

Pauta Certamen 1, Programación II

Prof. Rodrigo Olivares
Ayud. Juan Carlos Tapia
Septiembre 22, 2015

Instrucciones:

- El puntaje máximo del certamen es 100%, siendo el 60% el mínimo requerido para aprobar.
- Responda cada pregunta en la hoja indicada, agregando su nombre. Si no responde alguna pregunta, debe entregar la hoja con su nombre e indicar que **no responde**.
- El certamen es **individual**. Cualquier intento de copia, será sancionado con nota **1,0**.

1. *30pts.* De las siguientes afirmaciones, encierre en un círculo la o las alternativas correctas.

- | | |
|--|---|
| i. Algunos enfoques de la orientación a objeto son: | (b) La instancia de una clase. |
| (a) El enfoque reusable. | (c) Una abstracción del mundo real. |
| (b) El enfoque abstracto. | (d) Un sub-conjunto de atributos y métodos de la clase. |
| (c) El enfoque imperativo. | (e) Siempre estático. |
| (d) El enfoque procedural. | |
| (e) Ninguna de las anteriores. | vii. El principio de ocultamiento: |
| ii. En cuanto a la programación orientada a objeto: | (a) Es una técnica que protege el estado de una entidad. |
| (a) Se apoya en el paradigma estructural. | (b) Es indispensable en el paradigma de orientación a objeto. |
| (b) Divide el programa en pequeñas unidades de código. | (c) En Java, se logra utilizando los modificadores de acceso. |
| (c) Proporciona herramientas para modelar el mundo real. | (d) Es encapsular el conocimiento de una entidad. |
| (d) Es una herramienta del lenguaje JAVA. | (e) Ninguna de las anteriores. |
| (e) Ninguna de las anteriores. | |
| iii. Una clase es: | viii. Respecto a la herencia, las clases: |
| (a) Un arreglo de objetos. | (a) Heredan sólo los métodos privados. |
| (b) Una abstracción del mundo real. | (b) Heredan el comportamiento completo de la clase padre. |
| (c) Una herramienta de programación. | (c) Heredan sólo el comportamiento que se desea utilizar. |
| (d) Un punto a memoria. | (d) En Java, se implementan con la palabra implements. |
| (e) Ninguna de las anteriores. | (e) En Java, se implementan con la palabra extends. |
| iv. Respecto a una clase: | ix. El polimorfismo: |
| (a) Se declaran utilizando la palabra reservada clase. | (a) El mismo nombre implementa distintas funcionalidades. |
| (b) Debe tener el mismo nombre que el archivo. | (b) Una funcionalidad implementada con distintos nombres. |
| (c) Son declaradas estáticas con static. | (c) Es una característica de JAVA. |
| (d) Todas deben incluir el método main. | (d) Es una característica de POO. |
| (e) Ninguna de las anteriores. | (e) Un ejemplo es el símbolo %. |
| v. Respecto a una clase: | x. El método <i>main</i> : |
| (a) Puede o no tener atributos. | (a) Puede no ser void. |
| (b) Puede o no tener métodos. | (b) Debe ser static. |
| (c) Puede o no tener constructor. | (c) Puede no llevar argumentos de entrada. |
| (d) Puede o no tener un nombre. | (d) Debe retornar un valor. |
| (e) Puede o no tener un tipo. | (e) Debe incluirse en un programa. |
| vi. Un objeto es: | |
| (a) Un tipo de dato de la clase. | |

2. 30pts. Desarrolle una clase Calculadora que permita gestionar las operaciones elementales: suma, resta, multiplicación, división y módulo. Recuerde utilizar los principios de orientación a objetos.

```
public class Calculadora {

    public double numero;

    public double suma(double num) {
        return numero + num;
    }

    public double resta(double num) {
        return numero - num;
    }

    public double multiplicacion(double num) {
        return numero * num;
    }

    public double division(double num) {
        if (num == 0) {
            return Double.NaN;
        } else {
            return numero / num;
        }
    }

    public double modulo(double num) {
        if (num == 0) {
            return Double.NaN;
        } else {
            return numero % num;
        }
    }

    public static void main(String[] args) {
        Calculadora calc = new Calculadora();

        calc.numero = 50;
        double num = 10;

        System.out.println(calc.numero + " + " + num + " = " + calc.suma(num));
        System.out.println(calc.numero + " - " + num + " = " + calc.resta(num));
        System.out.println(calc.numero + " * " + num + " = " + calc.multiplicacion(num));
        System.out.println(calc.numero + " / " + num + " = " + calc.division(num));
        System.out.println(calc.numero + " % " + num + " = " + calc.modulo(num));
    }
}
```

3. *40pts.* Un palíndromo es una palabra, número o frase que puede ser leída de igual forma hacia adelante y hacia atrás, por ejemplo:

“anita lava la tina”.

“1234567890987654321”.

“la ruta nos aportó otro paso natural”.

Desarrolle una clase (en JAVA) que posea los métodos necesarios para determinar si una palabra es o no un palíndromo.

```
public class Palindrome {

    public String frase;

    private boolean isPalindrome() {
        String fraseLimpia = "";
        String fraseInvertida = "";
        if (frase != null) {
            fraseLimpia = frase.replace(" ", "");
            for (int i = 0; i < fraseLimpia.length(); i++) {
                fraseInvertida = fraseLimpia.charAt(i) + fraseInvertida;
            }
        }
        return fraseLimpia.equalsIgnoreCase(fraseInvertida);
    }

    public static void main(String[] args) {
        Palindrome pal = new Palindrome();

        pal.frase = "Anita lava la tina";

        if (pal.isPalindrome()) {
            System.out.println("La frase: \"" + pal.frase + "\" es palindrome");
        } else {
            System.out.println("La frase: \"" + pal.frase + "\" no es palindrome");
        }
    }
}
```