



ΕΠΛ031: ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ

Ακαδημαϊκό Έτος 2005 – 2006
Εαρινό Εξάμηνο

Διδάσκων Καθηγητής:	Χρύσης Γεωργίου, Λέκτορας
Γραφείο:	ΘΕΕ01 016, Πανεπιστημιούπολη
Τηλέφωνο:	22 892745
Ηλεκ. ταχυδρομείο:	chryssis@cs.ucy.ac.cy
Ώρες Γραφείου:	Τρίτη 10:30 – 11:30, 15:00 – 16:00 ή μετά από συνεννόηση
Διαλέξεις:	Τρίτη 12:30 – 14:00, Παρασκευή 13:30 – 15:00, ΧΩΔ01 109
Φροντιστήριο:	Τρίτη 14:00 – 15:00, ΧΩΔ01 109
Εργαστήρια:	Ακροατήριο Α: Τετάρτη 13:00 – 14:00, ΘΕΕ01 B141 Ακροατήριο Β: Τετάρτη 14:00 – 15:00, ΘΕΕ01 B141
Ιστοσελίδα Μαθήματος:	http://www.cs.ucy.ac.cy/~chryssis/EPL031/index.html

1. Γενική Περιγραφή

Το μάθημα αυτό έχει σαν σκοπό τη διδασκαλία των βασικών αρχών του προγραμματισμού και την εφαρμογή αυτών μέσω της γλώσσας προγραμματισμού FORTRAN.

Η ύλη του μαθήματος θα καλύψει το τι είναι αλγόριθμος, πρόγραμμα και τι σημαίνει “σωστός προγραμματισμός”. Θα μελετηθούν τα στάδια επίλυσης ενός προβλήματος μέσω υπολογιστή και θα παρουσιαστούν τα στάδια κατασκευής ενός προγράμματος. Η γλώσσα FORTRAN θα χρησιμοποιηθεί για την υλοποίηση της λύσης στον υπολογιστή.

Στα πλαίσια της διδασκαλίας της γλώσσας FORTRAN, θα δοθεί η δομή ενός τυπικού προγράμματος και θα παρουσιαστούν βασικές εντολές της γλώσσας σχετικά με τη ροή προγράμματος (διακλάδωση, επανάληψη), την είσοδο/έξοδο δεδομένων και την κλήση υπορουτινών. Επίσης θα περιγραφούν βασικές και σύνθετες δομές δεδομένων που παρέχει η γλώσσα. Όλα τα παραπάνω θα συνοδευτούν με παραδείγματα τόσο κατά τη διάρκεια των διαλέξεων όσο και κατά τη διάρκεια των εργαστηρίων.

1.1 Στόχοι

Να διδαχθούν οι βασικές αρχές του προγραμματισμού με έμφαση το δομημένο προγραμματισμό, την αφαιρετικότητα και τη σχεδίαση, υλοποίηση και έλεγχο αρθρωτών προγραμμάτων. Να γίνει εμπέδωση των διδασκόμενων εννοιών μέσω της επίλυσης πρακτικών προβλημάτων στις διαλέξεις και στα εργαστήρια.

1.2 Τρόπος Διδασκαλίας

Η διδασκαλία θα στηρίζεται στις διαλέξεις και στα εργαστήρια. Πέρα από τα εργαστήρια, θα καταβληθεί προσπάθεια για την ενεργή συμμετοχή των φοιτητών/τριών και κατά τη διάρκεια των διαλέξεων με ερωτήσεις και επίλυση ασκήσεων.

Τονίζεται ότι η παρακολούθηση των διαλέξεων και εργαστηρίων είναι όχι μόνο υποχρεωτική, αλλά και αναγκαία για την επιτυχή ολοκλήρωση του μαθήματος.

1.3 Συνοπτικός Πίνακας Ύλης

- Αλγόριθμος – Πρόγραμμα
- Λογικά διαγράμματα ροής
- Αναγνωριστικά στη Fortran – Μεταβλητές
- Τελεστές
- Δομές απόφασης (υπό συνθήκη / κατά περίπτωση)
- Επαναλήψεις (υπό συνθήκη / με αρίθμηση / αέναη)
- Διαδικασίες (συναρτήσεις / υπορουτίνες)
- Φορμαρισμένη είσοδος – έξοδος
- Πίνακες
- Αρχεία
- Δυναμική δέσμευση μνήμης

2. Βιβλίο Μαθήματος

Το βασικό βιβλίο που θα χρησιμοποιηθεί στο μάθημα είναι:

Ματαράς Δ. και Κουτελιέρης Φ., ‘Προγραμματισμός Fortran 90/95 για επιστήμονες και μηχανικούς’, Εκδόσεις ΤΖΙΟΛΑ, 1^η, 2003, ISBN 960-8050-43-X

Σαν βοήθημα προτείνετε επίσης:

Nyhoff L. and Leestma S., ‘Introduction to Fortran 90’, Prentice Hall 1999, ISBN 0-13-013146-6

Το πιο πάνω βιβλίο, παρόλο που είναι στα Αγγλικά, συστήνεται διότι είναι συνοπτικό και περιέχει πολλά παραδείγματα και ασκήσεις.

