



ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΕΠΛ131 Αρχές Προγραμματισμού I

Ακαδημαϊκό Έτος 2015/16 – Εαρινό Εξάμηνο

ΕΝΔΙΑΜΕΣΗ ΕΞΕΤΑΣΗ

ΗΜΕΡΟΜΗΝΙΑ: 16 Μαρτίου 2016
ΔΙΑΡΚΕΙΑ: 9:00πμ– 11:30πμ
ΑΙΘΟΥΣΑ: Κτήριο ΘΕΕ01, Αίθουσα 202
ΔΙΔΑΣΚΩΝ: Ελπίδα Κεραυνού-Παπαηλιού

Απαντήστε όλες τις ερωτήσεις

Κάθε ερώτηση βαθμολογείται με 25 μονάδες.

Ερώτηση 1

Στο αρχείο marks131 βρίσκονται τα στοιχεία που αφορούν τις βαθμολογίες κάποιας ομάδας φοιτητών που ολοκλήρωσαν το μάθημα ΕΠΛ131. Για παράδειγμα, το εν λόγω αρχείο μπορεί να περιλαμβάνει τα εξής στοιχεία:

1	89	76	95	65	72
2	67	54	89	54	75
3	75	90	85	90	85
4	67	86	95	45	50
5	45	64	0	24	21

Η πρώτη στήλη δίνει τον αύξοντα αριθμό του φοιτητή. Οι υπόλοιπες στήλες δίνουν βαθμολογίες από το 100. Συγκεκριμένα, η δεύτερη, τρίτη και τέταρτη στήλη δίνουν τις (συνολικές) βαθμολογίες σε τρεις κατηγορίες προγραμματιστικών ασκήσεων, η πέμπτη στήλη δίνει τις βαθμολογίες της ενδιάμεσης εξέτασης και η έκτη στήλη δίνει τις βαθμολογίες της τελικής εξέτασης. Συνεπώς, η κάθε σειρά δίνει τις βαθμολογίες για ένα φοιτητή.

Αναπτύξτε πρόγραμμα σε Java, το οποίο υπολογίζει τον τελικό βαθμό του κάθε φοιτητή στο μάθημα, λαμβάνοντας υπόψη ότι οι προγραμματιστικές ασκήσεις έχουν συνολική βαρύτητα 30% (όπου η κάθε κατηγορία προγραμματιστικής άσκησης έχει την ίδια βαρύτητα), η ενδιάμεση εξέταση έχει βαρύτητα 20% και η βαρύτητα της τελικής εξέτασης είναι 50%. Εάν ο βαθμός της τελικής εξέτασης είναι μεγαλύτερος από το βαθμό της ενδιάμεσης εξέτασης, τότε ανάμεσα στις δύο γραπτές εξετάσεις μετρά μόνο η τελική

εξέταση, κατά 70%. Ο τελικός βαθμός υπολογίζεται από το 10.0 με ακρίβεια δύο δεκαδικών ψηφίων. Ο τελικός βαθμός του κάθε φοιτητή και η κατηγορία του τελικού αυτού βαθμού, καθώς και η σύνοψη των αποτελεσμάτων, παρουσιάζονται ως εξής για το συγκεκριμένο παράδειγμα:

1	7.64	Very Good
2	7.35	Very Good
3	8.55	Excellent
4	5.98	Good
5	2.62	Fail

Average Mark: 6.43

Excellent: 1

Very Good: 2

Good: 1

Fail: 1

Η κατηγορία του τελικού βαθμού, TB, ενός φοιτητή προσδιορίζεται ως εξής:

Excellent, αν $TB \geq 8.5$

Very Good, αν $8.5 > TB \geq 6.5$

Good, αν $6.5 > TB \geq 5.0$

Fail, αν $TB < 5.0$

Η σύνοψη των αποτελεσμάτων αποτελείται από το μέσο όρο των τελικών βαθμών των φοιτητών, καθώς επίσης της κατανομής των τελικών βαθμών σύμφωνα με τις πιο πάνω κατηγορίες.

Σημείωση: Θεωρείστε ότι σε κάθε εκτέλεση του προγράμματός σας, το αρχείο εισόδου προσδιορίζεται στη γραμμή εντολής μέσω επανα-κατεύθυνσης (file redirection) και ως εκ τούτου στον κώδικά σας μπορείτε να χρησιμοποιήσετε εντολές για standard input.

Ερώτηση 2

Το έμμεσο κόστος διεξαγωγής ενός ερευνητικού προγράμματος ονομάζεται το υπερκεφαλικό κόστος (overhead) και υπολογίζεται σκαλωτά από το σύνολο της χρηματοδότησης του ερευνητικού προγράμματος ως εξής:

Μέχρι €10000: δεν υπάρχει υπερκεφαλικό κόστος

Από €10001 μέχρι €20000: 5% του ποσού

Από €20001 μέχρι €50000: 10% του ποσού

Πάνω από €50000: 15% του ποσού

Για παράδειγμα, το υπερκεφαλικό κόστος ενός ερευνητικού προγράμματος του οποίου η συνολική χρηματοδότηση είναι €125000 ισούται με €14750 ($0.05 \times 10000 + 0.1 \times 30000 + 0.15 \times 75000$), ενώ το υπερκεφαλικό κόστος ενός ερευνητικού προγράμματος του οποίου η συνολική χρηματοδότηση είναι

€45000 ισούται με €3000 ($0.05 \times 10000 + 0.1 \times 25000$). Τέλος ένα ερευνητικό πρόγραμμα με συνολική χρηματοδότηση €8000, δεν έχει υπερκεφαλικό κόστος.

