

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ
ΠΕΡΙΟΔΟΣ ΙΟΥΝΙΟΥ-ΣΕΠΤΕΜΒΡΙΟΥ 2002 •

ΕΙΣΑΓΩΓΗ ΣΤΟ ΛΟΓΙΣΜΙΚΟ ΙΙ (JAVA)

ΘΕΜΑ 1^ο (21)

Δίνονται οι εξής έννοιες/οντότητες: animal (ζώο), pet (κατοικίδιο), big-anim (μεγάλο ζώο τσίρκου), small-anim (μικρό ζώο τσίρκου), circus-anim (ζώο τσίρκου), mickey (Μίκυ, που είναι μικρό ζώο τσίρκου), dumbo (Ντάμπυ, που είναι μεγάλο ζώο τσίρκου), pluto (Πλούτο, που είναι κατοικίδιο).

- α) Προσδιορίστε ποιές αντιπροσωπεύουν κλάσεις και ποιές στιγμιότυπα. Ποιό είναι το κριτήριο διάκρισης; *class, object, instance*
- β) Προσδιορίστε τις υπερκλάσεις της small-anim και τις υποκλάσεις της circus-anim. *small anim, circus anim*
- γ) Ποιά έννοια είναι γενίκευση και ποιά εξιδεικευση της circus-anim; Εξηγήστε. *small anim, circus anim*
- δ) Σχεδιάστε την αντίστοιχη ιεραρχία των παραπάνω εννοιών/οντοτήτων.
- ε) Γράψτε γενικό κώδικα Java που να υλοποιεί την ιεραρχία αυτή.
- στ) Εξηγήστε τι είναι απλή κληρονομικότητα. Δώστε ένα παράδειγμα από τη ιεραρχία.
- ζ) Εξηγήστε τι είναι πολλαπλή κληρονομικότητα. Δώστε ένα παράδειγμα τροποποιώντας την ιεραρχία που σχεδιάσατε. Πώς υλοποιείται γενικά η πολλαπλή κληρονομικότητα στη Java; *class, interface*

ΘΕΜΑ 2^ο (24)

Απαντήστε ποιό από τους παρακάτω ισχυρισμούς είναι σωστό και ποιό λάθος (χωρίς εξήγηση).

- Σ (1) Η διεπαφή (interface) είναι μια κλάση χωρίς στιγμιότυπα. Σ
- Λ (2) Μια μέθοδος δεν μπορεί να είναι αναδρομική. Λ
- Σ (3) Στον χειρισμό εξαιρέσεων δεν μπορεί να υπάρξει ένα catch μπλοκ χωρίς αντίστοιχο try μπλοκ. Σ
- Σ (4) Ένα στιγμιότυπο μπορεί να δημιουργηθεί μόνο με τη χρήση της new. Σ
- Σ (5) Η μη αλήθεια της c1 == c2, σημαίνει ότι τα στιγμιότυπα στα οποία αναφέρονται τα c1, c2 δεν είναι ίδια μεταξύ τους. Σ
- Λ (6) Όλες οι τοπικές μεταβλητές σε μια μέθοδο πρέπει να δηλώνονται private. Λ
- Λ (7) Όλοι οι μέθοδοι μιας κλάσης πρέπει να δηλώνονται ως public. Λ
- Λ (8) Μια μεταβλητή που δηλώνεται ως protected είναι προσπελάσιμη μόνο από μεθόδους της κλάσης που ανήκει. Λ
- Σ (9) Μια μέθοδος δεν είναι υποχρεωτικό να έχει παραμέτρους (ορίσματα). Σ
- Λ (10) Σε μια κλάση δεν μπορεί να υπάρχουν περισσότεροι από ένα δημιουργοί. Λ
- Λ (11) Μια αφηρημένη (abstract) κλάση περιέχει μόνο αφηρημένες μεθόδους. Λ
- Λ (12) Για το ίδιο try μπλοκ δεν μπορεί να υπάρχουν περισσότερα του ενός catch μπλοκ. Λ
- Σ (13) Αν δεν εγερθεί εξαίρεση ένα catch μπλοκ δεν παίζει κανένα ρόλο. Σ
- Λ (14) Μια μέθοδος δεν μπορεί να επιστρέφει ένα στιγμιότυπο σαν τιμή. Λ
- Σ (15) Μια μεταβλητή μέλους μπορεί να αντιπροσωπεύει είτε μια πρωτογενή τιμή είτε ένα αντικείμενο. Σ
- Σ (16) Μια μέθοδος δεν χρειάζεται πάντα να επιστρέφει μια τιμή. Σ

ΘΕΜΑ 3^ο (20)

- α) Γράψτε μια μέθοδο (charCount) Java με ορίσματα s και c (όπου s είναι ένα αλφαριθμητικό και c ένας χαρακτήρας), που επιστρέφει τον αριθμό των εμφανίσεων του c στο s. [8]
- β) Γράψτε ένα πρόγραμμα Java που να μετρά και να εκτυπώνει τον αριθμό εμφανίσεων ενός χαρακτήρα, που είναι το τελευταίο όρισμα, στα υπόλοιπα ορίσματα που δίνονται από τη γραμμή εντολών, χρησιμοποιώντας και την μέθοδο στο (α). Η εκτύπωση θα έχει τη μορφή:

