

Σχεδίαση – Ανάπτυξη Εφαρμογών Πληροφορικής
23 Ιουνίου 2011

- Διάρκεια 2:30 ώρες
- Να απαντηθούν **ΟΛΑ** (5) τα θέματα.
- Καλή επιτυχία.

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

Θέμα 1^ο

Να υλοποιηθεί η στατική μέθοδος `PowerOf2LargerThan()` η οποία δέχεται ως παράμετρο ένα θετικό ακέραιο n και υπολογίζει την μικρότερη δύναμη του δύο η οποία είναι μεγαλύτερη του n . Να μην χρησιμοποιηθούν μέθοδοι βιβλιοθήκης. Εάν χρησιμοποιήσετε κάποια άλλη μέθοδο (πχ. ύψωση σε δύναμη) πρέπει δώσετε και την υλοποίησή της.

Θέμα 2^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>String getName()</code>	Επιστρέφει το όνομα.
<code>int getYoB()</code>	Επιστρέφει τη χρονολογία γέννησης.
<code>String toString()</code>	Εκτυπώνει τον άνθρωπο (σε μία γραμμή εξόδου)

Να γραφεί η στατική μέθοδος `averageAge()` η οποία δέχεται ως παραμέτρους i) ένα διάνυσμα από αντικείμενα τύπου `Person` και ii) την τρέχουσα χρονολογία (τύπου `int`), και επιστρέφει τη μέση ηλικία των ανθρώπων του διανύσματος.

Θέμα 3^ο

Το κόσκινο του Ερατοσθένη είναι μία μέθοδος που υπολογίζει τους πρώτους αριθμούς που είναι μικρότεροι ή ίσοι από έναν θετικό ακέραιο max . Η μέθοδος χρησιμοποιεί ένα σύνολο από ακεραίους που αρχικά περιέχει όλους τους ακεραίους από το 0 έως και το max , και εκτελεί τα ακόλουθα βήματα:

1. Σβήσε το 0 και το 1 από το σύνολο
2. Βρες τον επόμενο ακέραιο που ανήκει στο σύνολο και δεν έχει εξεταστεί μέχρι τώρα, έστω N , και σβήσε όλα τα πολλαπλάσια του ($2N, 3N, \dots$) από το σύνολο. Προσοχή! Ο N παραμένει στο σύνολο.
3. Συνέχισε με το βήμα 2 έως ότου εξεταστούν όλοι οι αριθμοί που ανήκουν στο σύνολο.

Να υλοποιηθεί η στατική μέθοδος `eratosthenesSieve()` η οποία δέχεται ως παράμετρο ένα θετικό ακέραιο max και εκτυπώνει όλους τους πρώτους αριθμούς στο διάστημα $1..max$.

Το σύνολο των ακεραίων στο διάστημα $[0..max]$ να υλοποιηθεί ως ένα διάνυσμα, έστω A , από Boolean όπου $A[i] == true$ εάν ο ακέραιος i ανήκει στο σύνολο, διαφορετικά $A[i] == false$.

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. setX(int x) | Θέτει/μετατρέπει την X-συντεταγμένη του σημείου |
| 3. setY(int y) | Θέτει/μετατρέπει την Y-συντεταγμένη του σημείου |
| 4. getX() | Επιστρέφει την X-συντεταγμένη του σημείου |
| 5. getY() | Επιστρέφει την Y-συντεταγμένη του σημείου |

Να υλοποιηθεί η κλάση **Square** η οποία έχει σκοπό να μοντελοποιήσει ένα τετράγωνο (παράλληλο με τους άξονες). Το κάθε τετράγωνο έχει ως σημείο αναφοράς την κάτω-αριστερά κορυφή του (τύπου **Point** στις δύο διαστάσεις) και την πλευρά του (τύπου **double**).

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

1. Square(Point a, double side)	Κατασκευάζει ένα τετράγωνο με σημείο αναφοράς το a και πλευρά side .
2. getReferencePoint()	Επιστρέφει το σημείο αναφοράς του τετραγώνου (τύπου Point).
3. getSide()	Επιστρέφει την πλευρά του τετραγώνου.
4. containsPoint(Point p)	Επιστρέφει true εάν το τετράγωνο περιέχει το σημείο p στο εσωτερικό του, false διαφορετικά.
5. isLargerThan(Square s)	Επιστρέφει true εάν το τετράγωνο είναι μεγαλύτερο από το τετράγωνο s , false διαφορετικά.
6. containsSquare(Square r)	Επιστρέφει true εάν το τετράγωνο περιέχει το τετράγωνο r , false διαφορετικά.

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 (επιτρέπονται οι επαναλήψεις) και υποστηρίζει τις παρακάτω μεθόδους:

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

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