

3222: Προγραμματισμός Υπολογιστών με Java (Μ-Ω)

Άσκηση 1

Παραδοτέα: Application1.java και Application2.java

Ημερομηνία παράδοσης: 11/4/2021 (e-class)

e-mail: mst@aueb.gr

Αντικείμενο εργασίας: Σύνταξη και εκτέλεση προγράμματος Java, βασικοί τύποι δεδομένων, υπολογισμός παραστάσεων, υπό συνθήκη εκτέλεση εντολών και εντολές ανακύκλωσης, είσοδος/έξοδος, δημιουργία και κλήση στατικών μεθόδων (συναρτήσεων), χρήση βιβλιοθηκών, κλάσεις, δημιουργία αντικειμένων, δημιουργία και κλήση μη στατικών μεθόδων.

Άσκηση 1.1 (class Application1)

Δίνεται το αρχείο Application1.java. Συμπληρώστε τον κώδικα με βάση την παρακάτω περιγραφή:

Συνάρτηση Q1- `public static boolean isPrime (int n)`

Η συνάρτηση **isPrime** ελέγχει αν ο ακέραιος n είναι πρώτος αριθμός (Prime number), όπου $n > 1$. Αν το n είναι πρώτος αριθμός η συνάρτηση επιστρέφει `true`, διαφορετικά επιστρέφει `false`. Πρώτος είναι ένας ακέραιος που διαιρείται μόνο με το 1 και τον εαυτό του.

Για παράδειγμα, prime numbers είναι το 2, 3, 5, 7 και 11.

Συνάρτηση Q2 - `public static int factorial (int n)`

Η συνάρτηση **factorial** επιστρέφει το παραγοντικό του n . Το παραγοντικό ορίζεται ως το γινόμενο όλων των ακεραίων από το 1 ως το n , για κάθε $n \geq 1$. Ειδικά το $0! = 1$.

$$n! = 1 \cdot 2 \cdot 3 \cdots (n - 2) \cdot (n - 1) \cdot n$$

Για παράδειγμα το $4!$ είναι $1 \cdot 2 \cdot 3 \cdot 4 = 24$.

Συνάρτηση Q3 - public static int combinations (int n, int k)

Η συνάρτηση **combinations** υπολογίζει και επιστρέφει τους συνδυασμούς των n αντικειμένων ανά k. Ο υπολογισμός των συνδυασμών γίνεται με βάση τον τύπο:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

Στη συνάρτηση combinations, όπου απαιτείται υπολογισμός παραγοντικού, να γίνεται με κλήση της συνάρτησης factorial (συνάρτηση Q2).

Συνάρτηση Q4 - public static int digitsOfInteger (int n)

Η συνάρτηση **digitsOfInteger** υπολογίζει και επιστρέφει το πλήθος των ψηφίων του ακεραίου n, όπου n ≥ 0.

Συνάρτηση Q5 - public static void quadraticEquation (double a, double b, double c)

Η συνάρτηση **quadraticEquation** επιλύει τη δευτεροβάθμια εξίσωση:

$$ax^2 + bx + c = 0$$

Με βάση τον τύπο:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Αν η διακρίνουσα ($b^2 - 4ac$) είναι αρνητικός αριθμός τότε εμφανίζει μήνυμα ότι δεν υπάρχουν πραγματικές τιμές για την επίλυση της εξίσωσης, διαφορετικά εμφανίζει τις τιμές αυτές με δύο δεκαδικά ψηφία. Υποθέστε ότι $a \neq 0$.

Συνάρτηση Q6 - `public static double findArea(double a, double b, double c)`

Η συνάρτηση **findArea** επιστρέφει το εμβαδόν ενός τριγώνου όπως αυτό καθορίζεται από τα μήκη των πλευρών του *a*, *b* και *c*. Αν με βάσει τα μήκη των πλευρών δεν ορίζεται τρίγωνο τότε η συνάρτηση επιστρέφει -1, δηλαδή όταν προκύπτει αρνητική υπόριξη ποσότητα στον τύπο του εμβαδού (τύπος του Ήρωνα).

