



## MÓDULO 1. INTRODUCCIÓN A LA PROGRAMACIÓN

### Solución Relación de Problemas N° 7 (Parte I) Arrays

#### Problema 1.

```
import java.util.Scanner;

public class Ej1_RP7 {

    static final int MAX = 10;

    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);

        int [] a;

        a = leer(teclado);

        System.out.println("El mayor número de la colección es: " + mayor(a));

        teclado.close();
    }

    private static int[] leer(Scanner teclado) {
        int [] res = new int[MAX];

        System.out.print("Introduzca " + MAX + " números enteros: ");
        for (int i = 0; i < MAX; i++) {
            res[i] = teclado.nextInt();
        }

        return res;
    }

    private static int mayor(int [] a) {
        int res = a[0];

        for (int i = 1; i < a.length; i++) {
            if (res < a[i]) {
                res = a[i];
            }
        }
    }
}
```

```

    }

    return res;
}

}

```

## Problema 2.

```

import java.util.Scanner;

public class Ej2_RP7 {

    static final int MAX = 10;

    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);
        int num;
        int [] a;

        a = leer(teclado);

        System.out.print("Introduzca el número a buscar: ");
        num = teclado.nextInt();

        if (está(num, a)) {
            System.out.println("El número " + num + " SI está en la colección");
        } else {
            System.out.println("El número " + num + " NO está en la colección");
        }

        teclado.close();
    }

    private static int[] leer(Scanner teclado) {
        int [] res;
        int tam;

        do {
            System.out.print("Introduzca la cantidad de números enteros a leer: ");
            tam = teclado.nextInt();
        } while (tam <= 0);

        res = new int[tam];

        System.out.print("Introduzca " + tam + " números enteros: ");

        for (int i = 0; i < tam; i++) {
            res[i] = teclado.nextInt();
        }

        return res;
    }
}

```

```

// versión 1: usando una variable booleana
/* private static boolean está(int num, int [] a) {
    boolean encontrado = false;

```

```

        int cont = 0;

        while ((cont < a.length) && !encontrado) {
            if (a[cont] == num) {
                encontrado = true;
            } else {
                cont++;
            }
        }

        return encontrado;
    }
}
*/
// versión 2: sin usarla
private static boolean está(int num, int [] a) {
    int cont = 0;

    while ((cont < a.length) && (a[cont] != num)) {
        cont++;
    }

    return cont < a.length;
}

}

```

### Problema 3.

```

import java.util.Arrays;
import java.util.Scanner;

public class Ej3_RP7 {

    static final int MAX = 10;

    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);

        int [] a;

        a = leer(teclado);

        System.out.println("Un elemento Mayor que el Mínimo es: " + mayorMinimo(a));

        teclado.close();
    }

    private static int [] leer(Scanner teclado) {
        int elem, numElementos;
        int [] a = new int[MAX];

        System.out.print("Introduzca la colección de números enteros (0 para
terminar): ");
        numElementos = 0;
        elem = teclado.nextInt();
    }
}

```

```

        while (elem != 0) {
            if (numElementos == a.length) {
                a = Arrays.copyOf(a, 2 * a.length);
            }
            a[numElementos] = elem;
            numElementos++;
            elem = teclado.nextInt();
        }

        return Arrays.copyOf(a, numElementos);
    }

    private static int mayorMinimo(int [] a) {
        int i = 0;
        int res;

        while ((i < a.length-1) && (a[i] == a[i+1])) {
            i++;
        }

        if (i >= a.length-1) { // robusto
            throw new RuntimeException("el array no tiene al menos dos números distintos");
        } else if (a[i] > a[i+1]) {
            res = a[i];
        } else {
            res = a[i+1];
        }

        return res;
    }
}

```

## Problema 4.

```

import java.util.Scanner;

public class Ej4_RP7 {

    static final int MAX = 10;

    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);
        double med;
        int [] clase = new int [MAX];

        System.out.print("Introduzca las estaturas (cm) de " + MAX
+ " alumnos: ");
        for (int i = 0; i < MAX; i++) {
            clase[i] = teclado.nextInt();
        }
    }
}