«Ο χαρακτήρας x εμφανίζεται n φορές», όπου x ο εκάστοτε χαρακτήρας και n ο αριθμός εμφανίσεών του στα υπόλοιπα ορίσματα. [12]

ΘΕΜΑ 5° (6)

Θεωρήστε τη συνάρτηση funx, όπου $y, z > 0$:

```
private int funx (int y, int z) {
    while (y != z) {
        while (y > z)
            y -= z;
        while (y < z)
            z -= y;
    }
    return y;
}
```

$y = y - z$
 $z = z - y$

Ποια είναι η τιμή της έκφρασης funx(16, 29); Εξηγήστε με τη βοήθεια πίνακα τιμών.

ΘΕΜΑ 4° (29)

Δίνεται ο παρακάτω κώδικας Java. α) Αφού εξηγήσετε τι είναι δημιουργός (constructor), βρείτε ποιές από τις κλάσεις του κώδικα έχουν δημιουργούς. Σε τι διαφέρουν αυτοί οι δημιουργοί μεταξύ τους; Τι πρόβλημα μπορεί να υπάρξει με τις κλάσεις που δεν έχουν δημιουργό [6]; β) Εξηγήστε συνοπτικά ποιά είναι το αποτέλεσμα της κάθε γραμμής κώδικα της κλάσης 'RunWindow' [15], γ) Τι ρόλο παίζει το 'this' στην πρόταση 'this.size = x', είναι απαραίτητο [2]; δ) Αν η size στην Window δηλωνόταν private, θα δημιουργούσε πρόβλημα [3]; ε) Αν η setSize στην StandSize δεν δηλωνόταν ως static θα δημιουργούσε πρόβλημα [3];

printSize1:
για το περικό μεταβλητών που έχει ορίσει στη κλάση Window

printSize2:
για το περικό μεταβλητών που έχει ορίσει στην κλάση Window

setSize1(3):
η συνάρτηση size της κλάσης Window θα αυξήσει την τιμή της size κατά 3

setSize2(2):
η συνάρτηση size της κλάσης Window θα αυξήσει την τιμή της size κατά 2

printSize2():
για το περικό μεταβλητών που έχει ορίσει στη κλάση Window

setSize(5):
αρχικά είναι συνάρτηση size2 της κλάσης Window η οποία θα αυξήσει την τιμή της size κατά 5

```
public class Window
{ protected int size = 1;
  public void setSize(int x)
  { size = x; }
  public void printSize1()
  { System.out.println (size); } }

public class XWindow extends Window
{ private int size = 0;
  public static z = 0;
  public XWindow () {}
  public XWindow (int x)
  { this.size = x; }
  public void setSize1(int x)
  { size += x; }
  public void setSize2(int x)
  { super.setSize(x); }
  public void setZ (int newz)
  { z = newz; }
  public void printSize1()
  { System.out.println (size); }
  public void printSize2()
  { System.out.println (super.size); } }
```

```
public class StandSize
{ public static int size = 3;
  public static void setSize (x)
  { x = size; } }

public class RunWindow
{ public static void main ()
{
  int x = 1;
  XWindow w1 = new XWindow (4);
  XWindow w2 = new XWindow (6);
  StandSize.setSize (x);
  w2.setZ (2);
  System.out.println (w1.z);
  System.out.println (w2.z);
  w2.printSize1 ();
  w2.printSize2 ();
  w2.setSize1 (3);
  w2.setSize2 (2);
  w2.printSize2 ();
  w1.setSize2 (5);
  w1.printSize1 ();
}
```

θα έπρεπε να ορίσει static για να χρησιμοποιήσει.

στατική συνάρτηση.

καλείται η συνάρτηση constructor της κλάσης Window που δε φέρει ορίσει.

καλείται ο 2ος constructor της κλάσης Window που δε φέρει ορίσει.

καλείται η συνάρτηση printSize1 της κλάσης Window που δε φέρει ορίσει.

καλείται η συνάρτηση printSize2 της κλάσης Window που δε φέρει ορίσει.

καλείται η συνάρτηση setSize της κλάσης StandSize που δε φέρει ορίσει.

καλείται η συνάρτηση setSize2 της κλάσης Window που δε φέρει ορίσει.

καλείται η συνάρτηση printSize1 της κλάσης Window που δε φέρει ορίσει.

καλείται η συνάρτηση printSize2 της κλάσης Window που δε φέρει ορίσει.

ΚΑΛΗ ΕΠΙΤΥΧΙΑ!

τυπώνουν στην οθόνη την τιμή της στατικής μεταβλητής z. (1 z → static) & στις 2 εντολές θα τυπωθεί η ίδια αντανάκλαση διότι το z έχει οριστεί ως στατική μεταβλητή της κλάσης Window, που σημαίνει ότι είναι η ίδια μεταβλητή