάσεις

Λήψη αποφάσεων

Αυτόνομη εργασία

Ομαδική εργασία

Εργασία σε διεθνές περιβάλλον

Εργασία σε διεπιστημονικό περιβάλλον

Παράγωγή νέων ερευνητικών ιδεών

Σχεδιασμός και διαχείριση έργων

Σεβασμός στη διαφορετικότητα και στην πολυπολιτισμικότητα

Σεβασμός στο φυσικό περιβάλλον

Επίδειξη κοινωνικής, επαγγελματικής και ηθικής υπευθυνότητας

και ευαισθησίας σε θέματα φύλου

Άσκηση κριτικής και αυτοκριτικής

Προαγωγή της ελεύθερης, δημιουργικής και επαγωγικής σκέψης

- Αυτόνομη Εργασία
- Ομαδική Εργασία

3. ΠΕΡΙΕΧΟΜΕΝΟ ΜΑΘΗΜΑΤΟΣ

Το μάθημα έχει ως στόχο την ανάπτυξη του τρόπου με τον οποίο γίνεται η υλοποίηση σύγχρονων προγραμματιστικών περιβαλλόντων (programming environments). Η μελέτη διαφορετικών προσεγγίσεων στην κατασκευή γλωσσών προγραμματισμού και των σχετικών αλγορίθμων και εννοιών που υπεισέρχονται, βελτιώνει την ικανότητα αποτελεσματικής επίλυσης πλήθους προβλημάτων από διαφορετικές εφαρμογές. Στο μάθημα η παρουσίαση των εννοιών θα γίνεται με ακαδημαϊκό τρόπο, δηλαδή επικεντρώνοντας στις έννοιες, ανεξάρτητα γλώσσας προγραμματισμού. Παραδείγματα θα χρησιμοποιούνται από συγκεκριμένες σύγχρονες γλώσσες προγραμματισμού όπως η Java, C++, Scala, Julia, Go. Ένα περίγραμμα της ύλης στην οποία θα επικεντρώνεται το μάθημα ακολουθεί:

1. Θεμελιώδη θέματα στην σχεδίαση γλωσσών

Θα είναι μια γλώσσα statically typed ή dynamically typed, ή θα επιτρέπει μια ανάμεικτη υλοποίηση όπου για παράδειγμα με χρησιμοποίηση οδηγιών στον μεταγλωττιστή ορισμένα τμήματα κώδικα θα αντιμετωπίζονται ως statically typed σε ένα δυναμικό περιβάλλον εκτέλεσης; Δεν υπάρχει γενική απάντηση στο ερώτημα αυτό και οι επιλογές που πρέπει να κάνει ένας σχεδιαστής /προγραμματιστής πρέπει να είναι πάντα προσαρμοσμένες στις ιδιαιτερότητες της εφαρμογής του.

Άλλο σημαντικό θέμα σε παρόμοιο πλαίσιο, είναι η αποτελεσματική σχεδίαση εφαρμογών βάση της αντικειμενοστραφούς (object oriented) προσέγγισης. Οι σύγχρονες γλώσσες προγραμματισμού παρέχουν και την δυνατότητα συναρτησιακού προγραμματισμού (functional programming). Θα γίνει εισαγωγή στις έννοιες του συναρτησιακού προγραμματισμού και των περιπτώσεων στις οποίες είναι αποτελεσματικότερος από τον αντικειμενοστραφή.

2. Συστήματα τύπων (Type Systems)

Θα αναπτυχθεί το τι είναι τα συστήματα τύπων και ποια είναι η χρησιμότητά τους. Επίσης οι αλγόριθμοι με τους οποίους ο μεταγλωττιστής πραγματοποιεί έλεγχο τύπων αν έχουμε στατικό περιβάλλον εκτέλεσης. Στις δυναμικές γλώσσες προγραμματισμού ο έλεγχος τύπων γίνεται από το περιβάλλον υποστήριξης εκτέλεσης (run time support system).

Στις αντικειμενοστραφείς ιδίως γλώσσες ιδιαίτερα σημαντικές είναι οι έννοιες του πολυμορφισμού (polymorphism) και της επιλογής μεθόδου (method dispatching).