3. Αξιολόγηση και Βαθμολόγηση

Η αξιολόγηση των φοιτητών/τριών θα προκύψει από τις επιδόσεις τους τόσο στις κατ’ οίκον προγραμματιστικές εργασίες όσο και στις εξετάσεις και διαγνωστικά. Αναλυτικά, η τελική βαθμολογία θα υπολογιστεί με βάση τους ακόλουθους συντελεστές:

Κατ’ οίκον εργασίες και Διαγνωστικά	30%
Εξέταση ημι-εξαμήνου	20%
Τελική εξέταση	50%

Η εκπόνηση *όλων* των εργασιών όπως και η συμμετοχή σε *όλες* τις εξετάσεις και διαγνωστικά είναι *υποχρεωτική*. Σε αντίθετη περίπτωση δεν θα επιτραπεί στον φοιτητή/τρια να δώσει την τελική εξέταση. Συνολική απόδοση χαμηλότερη του **50%** στις δύο εξετάσεις του μαθήματος σημαίνει *αυτόματα αποτυχία* στο μάθημα. Σε αυτήν την περίπτωση ο τελικός βαθμός του μαθήματος που θα δοθεί στο φοιτητή/τρια θα είναι ίσος με το συνολικό βαθμό του στις δύο εξετάσεις.

3.1 Εξετάσεις (Ημι-εξαμήνου και Τελική)

Και οι δύο εξετάσεις του μαθήματος θα διεξαχθούν με κλειστά βιβλία και κλειστές σημειώσεις. Η χρήση **οποιοδήποτε βοηθήματος** στη διάρκεια μιας εξέτασης αποτελεί πράξη δόλου που στην *καλύτερη περίπτωση* θα οδηγήσει σε μηδενισμό.

3.2 Διαγνωστικά

Ένα διαγνωστικό είναι μια προγραμματιστική εργασία υπό τη μορφή εξέτασης που θα εκτελείται ατομικά σε εργαστήριο υπολογιστών υπό την επίβλεψη διδακτικού προσωπικού (σε ώρα εργαστηριακού μαθήματος). Η ημερομηνία διεξαγωγής του κάθε διαγνωστικού θα ανακοινώνεται περίπου μια εβδομάδα πριν, προφορικά (στην τάξη) και ηλεκτρονικά (στην ιστοσελίδα του μαθήματος).

3.3 Κατ' οίκον Εργασίες

Οι κατ' οίκον προγραμματιστικές ασκήσεις αποτελούν το σημαντικότερο μέσο εμπέδωσης της ύλης του μαθήματος. Στόχος τους είναι η απόκτηση (α) εμπειρίας στην εφαρμογή των προγραμματιστικών τεχνικών που θα διδαχθούν στο μάθημα και (β) ευχέρειας στη χρήση της γλώσσας FORTRAN.

Οι λύσεις των ασκήσεων θα κρίνονται με βάση τα ακόλουθα κριτήρια:

I. **Ορθότητα.** Ένα πρόγραμμα που δεν δίνει τη σωστή απάντηση δεν έχει καμία αξία. Η ορθότητα των προγραμμάτων θα πρέπει να ελέγχεται προσεκτικά πριν την παράδοσή τους.

II. **Σαφήνεια.** Ο αλγόριθμος που χρησιμοποιείται πρέπει να περιγράφεται με σαφήνεια και ο κώδικας FORTRAN να είναι γραμμένος καθαρά και κατάλληλα σχολιασμένος.

III. **Χρόνος εκτέλεσης.** Σε κάθε περίπτωση θα πρέπει να επιδιώκεται η συγγραφή προγραμμάτων που επιλύουν το πρόβλημα με το μικρότερο δυνατό κόστος σε χρόνο εκτέλεσης και/ή χώρο μνήμης. Θα πρέπει ωστόσο να διατηρείται στο μέγιστο δυνατό η σαφήνεια του κώδικα.

Κώδικας που δεν μεταγλωττίζεται σωστά παίρνει **βαθμολογία 0%**. Οι εργασίες πρέπει να **παραδίδονται την προκαθορισμένη ημερομηνία παράδοσης**. Εργασίες που θα παραδίδονται καθυστερημένα (είναι υποχρεωτική η παράδοση εργασιών, έστω και καθυστερημένα) θα βαθμολογούνται **έως 50%**. Αναλόγως της περίπτωσης, καθυστερημένες εργασίες δύναται να πάρουν βαθμολογία **0%**.

Δόλος: Η ανταλλαγή απόψεων πάνω στις εργασίες συνίσταται. ΟΜΩΣ, η αντιγραφή κώδικα, ή η παροχή κώδικα προς αντιγραφή, ή η τροποποίηση των αποτελεσμάτων της δοκιμής του κώδικα, αποτελούν πράξεις δόλου που στην **καλύτερη περίπτωση θα οδηγήσουν σε μηδενισμό**. Ένας καλός κανόνας για αποφυγή αντιγραφής είναι να μην κοιτάζετε τον κώδικα άλλων και να μην δίνετε αντίγραφα του δικού σας κώδικα. Άλλος ένας κανόνας είναι να συζητάτε τα προβλήματα με τη χρήση πίνακα αλλά να μην παίρνετε σημειώσεις κατά τη συζήτηση με άλλους. *Όλες οι εργασίες θα ελέγχονται με χρήση λογισμικού που αναγνωρίζει παρόμοια δομή κώδικα.*