



Αντικειμενοστρεφής Προγραμματισμός

Εαρινό Εξάμηνο 2021-22 - Α' Εξεταστική

Εισηγητής: Αδαμίδης Παναγιώτης
Ασδρέ Κατερίνα

Τρίτη 14 Ιουνίου 2022 (11:30-13:00)

ΘΕΜΑΤΑ (Κ1)

ΠΡΟΣΟΧΗ!!! ΟΙ ΑΠΑΝΤΗΣΕΙΣ ΠΕΡΙΕΧΟΥΝ ΛΑΘΗ. ΝΑ ΕΛΕΓΧΘΟΥΝ ΚΑΛΑ.

ΘΕΜΑ 1 (3 μονάδες)

1. (Βαθμός 1) Εξηγείστε γιατί συμφωνείτε ή διαφωνείτε με τις παρακάτω προτάσεις:
 - I. Εάν το `x` είναι αναφορά αντικειμένου, η τιμή της έκφρασης "`x instanceof Object`" είναι πάντα `false`.
Λάθος γιατί το x ανήκει ΚΑΙ στην κλάση Object.
 - II. Οι λογικές (boolean) τοπικές μεταβλητές αρχικοποιούνται αυτόματα σε `true`.
Λάθος. Δεν αρχικοποιούνται αυτόματα. Πρέπει να τις αρχικοποιούμε εμείς.
 - III. Μη ελεγχόμενες εξαιρέσεις (unchecked exceptions) είναι αυτές που κληρονομούν την κλάση `Throwable`.
Σωστό. Όλες οι εξαιρέσεις κληρονομούν την Throwable.
 - IV. Εάν δεν θέλουμε να επιτρέψουμε την δημιουργία αντικειμένων κάποιας κλάσης, τότε η κλάση αυτή πρέπει να δηλωθεί `"private"`.
Λάθος. Δεν μπορούμε να δημιουργήσουμε private κλάσεις. Μόνο abstract.
2. (Βαθμός 1) Επιλέξτε ποια θεωρείτε σωστά (Σωστό: +0.5, Λάθος: -0.3)
 - I. Οι κλάσεις `Sedan`, `Suv` και `Coupe` επεκτείνουν την κλάση `Car`. Ποια από τα παρακάτω είναι σωστά:
 - a. `Sedan[] c1 = { new Sedan(), new Sedan(), new Car() };`
 - b. `Coupe[] cars = { new Car(), new Coupe(), new Sedan() };`
 - c. `Car[] car1 = new Sedan[3];`
 - d. `Suv[] c2 = { new Sedan(), new Suv(), new Coupe() };`
 - e. `Car[] car2 = { new Sedan(), new Suv(), new Coupe() };`
 - 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. `interface F implements A, B { }`
 - b. `class F extends A { }`
 - c. `class F extends B, E { }`
 - d. `class F implements B, C { }`
 - e. `interface F extends B, A { }`
 - f. `class F extends C implements B { }`

3. (Βαθμός 1) Δίνονται οι παρακάτω κλάσεις;

```
class Instrument { }  
class Strings extends Instrument { }  
class Wind extends Instrument { }
```

Εξηγήστε τι κάνουν οι παρακάτω εντολές με τη σειρά που δίνονται. Αν κάποια/ες είναι λάθος εξηγήστε:

- γιατί είναι λάθος,
- αν είναι λάθος κατά την μεταγλώττιση ή την εκτέλεση και
- αν και πως είναι δυνατό να διορθωθεί με χρήση casting.

Κάντε σχόλιο τις εντολές που δεν διορθώνονται (αφού τις κάνετε σχόλιο δεν τις παίρνετε υπόψη σας).

Πόσα αντικείμενα δημιουργούνται αρχικά (5) και πόσα έχουν απομείνει στο τέλος .

1 Instrument m = new Instrument();	Σωστή
2 Strings str1 = new Strings();	Σωστή
3 Strings str2 = new Strings();	Σωστή
4 Wind w1 = new Wind();	Σωστή
5 // Wind w2 = new Strings();	Λάθος
6 // str1 = m;	Λάθος. Αν γίνει downcasting δηλ: (Strings)m εμφανίζει Λάθος κατά τη μεταγλώττιση
7 m = str1;	Σωστή
8 // w1 = str1;	Λάθος. Δεν διορθώνεται με casting
9 // str2 = (Strings)w1;	Λάθος
10 str2 = m; Λάθος. Αν γίνει downcasting δηλ: (Strings)m εμφανίζει Λάθος κατά τη μεταγλώττιση	
11 w1 = (Wind)m;	Σωστή
12 Wind w2 = w1;	Σωστή

