

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών
Τομέας Μαθηματικών

Σχεδίαση – Ανάπτυξη Εφαρμογών Πληροφορικής
11 Σεπτεμβρίου 2014

- Διάρκεια 2:00 ώρες
- Να απαντηθούν **ΟΛΑ (5)** τα θέματα.
- Καλή επιτυχία.

Όνοματεπώνυμο:	A. Μητρώου:
----------------	-------------

Θέμα 1^ο

Να υλοποιηθεί η στατική μέθοδος `FibonacciGreaterThan()` η οποία δέχεται ως παράμετρο ένα θετικό ακέραιο `k` και υπολογίζει τον μικρότερο αριθμό `n` για τον οποίον ισχύει ότι ο `n`-οστός αριθμός Fibonacci `F(n)` είναι μεγαλύτερος ή ίσος του `k`.

Σημείωση: Ο `n`-οστός αριθμός Fibonacci, `F(n)`, ορίζεται για θετικούς ακεραίους ως εξής:

$$F(1)=1$$

$$F(2)=1$$

$$F(n) = F(n-1) + F(n-2), \quad n > 2$$

Θέμα 2^ο

Να δημιουργηθεί η στατική μέθοδος `numberedCells(int r, int c)` η οποία δημιουργεί και επιστρέφει ένα δισδιάστατο διάνυσμα (`r` γραμμών και `c` στηλών) από ακεραίους. Το στοιχείο σε κάθε θέση είναι ο αύξων αριθμός της θέσης (ξεκινώντας από το μηδέν) εάν η αρίθμηση γίνεται ανά γραμμές και σε κάθε γραμμή από αριστερά προς τα δεξιά.

Θέμα 3^ο

Δίνεται η κλάση `Person` (άνθρωπος) η οποία χρησιμοποιείται στην μοντελοποίηση ενός ανθρώπου. Κάθε αντικείμενο της κλάσης `Person` υλοποιεί τις μεθόδους:

<code>Person(String name, int yearOfBirth)</code>	Κατασκευαστής. Θέτει το όνομα και τη χρονολογία γέννησης του κάθε ανθρώπου.
<code>void setName(String newName)</code>	Θέτει το όνομα.
<code>void setYoB(int newYoB)</code>	Θέτει τη χρονολογία γέννησης.
<code>void setGender(char c)</code>	Θέτει το φύλλο. «M» για άντρες, «F» για γυναίκες.
<code>String getName()</code>	Επιστρέφει το όνομα.
<code>int getYoB()</code>	Επιστρέφει τη χρονολογία γέννησης.
<code>char getGender()</code>	Επιστρέφει το φύλλο.
<code>String toString()</code>	Εκτυπώνει τον άνθρωπο (σε μία γραμμή εξόδου)

Να γραφεί η στατική μέθοδος `oldestMale()` η οποία δέχεται ως παράμετρο ένα διάνυσμα από αντικείμενα τύπου `Person` και επιστρέφει τη χρονολογία γέννησης του γρηαιότερου από τους άντρες του διανύσματος.

Θέμα 4^ο (Κλάσεις και υλοποιήσεις τους)

Δίνεται η κλάση **Point** η οποία μοντελοποιεί ένα σημείο στο επίπεδο (καθορισμένο από δύο ακέραιες συντεταγμένες) και υποστηρίζει τις μεθόδους:

- | | |
|---------------------------------|---|
| 1. Point(int x, int y). | Κατασκευάζει το σημείο (x,y) |
| 2. void setX(int x) | Θέτει/μετατρέπει την X-συντεταγμένη του σημείου |
| 3. void setY(int y) | Θέτει/μετατρέπει την Y-συντεταγμένη του σημείου |
| 4. int getX() | Επιστρέφει την X-συντεταγμένη του σημείου |
| 5. int getY() | Επιστρέφει την Y-συντεταγμένη του σημείου |
| 6. double distanceFrom(Point p) | Επιστρέφει την απόσταση από το σημείο p. |

Να υλοποιηθεί η κλάση **LS** (ευθύγραμμο τμήμα) η οποία έχει σκοπό να μοντελοποιήσει ένα ευθύγραμμο τμήμα. Το κάθε ευθύγραμμο τμήμα ορίζεται από τα δύο άκρα του (σημεία στο επίπεδο, τύπου **Point** στις δύο διαστάσεις).

Αντικείμενα της κλάσης **LineSegment** υποστηρίζουν τις παρακάτω μεθόδους:

1. LS(Point a, Point b)	Κατασκευάζει ένα ευθύγραμμο τμήμα με άκρα τα σημεία a και b .
2. getFirstEndPoint()	Επιστρέφει το πρώτο άκρο (δηλ. το σημείο a) του ευθύγραμμου τμήματος.
3. getSecondEndPoint()	Επιστρέφει το δεύτερο άκρο (δηλ. το σημείο b) του ευθύγραμμου τμήματος.
3. length()	Επιστρέφει το μήκος του ευθύγραμμου τμήματος.
4. isDiagonal()	Επιστρέφει true εάν το ευθύγραμμο τμήμα είναι παράλληλο με μία από τις διαγωνίους του συστήματος αξόνων X-Y, false διαφορετικά.
5. slope()	Επιστρέφει την «κλίση» του ευθύγραμμου τμήματος. Η μέθοδος καλείται μόνο όταν το ευθύγραμμο τμήμα δεν είναι κάθετο στον άξονα των X. Σημείωση: Ως κλίση του ευθύγραμμου τμήματος <u>ορίζεται</u> η εφαπτομένη της γωνίας που σχηματίζει το ευθύγραμμο τμήμα με μία ευθεία που διέρχεται από το σημείο του με μικρότερη X-συντεταγμένη και είναι παράλληλη με τον άξονα των X.

Θέμα 5^ο (Διαπροσωπείες και υλοποιήσεις τους)

Η διαπροσωπεία **IntPriorityQueue** μοντελοποιεί μία **ουρά προτεραιότητας ακεραίων**, δηλαδή μία δομή δεδομένων που αποτελείται από ένα σύνολο ακεραίων (ΔΕΝ επιτρέπονται οι επαναλήψεις) και υποστηρίζει τις παρακάτω μεθόδους:

- | | |
|-----------------------|---|
| void insert(int elem) | Εισάγει το στοιχείο-ακέραιο elem στην ουρά προτεραιότητας, εάν αυτό δεν υπάρχει ήδη. Υποθέτει ότι η ουρά προτεραιότητας δεν είναι γεμάτη. |
| int deleteMax() | Διαγράφει και επιστρέφει το στοιχείο με την μεγαλύτερη προτεραιότητα. Εξ ορισμού, αυτό είναι το « μεγαλύτερο » στοιχείο της ουράς. |
| int size() | Επιστρέφει τον αριθμό των στοιχείων που βρίσκονται στην ουρά προτεραιότητας. |
| boolean isFull() | Ελέγχει εάν η ουρά προτεραιότητας είναι γεμάτη. |

Να αναπτυχθεί κώδικας για την κλάση **ArrayIntPQ** η οποία υλοποιεί τη διαπροσωπεία **IntPriorityQueue** ως ένα **διατεταγμένο σε φθίνουσα διάταξη** διάνυσμα από ακεραίους. Το μέγεθος της ουράς προτεραιότητας, και κατά συνέπεια του διανύσματος, να δοθεί ως παράμετρος στον κατασκευαστή της **ArrayIntPQ**.