

Primer Certamen

~~Todas las preguntas tienen igual puntaje.~~

25 pts.

1.- En la documentación de la clase Math se tiene:

static double random() Returns a double value with a positive sign, greater than or equal to 0.0 and less than 1.0.

static long round(double a) Returns the closest long to the argument.

Haga un programa que pida por consola hacer la suma de dos valores aleatorios positivos menor o igual que 100, luego lea el resultado ingresado por el usuario y envíe un mensaje indicando si fue correcto o no y el tiempo que demoró el usuario en ingresar la respuesta.

```
import java.util.*;
import java.io.*;
```

```
public class SumaConTiempo
{
    public static void main(String args[]) throws IOException // 3pts por throws ....
    {
        Scanner s = new Scanner (System.in); // 4pts
        long a = Math.round(Math.random()*100);
        long b = Math.round(Math.random()*100); // 4 pts
        long t_final;
        long t_inicial = new Date().getTime(); // 3 pts
        System.out.print(a + " + " + b + " = ");
        int respuesta = s.nextInt(); // 4 pts
        t_final = new Date().getTime();
        if (respuesta == a+b) // 4 pts por if completo
            System.out.print("Respuesta correcta en ");
        else
            System.out.print("Respuesta errada en ");
        System.out.println((t_final-t_inicial) + " milisegundos"); // 3 pts.
    }
}
```

30 pts.

2.- 5 pts cada subpregunta.

a) ¿Para qué se usa el operador >>> en Java?

Se usa para hacer desplazamientos hacia la derecha ingresando el bit de signo por la izquierda. >> hace algo similar pero ingresando ceros.

b) ¿Para qué se usa la variable de ambiente CLASSPATH en Java?

Esta variable se usa para señalar al compilado o a la máquina virtual los directorios donde se encuentran los archivos de las clases usadas por aquel en compilación o ejecución.

c) ¿Cómo se configura esta variable en Linux?

En la consola de interés podemos:

export CLASSPATH= <seguido del listado de clases separados por ;>

d) Podemos decir que un stack es una lista cuyas operaciones de inserción y eliminación se limitan sólo a uno de sus extremos. Del lenguaje natural se lee "un stack es una lista", ¿podría desprender de esto una relación de herencia entre un stack y una lista? Justifique.

No, el stack no debería heredar de lista debido a que no cumpliría el principio de sustitución. Si aceptamos que stack puede heredar de lista, donde se requiere una lista podríamos pasar una referencia a stack. Si esperamos insertar un elemento al final de la lista, el objeto stack pasado no lo podría hacer y generaría un resultado distinto a lo deseado.

esta situación es similar a la explicación vista en clases a por qué un cuadrado no puede ser visto como un rectángulo de lados iguales.

e) ¿En qué caso no podemos relanzar una excepción y ésta debe ser manejada por el método?
Cuando deseamos re-definir un método de la clase base cuyo prototipo no lanza excepciones.

f) ¿Tiene sentido definir un método privado? ¿Por qué no, o en qué caso se justifica?

Sí tiene sentido. Esto se justifica cuando el implementador de una clase ve la conveniencia de definir nuevos métodos para facilitar o estructurar de mejor forma la implementación de los métodos públicos o en general los accesibles desde fuera de la clase.

15 pts.

3.- La ejecución del programa mostrado es la que se indica. Explique por qué el mensaje "Ingrese un número" aparece dos veces. Indique cómo corregir este problema.

```
public class InputErrorExample
{
    public static void main(String args[]) throws IOException
    {
        Scanner s = new Scanner (System.in);
        System.out.print("Ingrese su estatura en metros: ");
        float height = s.nextFloat();
        System.out.print("Ingrese su peso en kg: ");
        while(!s.hasNextFloat()){
            s.nextLine();
            System.out.println("Ingrese un número real por favor:");
        }
        float weight = s.nextFloat();
        float imc = weight/(height*height);
        System.out.println("Su índice de masa corporal es" +imc+".");
    }
}
```

Resultado de ejecución:

```
Ingrese su estatura en metros: 1,83
Ingrese su peso en kg: ab
Ingrese un número real por favor:
Ingrese un número real por favor:
90
Su índice de masa corporal es 26.874495.
```

10 pts. Al ingresar la estatura el método nextFloat() lee los dígitos ingresados hasta encontrar un carácter no dígito. Éste será el retorno de carro (enter). Por ello al ingresar el peso en formato errado el primer nextLine() lee el retorno de carro después de al estatura. La segunda vez recién lee el dato con formato errado.

5 pts Una solución es agregar cambiar:

```
:
System.out.print("Ingrese su estatura en metros: ");
float height = s.nextFloat();
System.out.print("Ingrese su peso en kg: ");
por
:
System.out.print("Ingrese su estatura en metros: ");
float height = s.nextFloat();
s.nexLine(); // con esto se lee el retorno de carro ingresado.
System.out.print("Ingrese su peso en kg: ");
```

30 pts.

4.- Haga un programa en Java que muestre un botón con nombre "Crear botón". Cuando éste es presionado, se incorpora un segundo botón igual al primero. Cuando se presiona cualquiera de los dos, se crea un tercer botón igual a los anteriores. Este comportamiento se repite para todo botón creado. La aplicación termina cuando presionamos el botón de cierre de la esquina superior derecha de la ventana. ¿Con qué comando compila y luego ejecuta su programa en una consola de aragorn?

4 pts por estructura general de la solución

En archivo CreaBotones.java tenemos:

```
import java.awt.*;
import java.awt.event.*;
import java.util.*;
import javax.swing.*;
public class CreaBotones
{
    public static void main(String[] args)
    {
        CreaBotonesFrame frame = new CreaBotonesFrame();
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); // 4 pts
        frame.setVisible(true);
    }
}
class CreaBotonesFrame extends JFrame // 4 pts.
{
    public CreaBotonesFrame()
    {
        setTitle("Crea Botones");
        setSize(250, 150);
        CreaBotonesPanel panel = new CreaBotonesPanel(); // 5 pts por incorporación de panel
        Container contentPane = getContentPane(); //
        contentPane.add(panel); //
    }
}
class CreaBotonesPanel extends JPanel
{
    public CreaBotonesPanel()
    {
        creaBoton();
    }
    private void creaBoton()
    {
        JButton boton = new JButton("Crea Botón"); // 8 pts por actino listener adecuado
        boton.addActionListener( new ActionListener (){
            public void actionPerformed(ActionEvent event) {
                creaBoton();
            };
        });
        add(boton);
        validate(); // debe ir, pero no descuenta puntos en certamen.
    }
}
}
```

5 Pts. Para Compilar:

```
$ javac CreaBotones.java
```

Para ejecutar:

```
$ java CreaBotones
```