

Laboratorio de Fundamentos de Programación

Hoja de Ejercicios 3

Deben probarse todos los métodos que se piden con un programa que los utilice.

1. Escribir un programa Java que pida al usuario un número positivo y escriba en pantalla sus raíces cuadrada, cúbica, cuarta, y quinta.
2. Escribir un programa que, cuando se ejecute, escriba en pantalla un número entero entre 1 y 6 (como la tirada de un dado). Cada vez que se ejecute podrá dar un resultado distinto (use la función `Math.random()`).
3. Escribir un método Java que tome como parámetros 3 números enteros, para día, mes y año (supuestamente correctos), y escriba en pantalla la fecha en formato *dia/mes/año*. Escriba otro método que lo escriba en forma *mes año, día* (al estilo EEUU y algunos otros países). Utilice estos métodos para escribir un programa que pida al usuario 3 números enteros para una fecha, y escriba en pantalla la fecha en los dos formatos.
4. Escribir un método Java que tome como parámetros una cadena de caracteres para el día de la semana, y 3 números enteros, para día, mes y año (supuestamente correctos), y escriba en pantalla la fecha en formato largo, como por ejemplo *"Domingo, 31 del 12 de 2023"*.
5. Escribir el método Java necesario para completar el programa siguiente:

```
import java.util.Locale;
import java.util.Scanner;

public class H3E3 {
    public static void main (String[] args){
        /* De polares a cartesianas, dim 2*/
        Scanner in = new Scanner (System.in);
        in.useLocale (Locale.US);
        double x, y; double angulo, radio;

        System.out.print ("Escriba un ángulo (en radianes): ");
        angulo = in.nextDouble();
        System.out.print ("y un radio (positivo): ");
        radio = in.nextDouble();

        System.out.println ("Ángulo " + angulo + " , radio " + radio);
        System.out.print ("El punto en cartesianas es ");
        EscribirEnCartesianasLasPolares (radio, angulo); // SENTENCIA
        System.out.println();
    }
}
```

La ejecución del programa mostrará un aspecto como el siguiente, para la entrada $(\approx)\pi/4$, 1:

```
Escriba un ángulo (en radianes): 0.785398
y un radio (positivo): 1
Ángulo 0.785398 , radio 1.0
El punto en cartesianas es (0.7071068967259818, 0.7071066656470943)
```

6. a) Considere la siguiente clase Java, que consta de 4 métodos:

```
public class H3E8 {  
  
    public static void zoop () {  
        baffle ();  
        System.out.print ("You wugga ");  
        baffle ();  
    }  
  
    public static void main (String[] args) {  
        System.out.print ("No, I ");  
        zoop ();  
        System.out.print ("I ");  
        baffle ();  
    }  
  
    public static void baffle () {  
        System.out.print ("You wug ");  
        ping ();  
    }  
  
    public static void ping () {  
        System.out.println (".");  
    }  
}
```

Los métodos no están documentados. Complete la siguiente "documentación":

efecto del método ping: *escribe un punto*

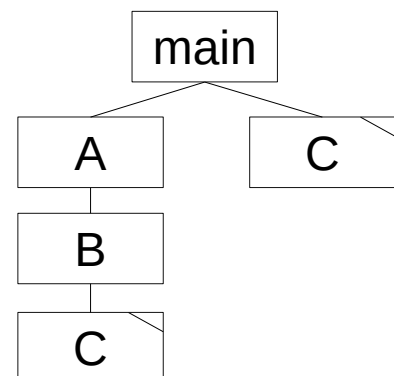
efecto del método baffle: *escribe "You wug."*

efecto del método zoop: _____

efecto del programa: _____

b) Una forma de representar la relación de llamadas es un diagrama modular como el del siguiente ejemplo:

```
public class Ejemplo {  
  
    public static void A () {  
        ... // código  
        B();  
    }  
  
    public static void main (String[] args) {  
        A();  
        ...  
        A();  
        C();  
    }  
  
    public static void B () {  
        ...  
        C();  
    }  
  
    public static void C() { ... }  
}
```



Como se puede ver, el orden de definición de los métodos no tiene por qué tener mucha relación con el orden en que se ejecutan, ni con el número de veces que son utilizados (gracias a la forma de traducción de los programas Java). Con la misma idea del ejemplo, elabore el diagrama modular del programa del apartado a).

7. Escribir funciones para transformar grados en radianes y viceversa
8. Escribir el método Java necesario para completar el programa siguiente:

```
import java.util.Scanner;
import java.util.Locale;
public class H3E4 {
    public static void main (String[] args){
        /* De cartesianas a polares, dim 2 */
        Scanner in = new Scanner (System.in);
        in.useLocale(Locale.US);
        double x, y;

        System.out.print ("Escriba las 2 coordenadas de un punto: ");
        x = in.nextDouble(); y = in.nextDouble();

        System.out.println ("Punto (" + x + ", " + y + ")");
        System.out.print ("En polares es: ");
        System.out.println ( "Radio= " + rad(x,y) +
                             ", ángulo = " + ang (x,y)); // EXPRESIONES
    }
}
```

La ejecución del programa mostrará un aspecto como el siguiente, para la entrada 1 0 :

```
Escriba las 2 coordenadas de un punto: 1 0
Punto (1.0, 0.0)
En polares es: Radio= 1.0, ángulo = 0.0
```

9. Escribir un programa Java que pida al usuario 3 números enteros, para día, mes y año (supuestamente correctos), y un número de opción, 1 ó 0. El programa debe entonces escribir en pantalla la fecha en formato *día/mes/año* ó *mes día, año*, según la opción hay sido 1 ó 0, y usando los métodos del ejercicio 3.
10. Escribir una función Java que devuelva el menor de tres números que recibe como parámetros.
11. Escribir un método Java que tome tres valores positivos como parámetros y devuelva True o False según sea posible o no formar un triángulo con segmentos de esos valores. (Por ejemplo, si los valores son 12, 1, 2, no se podrá formar un triángulo con lados de esas longitudes). Para que pueda formarse, cada una de las longitudes tiene que ser menor que la suma de las otras dos.
12. Escribir un programa Java que lea tres valores positivos y escriba en pantalla el área de un triángulo cuyos lados tengan esas longitudes, o 0 si no es posible formarlo.
13. Escribir un método Java que tome como parámetros una cadena de caracteres para el día de la semana, y 3 números enteros, para día, mes y año (supuestamente correctos), y **escriba en pantalla** la fecha en formato largo, con el nombre del mes, como por ejemplo *"Domingo, 31 de diciembre de 2023"*.
14. Escribir un método Java que tome como parámetros una cadena de caracteres para el día de la semana, y 3 números enteros, para día, mes y año (supuestamente correctos), y **devuelva** la cadena de caracteres que contiene la fecha en formato largo con el nombre del mes, como por ejemplo *"Domingo, 31 de diciembre de 2023"*.