



Αντικειμενοστρεφής Προγραμματισμός

Εαρινό Εξάμηνο 2021-22 - Α' Εξεταστική

Εισηγητής: Αδαμίδης Παναγιώτης
Ασδρέ Κατερίνα

Τρίτη 14 Ιουνίου 2022 (14:00-15:30)

ΘΕΜΑΤΑ (Κ2)

ΘΕΜΑ 1 (3 μονάδες)

1. (Βαθμός 1) Εξηγήστε γιατί συμφωνείτε ή διαφωνείτε με τις παρακάτω προτάσεις:
 - I. Ο garbage collector καθαρίζει το stack.
 - II. Ο προσδιοριστή πρόσβασης `protected` στη μέθοδο κάποιας κλάσης, επιτρέπει την πρόσβαση στην μέθοδο μόνο από τις κλάσεις που την κληρονομούν.
 - III. Για τις παρακάτω εντολές `catch` που ακολουθούν ένα `block try` η σωστή σειρά είναι: `Throwable`, `ArithmeticException`, `Exception`, `IOException`.
 - IV. Εάν μία μέθοδος είναι `static` τότε έχω πρόσβαση σε αυτήν μόνο μέσω του ονόματος της κλάσης.
2. (Βαθμός 1) Επιλέξτε ποια θεωρείτε σωστά (Σωστό: +0.5, Λάθος: -0.3)
 - I. Οι κλάσεις `Wind`, `Strings`, και `Keyboards` επεκτείνουν την κλάση `Instrument`. Ποια από τα παρακάτω είναι σωστά:
 - a. `Strings[] ins2 = new Instrument [4];`
 - b. `Strings str = {new Strings (), new Strings (), new Instrument ()};`
 - c. `Instrument [] ins = {new Keyboards (), new Wind (), new Keyboards ()};`
 - d. `Keyboards [] ins1 = new Keyboards [3];`
 - e. `Wind [] com = { new Wind (), new Strings (), new Keyboards () };`
 - II. Δίνονται τα interfaces **A** και **B**, η abstract κλάση **C** και οι κλάσεις **D** και **E**. Ποια από τα παρακάτω είναι σωστά για τον ορισμό του **F**:
 - a. `class F extends D, E { }`
 - b. `interface F implements A { }`
 - c. `class F implements A, B { }`
 - d. `class F implements B, C { }`
 - e. `interface F extends B, C { }`
 - f. `class F extends E implements A { }`
3. (Βαθμός 1) Εξηγήστε το αποτέλεσμα της εκτέλεσης του παρακάτω κώδικα.

```
public class Q1_3 {  
    public static void main(final String args[]) {  
        System.out.println(f(4));  
    }  
    public static int f(int n) {  
        if (n==0) return 1;  
    }  
}
```

```

        return n-m(f(n-1));
    }
    public static int m(int n) {
        if (n==0) return 0;
        return n-f(m(n-1));
    }
}

```

ΘΕΜΑ 2 (3 μονάδες)

1. Ταξινομήσετε σε αύξουσα σειρά τον πίνακα: «15, 11, 7, 2, 19, 13, 12» με τις μεθόδους: bubble sort, ταξινόμηση με εισαγωγή (insertion sort), ταξινόμηση με επιλογή (selection sort), ταξινόμηση με συγχώνευση (merge sort) εμφανίζοντας τους αριθμούς μετά από κάθε “πέρασμα”. **ΟΧΙ κώδικας.** (2 μον.)
2. Γράψτε στην γλώσσα προγραμματισμού Java, την αναδρομική μέθοδο `incArray()` η οποία αυξάνει κάθε θέση ενός πίνακα ακεραίων κατά μία ακέραια ποσότητα ξεκινώντας από μία συγκεκριμένη θέση η οποία δίνεται ως παράμετρος. Παράμετροι της μεθόδου: πίνακας, θέση έναρξης, ποσότητα αύξησης. Για παράδειγμα αν έχουμε τον πίνακα
`pin = { 1, 2, 3, 4, 5, 6, 7, 8 }`
και καλέσουμε την μέθοδο `incArray(pin, 3, 5)` ο πίνακας `pin` θα γίνει:
`pin = { 1, 2, 8, 9, 10, 11, 12, 13 }`

1μον.

ΘΕΜΑ 3 (4 μονάδες)

Ορίστε τη διασύνδεση (interface) με όνομα `Play`, η οποία διαθέτει τη μέθοδο `play()`.

Ορίστε την κλάση `Championship` η οποία περιγράφει μια αθλητική διοργάνωση. Η κλάση διαθέτει τα χαρακτηριστικά/ιδιότητες: όνομα αθλητικής διοργάνωσης (`cName`, `String`), πλήθος ομάδων (`teamNo`, `int`).