Συνάρτηση Q7 - `public static void reverseDigits (int n)`

Η συνάρτηση **reverseDigits** υπολογίζει και επιστρέφει τον αντίστροφο ενός θετικού ακεραίου (*n*).

Για παράδειγμα ο αντίστροφος αριθμός του 5461 είναι 1645.

Συνάρτηση Q8 - `public static int minDigit(int n)`

Η συνάρτηση **minDigit** υπολογίζει και επιστρέφει το μικρότερο ψηφίο ενός θετικού ακεραίου (*n*).

Για παράδειγμα το μικρότερο ψηφίο του 5461 είναι το 1.

Κύριο Πρόγραμμα: `public static void main(String args[])`

Στο κύριο πρόγραμμα διαβάστε από το χρήστη τα απαιτούμενα δεδομένα και δώστε ένα παράδειγμα κλήσης της κάθε συνάρτησης (Q1-Q8).

Παρακάτω δίνεται ένα παράδειγμα εκτέλεσης του προγράμματος στο οποίο μπορείτε να δείτε την είσοδο και την έξοδό του:

Q1 - Prime Number (n)

n (>1) = 7

7 is a Prime Number

Q2 - Factorial (n)

n= 4

Factorial (4) = 24

Q3 - Combinations (n, k)

n= 5

k= 2

Combinations (5,2) = 10

Q4 - Number of digits

Enter a positive integer:

n = 6543

Number of digits = 4

Q5 - Quadratic equation...

Enter a (<>0): 1

Enter b: -5

Enter c: 3

The first solution is: 4,30

The second solution is: 0,70

Q6 - Area of a triangle

Enter the length of the three sides.

Enter a: 4

Enter b: 4

Enter c: 5

The area of a triangle is: 7,81

Q7 - Reverse digits

Enter a positive integer:

n= 6543

The reversed number is = 3456

Q8 - Min Digit

Enter a positive integer:

n= 6543

Min Digit = 3

Άσκηση 1.2 (class Application2)

Δημιουργείστε ένα αντίγραφο της κλάσης Application1 με όνομα Application2 (αρχείο Application2.java).

Στην κλάση Application2 να υλοποιήσετε την άσκηση με non-static συναρτήσεις, για παράδειγμα:

```
public boolean isPrime (int n)
```

Προσαρμόστε το κύριο πρόγραμμα έτσι ώστε να είναι δυνατή η κλήση των συναρτήσεων.

Παρατηρήσεις:

1. Σε όλους τους παραπάνω υπολογισμούς αγνοείτε την περίπτωση της υπερχείλισης (για παράδειγμα όταν προσπαθείτε να υπολογίσετε το παραγοντικό μεγάλου αριθμού).
2. Υποθέστε ότι τα δεδομένα που δίνει ο χρήστης είναι έγκυρα, για παράδειγμα ο χρήστης δίνει θετικό ακέραιο όπου απαιτείται.
3. Μη χρησιμοποιείτε ελληνικούς χαρακτήρες ούτε ως σχόλια, ούτε ως μηνύματα προς το χρήστη.
4. Κάθε πρόγραμμα σας πρέπει να μεταγλωττίζεται και να εκτελείται στη γραμμή εντολών (όχι μέσω κάποιου IDE).
5. Η άσκηση είναι ατομική και πρέπει να γίνει από όλους τους φοιτητές, είτε επιθυμείτε να κρατήσετε εργασίες από προηγούμενη χρονιά είτε όχι.
6. Η βαθμολογία της 1^{ης} άσκησης αντιστοιχεί στο 5% του τελικού βαθμού του μαθήματος.
7. Σε κάθε πρόγραμμα (Application1.java και Application2.java) αναφέρατε το όνομά σας, με κεφαλαίους λατινικούς χαρακτήρες και τον αριθμό μητρώου σας, ως σχόλιο στην αρχή του προγράμματος.
8. Παρακαλούμε να υποβάλετε και τα δύο προγράμματα (Application1.java και Application2.java) ως εργασία στο eclass (Άσκηση 1), **αφού πρώτα τα συμπίεσετε (zip files)** έτσι ώστε να γίνουν δεκτά από το eclass.