ΘΕΜΑ 2 (3 μονάδες)

1. Ταξινομήσετε σε αύξουσα σειρά τον πίνακα: «21, 11, 7, 2, 19, 15, 8» 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. Βάζουμε στη σειρά “n” καρέκλες. Αυτές που είναι σε μονή θέση (1^η, 3^η, 5^η κλπ) έχουν 4 πόδια και αυτές που είναι σε ζυγή θέση (2^η, 4^η, 6^η κλπ) έχουν 3 πόδια. Γράψτε στην γλώσσα προγραμματισμού Java, την αναδρομική μέθοδο chairFeet() η οποία δέχεται ως παράμετρο το πλήθος “n” των καρεκλών και υπολογίζει και επιστρέφει το πλήθος των ποδιών τους. Για παράδειγμα η chairFeet(5) επιστρέφει 18 και η chairFeet(10) επιστρέφει 35 (1 μον.)

ΘΕΜΑ 3 (4 μονάδες)

Ορίστε την κλάση **Room** η οποία περιγράφει ένα δωμάτιο ξενοδοχείου με χαρακτηριστικό το εμβαδό του δωματίου.

Ορίστε την διασύνδεση **Rentable** την οποία πρέπει να υλοποιούν τα δωμάτια που ενοικιάζονται. Υπάρχουν δύο είδη δωματίων: α) **HouseRoom** (δεν ενοικιάζονται) και β) **HotelRoom** (ενοικιάζονται).

Τα δωμάτια τα οποία ενοικιάζονται (HotelRoom) έχουν επίσης ως χαρακτηριστικό την τιμή του δωματίου και χωρίζονται σε:

α) Απλά δωμάτια (**ClassicHotelRoom**) και

β) Σουίτες (**HotelSuite**) που διαθέτουν χώρο υποδοχής. Το εμβαδό του χώρου υποδοχής αποτελεί χαρακτηριστικό της κλάσης.

Η τιμή κάθε απλού δωματίου είναι 30-120 ευρώ και η τιμή κάθε σουίτας 100-400 ευρώ. Εάν δοθεί τιμή έξω από τα όρια για το αντίστοιχο είδος δωματίου, τότε δημιουργείται η εξαίρεση **WrongRoomPrice** την οποία επίσης πρέπει να ορίσετε. Στο χειρισμό της εξαίρεσης, η τιμή του δωματίου να γίνεται η μικρότερη της αντίστοιχου είδους δωματίων και να εμφανίζεται το μήνυμα "Room price is set to minimum".

Ορίστε την κλάση **Hotel** η οποία περιγράφει ένα ξενοδοχείο και διαθέτει τα χαρακτηριστικά:

α) Όνομα ξενοδοχείου (String) και

β) Πίνακα δωματίων. Ο πίνακας είναι δύο διαστάσεων. Κάθε γραμμή αντιπροσωπεύει τα δωμάτια κάθε ορόφου. Η γραμμή 0 αντιστοιχεί στο ισόγειο.

Στον default constructor της **Hotel**, ορίστε ένα ξενοδοχείο με ισόγειο και 5 ορόφους (0-5). Κάθε όροφος έχει 20 δωμάτια, επομένως υπάρχουν συνολικά 120 δωμάτια. Κάθε δωμάτιο μπορεί να είναι ενοικιαζόμενο (πιθανότητα 90%) ή να μην νοικιάζεται (πιθανότητα 10%). Από τα δωμάτια που ενοικιάζονται το 80% είναι απλά δωμάτια (**ClassicHotelRoom**) και το 20% σουίτες (**HotelSuite**).

Η κλάση **Hotel** εκτός των set/get, διαθέτει και τις μεθόδους:

- **dailyFullRentAmount:** υπολογίζει και επιστρέφει το συνολικό ποσό είσπραξης από την ενοικίαση **όλων** των δωματίων του ξενοδοχείου τα οποία είναι δυνατό να ενοικιαστούν.
- **totalSuiteHallSpace:** υπολογίζει και επιστρέφει το συνολικό χώρο υποδοχής που διαθέτουν όλες οι σουίτες του ξενοδοχείου.
- **totalRoomArea:** υπολογίζει και επιστρέφει το συνολικό εμβαδόν όλων των δωματίων του ξενοδοχείου.

Υποδείξεις:

Όλα τα χαρακτηριστικά να είναι private. Ορίστε δομητές και μεθόδους όπου χρειάζεστε. Μέθοδοι πρόσβασης/μεταβολής (get/set) τουλάχιστον για τον πίνακα δωματίων (ολόκληρο, τις γραμμές και κάθε κελί του).

Υπερβείτε την μέθοδο toString της Object.