3. Αφαίρεση δεδομένων και Αντικειμενοστρέφεια (Data Abstraction and Object Orientation)

Θα αναπτυχθούν προηγμένα θέματα αντικειμενοστραφούς προγραμματισμού, όπως η επιλογή κατασκευαστή (constructor), η συλλογή σκουπιδιών (garbage collection), οι εικονικές και μη εικονικές μέθοδοι, οι αφηρημένες κλάσεις, η αναζήτηση μελών, ο πολυμορφισμός, τα κλεισίματα (closures) και η πολλαπλή κληρονομικότητα (multiple inheritance).

4. Συναρτησιακός προγραμματισμός (Functional Programming)

Ο συναρτησιακός προγραμματισμός αντιμετωπίζει την επίλυση προβλημάτων ως μια σειρά ανάπτυξης εκφράσεων (συναρτήσεων). Για ορισμένα προβλήματα, όπως για παράδειγμα αυτά που

συσχετίζονται με την επεξεργασία μεγάλου όγκου δεδομένων (“big data applications”) είναι ιδιαίτερα πρόσφορος και αποτελεσματικός. Επίσης ο συναρτησιακός προγραμματισμός διευκολύνει σημαντικά την υλοποίηση πραγματικά παράλληλης επεξεργασίας στους σημερινούς πολυπύρηνους (multicore) επεξεργαστές. Οι σύγχρονες γλώσσες προγραμματισμού όπως η Java, C++ και η Scala παρέχουν μοντέλα αποτελεσματικού συνδυασμού του αντικειμενοστραφούς με τον συναρτησιακό προγραμματισμό.

5. Ταυτοχρονισμός (concurrency)

Θα αναπτυχθεί η έννοια και η υλοποίηση των νημάτων (threads), οι χαμηλού επιπέδου μέθοδοι συγχρονισμού καθώς και η υλοποίηση συγχρονισμού με υψηλότερου επιπέδου κατασκευές όπως οι σηματοφόροι (semaphores) και τα μόνιτορς. Επίσης θα αναπτυχθούν μοντέλα που βασίζονται σε ανταλλαγή μηνυμάτων (π.χ. actors) και μοντέλα ασύγχρονου υπολογισμού όπως τα futures και promises.