Αναπτύξτε ένα πρόγραμμα το οποίο λαμβάνει τις συνολικές χρηματοδοτήσεις ενός αριθμού ερευνητικών προγραμμάτων και υπολογίζει το υπερκεφαλικό κόστος για το κάθε πρόγραμμα, καθώς και το συνολικό υπερκεφαλικό κόστος για όλα τα προγράμματα, όπως δεικνύεται παρακάτω (το input από τον χρήστη απεικονίζεται υπογεγραμμένο):

Enter number of projects: 3

Enter budget for project No. 1: 125000

Overhead for project No. 1: 14750.00

Enter budget for project No. 2: 45000

Overhead for project No. 2: 3000.00

Enter budget for project No. 3: 8000

Overhead for project No. 3: 0.00

Overall overhead: 17750.00

Ερώτηση 3

Κατασκευάστε πρόγραμμα το οποίο υπολογίζει τον αριθμό της ημέρας (από το 1 στο 366) σε κάποιο έτος για δεδομένη ημερομηνία. Για παράδειγμα, η 1^η Ιανουαρίου 1994 είναι η 1^η μέρα του 1994, η 31^η Δεκεμβρίου 1993 είναι η 365^η μέρα του 1993, ενώ η 31^η Δεκεμβρίου 1996 είναι η 366^η μέρα του 1996 αφού το 1994 ήταν δίσεκτο. Ένα έτος είναι δίσεκτο εάν διαιρείται με το 4, εκτός και αν διαιρείται και με το 100, οπότεν πρέπει να διαιρείται και με το 400. Το πρόγραμμα λαμβάνει από το χρήστη την ημέρα, μήνα και έτος ως φυσικούς αριθμούς (θετικούς ακέραιους αριθμούς) διασφαλίζοντας ότι τα στοιχεία που δίνει ο χρήστης είναι έγκυρα. Το πρόγραμμα πρέπει να κάνει χρήση της συνάρτησης leap η οποία επιστρέφει true εάν το όρισμά της είναι δίσεκτο έτος, διαφορετικά επιστρέφει false. Η συνάρτηση leap πρέπει να οριστεί.

Πιο κάτω δίνονται κάποια παραδείγματα χρήσης του προγράμματος, όπου το input από τον χρήστη απεικονίζεται υπογεγραμμένο:

Παράδειγμα 1

Enter date as day, month, year: 1 1 1994

This is day 1

Παράδειγμα 2

Enter date as day, month, year: 31 12 1993

This is day 365

Παράδειγμα 3

Enter date as day, month, year: 31 12 2016
This is day 366

Παράδειγμα 4

Enter date as day, month, year: 3 2 2000
This is day 34

Ερώτηση 4

Ορίστε τη συνάρτηση `max3` η οποία λαμβάνει τρεις ακέραιους αριθμούς και επιστρέφει το μεγαλύτερο αριθμό.

Κατασκευάστε ένα πρόγραμμα το οποίο λαμβάνει N τριάδες ακέραιων αριθμών, όπου το N προσδιορίζεται από το χρήστη, και παρουσιάζει την τριάδα η οποία περιέχει το μεγαλύτερο αριθμό. Το πρόγραμμα πρέπει να κάνει χρήση της συνάρτησης `max3`.

Πιο κάτω δίνονται κάποια παραδείγματα χρήσης του προγράμματος, όπου το `input` από τον χρήστη απεικονίζεται υπογεγραμμένο:

Παράδειγμα 1

Enter number of triplets: 4
Enter first triplet of numbers: 67 34 12
Enter next triplet of numbers: 68 45 30
Enter next triplet of numbers: 125 6 1
Enter next triplet of numbers: 67 45 7
Maximum Triplet: 125 6 1

Παράδειγμα 2

Enter number of triplets: 1
Enter first triplet of numbers: 45 67 12
Maximum Triplet: 45 67 12

Παράδειγμα 3

Enter number of triplets: 2
Enter first triplet of numbers: 3 9 7
Enter next triplet of numbers: 3 3 3
Maximum Triplet: 3 9 7

Παράδειγμα 4

Enter number of triplets: 3
Enter first triplet of numbers: 45 8 3
Enter next triplet of numbers: 5 9 45
Enter next triplet of numbers: 1 2 3
Maximum Triplet: 45 8 3

ΤΕΛΟΣ ΕΡΩΤΗΣΕΩΝ

ΧΩΡΟΣ ΑΠΑΝΤΗΣΕΩΝ

Όνοματεπώνυμο Φοιτητή: -----

Ταυτότητα: -----

Υπογραφή: -----

Ερωτήσεις	Μονάδες
Ερώτηση 1	
Ερώτηση 2	
Ερώτηση 3	
Ερώτηση 4	
Σύνολο Μονάδων	

Παρατηρήσεις Διδάσκοντα

Απάντηση στην Ερώτηση 1

(τυχόν συνέχεια στην απάντηση της Ερώτησης 1)

Απάντηση στην Ερώτηση 2

(τυχόν συνέχεια στην απάντηση της Ερώτησης 2)

Απάντηση στην Ερώτηση 3

(τυχόν συνέχεια στην απάντηση της Ερώτησης 3)

Απάντηση στην Ερώτηση 4

(τυχόν συνέχεια στην απάντηση της Ερώτησης 4)

Επιπρόσθετο φύλλο (για συμπλήρωση απάντησης ή πρόχειρο)

Επιπρόσθετο φύλλο (για συμπλήρωση απάντησης ή πρόχειρο)

Επιπρόσθετο φύλλο (για συμπλήρωση απάντησης ή πρόχειρο)

Επιπρόσθετο φύλλο (για συμπλήρωση απάντησης ή πρόχειρο)

ΤΕΛΟΣ ΕΞΕΤΑΣΤΙΚΟΥ ΔΟΚΙΜΙΟΥ