```

```

    }

    med = media(clase);
    System.out.println("La media es: " + med);
    System.out.println("Número de alumnos más altos que la
media: " + masAltos(clase,med));
    System.out.println("Número de alumnos más bajos que la
media: " + masBajos(clase,med));

    teclado.close();

}

private static double media(int [] clase) {
    double suma = 0;

    for (int i = 0; i < clase.length; i++) {
        suma += clase[i];
    }

    return suma / clase.length;
}

private static int masAltos(int [] clase, double media) {
    int res = 0;

    for (int i = 0; i < clase.length; i++) {
        if (clase[i] > media) {
            res++;
        }
    }

    return res;
}

private static int masBajos(int [] clase, double media) {
    int res = 0;

    for (int i = 0; i < clase.length; i++) {
        if (clase[i] < media) {
            res++;
        }
    }

    return res;
}

```

```
}
```

## Problema 5.

```
import java.util.Scanner;

public class Ej5_RP7 {

    static final int MAX = 10;

    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);
        int [] a = new int [MAX];

        System.out.print("Introduzca " + MAX + " números enteros:
");
        for (int i = 0; i < MAX; i++) {
            a[i] = teclado.nextInt();
        }

        invertirIter(a);
        //invertirRecur(a,0);
        System.out.print("Esos números en orden inverso son: ");
        for (int i = 0; i < MAX; i++) {
            System.out.print(a[i] + " ");
        }

        teclado.close();
    }

    private static void invertirIter(int [] a) {
        int temp;
        for (int i = 0; i <= (a.length-1) / 2; i++) {
            temp = a[i];
            a[i] = a[a.length-1-i];
            a[a.length-1-i] = temp;
        }
    }

    private static void invertirRecur(int [] a, int i) {
        int temp;
        if (i <= (a.length-1) / 2) {
            temp = a[i];
            a[i] = a[a.length-1-i];
            a[a.length-1-i] = temp;
            invertirRecur(a,i+1);
        }
    }
}
```

```
}
```

## Problema 6.

```
import java.util.Scanner;

public class Ej6_RP7 {

    static final int RANGO_DIG = 10;

    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);

        int [] frec;

        frec = calcularFrecuencias(teclado);

        System.out.println("La frecuencia de cada dígito es:");
        for (int i = 0; i < RANGO_DIG; i++) {
            System.out.print(i + ": " + frec[i] + "; ");
        }

        teclado.close();
    }

    private static int [] calcularFrecuencias(Scanner teclado) {
        int [] res = new int [RANGO_DIG]; // se inicializan las casillas a 0
        int dig;

        System.out.print("Introduzca una secuencia de dígitos (negativo para
terminar): ");
        dig = teclado.nextInt();
        while (dig >= 0) {
            if ((0 <= dig) && (dig <= 9)) { // robusto
                res[dig]++;
            }
            dig = teclado.nextInt();
        }

        return res;
    }
}
```

## Problema 7.

```
import java.util.Scanner;

public class Ej7_RP7 {

    static final int RANGO_DIG = 10;

    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);
        int dig;
```

```

    int [] frec;

    frec = calcularFrecuencias(teclado);

    imprimirHistograma(frec);

    teclado.close();
}

private static int [] calcularFrecuencias(Scanner teclado) {
    int [] res = new int [RANGO_DIG]; // se inicializan las casillas a 0
    int dig;

    System.out.print("Introduzca una secuencia de dígitos (negativo para
terminar): ");
    dig = teclado.nextInt();
    while (dig >= 0) {
        if ((0 <= dig) && (dig <= 9)) { // robusto
            res[dig]++;
        }
        dig = teclado.nextInt();
    }

    return res;
}

private static int mayorFrecuencia(int [] frec) {
    int res = frec[0];

    for (int dig = 0; dig < frec.length; dig++) {
        if (frec[dig] > res) {
            res = frec[dig];
        }
    }

    return res;
}

private static void imprimirAsteriscos(int [] frec, int linea) {
    for (int dig = 0; dig < frec.length; dig++) {
        if (frec[dig] >= linea) {
            System.out.print("* ");
        } else {
            System.out.print(" ");
        }
    }
    System.out.println();
}

private static void imprimirHistograma(int [] frec) {
    int mayorFrec;

    mayorFrec = mayorFrecuencia(frec);

    for (int linea = mayorFrec; linea >= 1; linea--) {
        imprimirAsteriscos(frec, linea);
    }

    for (int dig = 0; dig < frec.length; dig++) {
        System.out.print(dig + " ");
    }
}

```



}

}