4. ΔΙΔΑΚΤΙΚΕΣ και ΜΑΘΗΣΙΑΚΕΣ ΜΕΘΟΔΟΙ - ΑΞΙΟΛΟΓΗΣΗ

<p>ΤΡΟΠΟΣ ΠΑΡΑΔΟΣΗΣ Πρόσωπο με πρόσωπο, Εξ αποστάσεως εκπαίδευση κ.λπ.</p>	Πρόσωπο με πρόσωπο													
<p>ΧΡΗΣΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΑΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ Χρήση Τ.Π.Ε. στη Διδασκαλία, στην Εργαστηριακή Εκπαίδευση, στην Επικοινωνία με τους φοιτητές</p>	Εξειδικευμένο open-source Λογισμικό σε περιβάλλον Linux													
<p>ΟΡΓΑΝΩΣΗ ΔΙΔΑΣΚΑΛΙΑΣ Περιγράφονται αναλυτικά ο τρόπος και μέθοδοι διδασκαλίας. Διαλέξεις, Σεμινάρια, Εργαστηριακή Άσκηση, Άσκηση Πεδίου, Μελέτη & ανάλυση βιβλιογραφίας, Φροντιστήριο, Πρακτική (Τοποθέτηση), Κλινική Άσκηση, Καλλιτεχνικό Εργαστήριο, Διαδραστική διδασκαλία, Εκπαιδευτικές επισκέψεις, Εκπόνηση μελέτης (project), Συγγραφή εργασίας / εργασιών, Καλλιτεχνική δημιουργία, κ.λπ. Αναγράφονται οι ώρες μελέτης του φοιτητή για κάθε μαθησιακή δραστηριότητα καθώς και οι ώρες μη καθοδηγούμενης μελέτης ώστε ο συνολικός φόρτος εργασίας σε επίπεδο εξαμήνου να αντιστοιχεί στα standards του ECTS</p>	<table border="1"> <thead> <tr> <th data-bbox="676 904 1010 965">Δραστηριότητα</th> <th data-bbox="1010 904 1348 965">Φόρτος Εργασίας Εξαμήνου</th> </tr> </thead> <tbody> <tr> <td data-bbox="676 965 1010 999">Διαλέξεις</td> <td data-bbox="1010 965 1348 999">26 x 2 = 52 ώρες</td> </tr> <tr> <td data-bbox="676 999 1010 1099">Φροντιστηριακές Ασκήσεις που εστιάζουν στην εφαρμογή</td> <td data-bbox="1010 999 1348 1099">26 x 2 = 52 ώρες</td> </tr> <tr> <td data-bbox="676 1099 1010 1133">Αυτοτελής Μελέτη</td> <td data-bbox="1010 1099 1348 1133">44 ώρες</td> </tr> <tr> <td data-bbox="676 1133 1010 1167">Γραπτές Εξετάσεις</td> <td data-bbox="1010 1133 1348 1167">2 x 1 = 2 ώρες</td> </tr> <tr> <td data-bbox="676 1167 1010 1256">Σύνολο Μαθήματος (25 ώρες φόρτου εργασίας ανά πιστωτική μονάδα)</td> <td data-bbox="1010 1167 1348 1256">150 ώρες</td> </tr> </tbody> </table>	Δραστηριότητα	Φόρτος Εργασίας Εξαμήνου	Διαλέξεις	26 x 2 = 52 ώρες	Φροντιστηριακές Ασκήσεις που εστιάζουν στην εφαρμογή	26 x 2 = 52 ώρες	Αυτοτελής Μελέτη	44 ώρες	Γραπτές Εξετάσεις	2 x 1 = 2 ώρες	Σύνολο Μαθήματος (25 ώρες φόρτου εργασίας ανά πιστωτική μονάδα)	150 ώρες	
Δραστηριότητα	Φόρτος Εργασίας Εξαμήνου													
Διαλέξεις	26 x 2 = 52 ώρες													
Φροντιστηριακές Ασκήσεις που εστιάζουν στην εφαρμογή	26 x 2 = 52 ώρες													
Αυτοτελής Μελέτη	44 ώρες													
Γραπτές Εξετάσεις	2 x 1 = 2 ώρες													
Σύνολο Μαθήματος (25 ώρες φόρτου εργασίας ανά πιστωτική μονάδα)	150 ώρες													
<p>ΑΞΙΟΛΟΓΗΣΗ ΦΟΙΤΗΤΩΝ Περιγραφή της διαδικασίας αξιολόγησης Γλώσσα Αξιολόγησης, Μέθοδοι αξιολόγησης, Διαμορφωτική ή Συμπερασματική, Δοκιμασία Πολλαπλής Επιλογής, Ερωτήσεις Σύντομης Απάντησης, Ερωτήσεις Ανάπτυξης Δοκιμίων, Επίλυση Προβλημάτων, Γραπτή Εργασία, Εκθεση / Αναφορά, Προφορική Εξέταση, Δημόσια Παρουσίαση, Εργαστηριακή Εργασία, Κλινική Εξέταση Ασθενούς, Καλλιτεχνική Ερμηνεία, Άλλη / Άλλες Αναφέρονται ρητά προσδιορισμένα κριτήρια αξιολόγησης και εάν και που είναι προσβάσιμα από τους φοιτητές.</p>	Γραπτή τελική εξέταση (100%)													

5. ΣΥΝΙΣΤΩΜΕΝΗ-ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Alfred Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, *Compilers, Principles, Techniques, & Tools*, Second Edition, Addison-Wesley, 2007
2. Michael L. Scott, *Programming Language Pragmatics*, 4th edition, Elsevier, 2015
3. Torben Egidius Mogensen, *Introduction to Compiler Design*, Springer-Verlag, 2011
4. Bjarne Stroustrup, *The C++ Programming Language (4th edition)*, Addison-Wesley, 2013