Η κλάση `Football` κληρονομεί/επεκτείνει την κλάση `Championship` και υλοποιεί το interface `Play`. Η κλάση διαθέτει δύο πίνακες. Τον μονοδιάστατο πίνακα ακεραίων `vathmoi`, ο οποίος περιέχει τους βαθμούς κάθε ομάδας και τον τετραγωνικό διδιάστατο πίνακα ακεραίων `results`, με τα αποτελέσματα των ομάδων. Το πλήθος τόσο των γραμμών όσο και των στηλών του πίνακα `results` είναι ίσο με το πλήθος των ομάδων. Κάθε ομάδα παίζει δύο αγώνες με όλες τις άλλες ομάδες (εντός-εκτός έδρας). Κάθε γραμμή του πίνακα `results` περιέχει τους εντός έδρας αγώνες της ομάδας και κάθε στήλη τους εκτός έδρας αγώνες της ομάδας. Κάθε κελί του πίνακα `results` περιέχει το αποτέλεσμα του αγώνα μεταξύ των δύο ομάδων που αντιστοιχούν στη γραμμή και στη στήλη του κελιού. Έτσι η τιμή 1 σε κάποιο κελί, αντιπροσωπεύει νίκη για την ομάδα που αντιστοιχεί στη γραμμή και ήττα για την ομάδα που αντιστοιχεί στη στήλη. Αντιθέτως η τιμή 2 σε κάποιο κελί αντιπροσωπεύει ήττα για την ομάδα που αντιστοιχεί στη γραμμή και νίκη για την ομάδα που αντιστοιχεί στη στήλη. Η τιμή 3 αντιπροσωπεύει ισοπαλία μεταξύ των ομάδων. Η τιμή 0 μας λέει ότι δεν υπάρχει τέτοιος αγώνας, επομένως τα κελιά με ίδιο αριθμό γραμμής και στήλης θα πρέπει να περιέχουν την τιμή 0.

Η κλάση `Football` υλοποιεί την μέθοδο `play` η οποία γεμίζει τυχαία τον πίνακα με τις τιμές 1, 2 και 3, οι οποίες αντιπροσωπεύουν τα αποτελέσματα (1: νίκη ομάδας γραμμής και ήττα ομάδας στήλης, 2 ήττα ομάδας γραμμής και νίκη ομάδας στήλης, 3 ισοπαλία). Τα κελιά με ίδιο αριθμό γραμμής και στήλης θα πρέπει να περιέχουν την τιμή 0. Σε αντίθετη περίπτωση θα πρέπει να

δημιουργείται η εξαίρεση `InvalidGameResultException` την οποία θα πρέπει να ορίσετε. Να γίνεται έλεγχος όπου χρειάζεται (πχ `set`).

Π.χ. για 5 ομάδες ο πίνακας `results` θα μπορούσε να είναι:

Ομάδες	1	2	3	4	5
1	0	1	2	3	1
2	1	0	1	2	2
3	1	3	0	2	2
4	1	3	3	0	1
5	2	3	2	3	0

Τα κελιά `[0, 2]` και `[2, 0]` δείχνουν τα αποτελέσματα των δύο αγώνων μεταξύ των ομάδων 1 και 3.

Η τιμή "2" του κελιού `[0, 2]` δείχνει ήττα της ομάδας 1 που αντιστοιχεί στη γραμμή 0 και νίκη της ομάδας 3 που αντιστοιχεί στη στήλη 2 (3 βαθμούς για την 3 και 0 βαθμούς για την 1).

Η τιμή "1" του κελιού `[2, 0]` δείχνει νίκη της ομάδας 3 που αντιστοιχεί στη γραμμή 2 και ήττα της ομάδας 1 που αντιστοιχεί

στη στήλη 0 (3 βαθμούς για την 3 και 0 βαθμούς για την 1).

Η ομάδα 3 δηλαδή είχε δύο νίκες με την ομάδα 1 και πήρε 6 βαθμούς από αυτούς τους δύο αγώνες. Στην κλάση `Football` μαζί με τις μεθόδους `get/set` για τους πίνακες (με έλεγχο εξαιρέσεων) και την μέθοδο `toString()`, ορίστε επίσης τις μεθόδους:

- **winner:** υπολογίζει και επιστρέφει τον νικητή της διοργάνωσης δηλ. την ομάδα με τους περισσότερους βαθμούς. Για τον πίνακα του παραδείγματος, νικήτρια είναι η ομάδα 4 με 16 βαθμούς.
- **vathmologia:** γεμίζει τον πίνακα `vathmoi` με τους βαθμούς κάθε ομάδας, χρησιμοποιώντας τον πίνακα `results` που έχει τα αποτελέσματα, προσθέτοντας 3 βαθμούς για κάθε νίκη, 1 για κάθε ισοπαλία και 0 για κάθε ήττα. Για τον πίνακα του παραδείγματος ο πίνακας `vathmoi` θα γίνει: 10, 9, 11, 16, 8