Programación en Lenguaje Java

Problema 11.2. Cuestiones: referencias, objetos y excepciones

Michael González Harbour

Mario Aldea Rivas

Departamento de Matemáticas,
Estadística y Computación

Problema 11.2. Cuestiones: referencias, objetos y excepciones

Objetivos

- Repasar los conceptos de referencias/objetos y excepciones estudiados durante el curso.

Cuestión 1. Referencias y objetos

Se dispone de la clase mostrada a continuación:

```
public class Cosa {
private int num;
private static int cont;

public Cosa(int num) {
this.num = num;
}

public int getNum() {
return num;
}

public void setNum(int num) {
this.num = num;
cont++;
}

public static int getCont() {
return cont;
}
}
```

Responde las cuestiones relativas a cada uno de los siguientes programas:

1.a)

```
1   public static void main(String[] args) {
2     ArrayList<Cosa> cosas = new ArrayList<Cosa>();
3     int cont;
4
5     Cosa c1 = new Cosa(1);
6
7     cosas.add(new Cosa(2));
8     cosas.add(c1);
9     cosas.add(c1);
10
11    cont = c1.getCont();
12    System.out.println("cont:" + cont);
13    for(Cosa c: cosas) {
14      System.out.println(c.getNum());
15    }
```

```
16
17    cosas.get(2).setNum(3);
18    cont = c1.getCont();
19    System.out.println("cont:" + cont);
20    for(Cosa c: cosas) {
21      System.out.println(c.getNum());
22    }
23
24    System.out.println("c1.num:" + c1.getNum());
25  }
```

- Indica la salida por consola que produciría la ejecución del programa.
- Las líneas 11 y 17 sería más correcto haberlas escrito de otra manera ¿cómo?

1.b)

```
1   public static void main(String[] args) {
2     Cosa c1 = new Cosa(1);
3
4     metodo(c1, 2);
5     System.out.println("c1.num:" + c1.getNum());
6
7
8     Cosa c2 = new Cosa(3);
9     c1 = c2;
10    metodo(c2, 4);
11    System.out.println("c1.num:" + c1.getNum());
12    System.out.println("c2.num:" + c2.getNum());
13  }
14
15  private static void metodo(Cosa c, int n) {
16    c.setNum(n);
17    c = null;
18  }
```

- Indica la salida por consola que produciría la ejecución del programa.
- Indica los objetos que se crean y en qué línea se produce la creación.
- Indica las líneas en las que un objeto se convierte en basura y di cual es ese objeto.

Cuestión 2 (Examen junio 2014)

Se dispone de la clase `Red`, que representa una red de computadores:

```
public class Red {
  public static class ErrorEnvio extends Exception {}

  /**
   * Envía un mensaje al destino indicado.
   * @param destino dirección del computador destino
   * @param mensaje mensaje a enviar
   * @throws ErrorEnvio si no es posible enviar el mensaje
   */
  public void enviaMensaje(int direccionDestino, String mensaje)
                          throws ErrorEnvio {
    ... código no relevante para el problema planteado
  }

  ... otros métodos no relevantes para el problema planteado
}
```

Se pide escribir el código de un método de la misma clase que llame a `enviaMensaje` y, utilizando el patrón de excepciones recuperables, verifique el comportamiento descrito por el siguiente comentario de documentación:

```
/**
 * Envía uno a uno los mensajes contenidos en un array de mensajes.
 * El envío de los mensajes restantes se aborta en el momento que se
 * supere el número máximo de errores permitidos al intentar enviar
 * uno de los mensajes.
 * @param destino dirección del computador destino
 * @param mensajes array con los mensajes a enviar
 * @param erroresPermitidos máximo número de errores permitidos
 *        para un mensaje
 * @throws ErrorEnvio si se supera el máximo número de errores
 *         permitidos para un mensaje
 */
```