

**PRUEBA 1 PROGRAMACIÓN  
Marzo 2007  
INGENIERÍA INFORMÁTICA**

LEA ATENTAMENTE ESTAS INSTRUCCIONES ANTES DE COMENZAR LA PRUEBA:

- Rellene todas las hojas a bolígrafo, tanto los datos personales como las respuestas
- No olvide rellenar el NIA y el grupo real al que pertenece.
- El tiempo máximo de realización es de 50 minutos
- El único material permitido sobre la mesa es la hoja de test y un bolígrafo

NO PASE DE ESTA HOJA, hasta que se le indique

<i>Apellidos</i>	<i>Nombre</i>	
<i>Firma</i>	<i>NIA</i>	<i>Grupo</i>

**PARTE 1: CUESTIONES**

**Pregunta 1 (1 Punto).**- Indicar si la siguiente afirmación es cierta, y explicar brevemente por qué.

*"Un constructor de una clase puede utilizar todos los métodos de esa clase y un método puede utilizar un constructor utilizando la palabra "this" cuando se quiere hacer referencia a él."*

Falso. Un constructor puede utilizar todos los métodos de una clase, pero los métodos no pueden usar ningún constructor.

"this" se usa para hacer referencia a un constructor dentro de otro constructor, pero no dentro de un método cualquiera.

---

**Pregunta 2 (1 Punto).**- Indicar si la siguiente afirmación es cierta, y explicar brevemente por qué.

*"En Java, todos los métodos pueden usar los atributos de la clase donde se encuentren. Asimismo, las variables de un método pueden ser usadas por todos los métodos de esa clase."*

Falso. Todos los métodos de una clase pueden usar los atributos de esa clase, pero no ocurre lo mismo con las variables de un método, que sólo se pueden usar dentro del método en que han sido definidas.

---

**Pregunta 3 (1 Punto).**- Indicar si la siguiente afirmación es cierta, y explicar brevemente por qué.

*"Java es un lenguaje orientado a objetos puro ya que todo en Java son objetos"*

Falso, Java no es un lenguaje orientado a objetos puro, ya que los tipos básicos no son objetos.

**Pregunta 4 (1 Punto).**- Dado el siguiente código explicar qué es lo que hace el método mostrado en la clase Pregunta2 y cuál es la salida por pantalla.

```
public class Pregunta2{
    int metodo(int[] vector){

        int salida = 0;
        for (int i=0; i<vector.length; i++){
            if (i%2==0) salida = salida + vector[i];
            else salida = salida - vector[i+1];
        }
        return salida;
    }
}

public class Ejecutar2{
    public static void main(String[] args){

        int[] lista1= {2,3,3,4,3};
        int[] lista2= {2,5,6,2};
        Pregunta2 p2 = new Pregunta2();
        System.out.println(p2.metodo(lista1));
        System.out.println(p2.metodo(lista2));
    }
}
```

Solución: Si el tamaño del array es impar la salida se corresponde con el primer elemento de la lista pues va sumando y restando sucesivamente el resto de posiciones pares. Si el tamaño del array es par se comete un error de ejecución al acceder a una posición no existente en la última vuelta del for.

Salida:

2

Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 4  
at Pregunta2.metodo(Pregunta2.java:8)  
at Ejecutar2.main(Ejecutar2.java:9)

---

**Pregunta 5 (1 Punto).**- Escribir un método que reciba un *array* de números reales (*float*) y devuelva un *array* de enteros truncando la parte real de los números originales. Ejemplo: recibe {1.3F, 2.6F, 4.3F, 3.9F} y devuelve {1,2,4,3}

```
public int [] trunca (float a []){
    int [] resultado = new int [a.length];
    for (int i=0; i<a.length; i++)
        resultado[i]=(int)a[i];
    return resultado;
}
```

## **PARTE 2: PROBLEMAS**

**Problema 1 (3 Puntos).**- Dado el siguiente código java:

```
public class Primitiva {  
  
    private final int maxValor = 49, minValor = 1;  
    private String dia; //día del sorteo (jueves o sábado)  
    private double recaudacion; // recaudacion de la jornada  
    private int [] numeros = new int [7]; //números premiados  
    private int [] acertantes = new int [5]; // acertantes de cada  
    //una de las categorías 3,4,5,5+complementario y 6 aciertos.
```

Crear los siguientes métodos de la clase Primitiva:

1. Un método *compruebaNumeros* que reciba un *array* de números de tipo *int* con la combinación ganadora de una primitiva y compruebe que son 7 números (6 más el complementario) y que están entre el 1 y el 49. Devolverá *true* si la combinación es correcta y *false* en caso contrario.
2. Un método *ponerResultados* para dar valor a los atributos *recaudacion*, *numeros* y *acertantes*. Debe usar el método anterior para comprobar que la combinación es correcta.
3. Un constructor que reciba valores para todos los atributos y les dé valor, usando si es necesario el método *ponerResultados*.

```
public boolean comprobarCombinacion (int [] n){  
    int contador = 0;  
    boolean correcto = true;  
    if (n.length != 7) correcto=false;  
    while (contador<n.length && correcto){  
        if(n[contador]<minValor|| n[contador]>maxValor)  
            correcto = false;  
        contador++;  
    }  
    return correcto;  
}  
  
public boolean ponerResultados (int [] n, int a [], double r){  
    if (!comprobarCombinacion(n)) return false;  
    System.arraycopy (n,0,numeros,0,numeros.length);  
    System.arraycopy (a,0,acertantes,0,acertantes.length);  
    recaudacion = r;  
    return true;  
}  
  
public G82P1 (String d, double r, int [] n, int []a){  
    dia = d;  
    ponerResultados (n, a, r);  
}
```

**Problema 2 (2 Puntos).-** Dado el siguiente código java:

```
public class UsaOpera {  
    public static void main(String[] args) {  
        int a = 3, b=5;  
        System.out.println(Opera.doblar(a));  
        System.out.println(Opera.doblar(b));  
        System.out.println(Opera.dividirPorDos (a));  
        System.out.println(Opera.dividirPorDos (b));  
    }  
}
```

Crear la clase Opera, sabiendo que no tiene atributos ni constructor, para que el resultado de ejecutar la clase UsaOpera anterior sea:

```
6  
10  
1  
2
```

Hay varias alternativas para este problema, una posible solución es:

```
public class Opera {  
    public static int doblar (int a){  
        return a*2;  
    }  
  
    public static int dividirPorDos (int a){  
        return a/2;  
    }  
}
```