



MÓDULO 1. INTRODUCCIÓN A LA PROGRAMACIÓN

Solución Relación de Problemas N° 7 (Parte II) Arrays

Problema 8.

```
import java.util.Scanner;

public class Ej8_RP7 {

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

        do {
            System.out.print("Introduzca el límite para calcular
los primos: ");
            N = teclado.nextInt();
        } while (N <= 1);

        eratostenes(N);

        teclado.close();

    }

    private static void inicializar(boolean [] num) {

        num[1] = false;
        for (int i = 2; i < num.length; i++) {
            num[i] = true;
        }

    }

    private static void eliminarMultiplos(boolean [] num, int primo)
{
```

```

        for (int i = 2*primo; i < num.length; i += primo) {
            num[i] = false;
        }
    }

    private static void encontrarPrimos(boolean [] num) {

        for (int i = 2; i <= Math.sqrt(num.length-1); i++) {
            if (num[i]) {
                eliminarMultiplos(num,i);
            }
        }
    }

    private static void imprimirPrimos(boolean [] num) {

        System.out.println("Los números primos menores o iguales
que "
            + num.length + " son:");

        for (int i = 2; i < num.length; i++) {
            if (num[i]) {
                System.out.print(i + " ");
            }
        }
    }

    private static void eratostenes(int N) {
        boolean [] num = new boolean [N+1]; // la casilla 0 no se
usará

        inicializar(num);

        encontrarPrimos(num);

        imprimirPrimos(num);
    }

}

```

Problema 9.

```

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

```

```

public class Ej9_RP7 {

    static final int MAX = 50;

    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);
        int numero, base;
        int [] restos;

        do {
            System.out.print("Introduzca un número natural en
base 10: ");
            numero = teclado.nextInt();
            System.out.print("Introduzca la base a la que
convertirlo (2<=base<=9): ");
            base = teclado.nextInt();
        } while ((numero < 0) || (base < 2) || (base > 9));

        restos = convertir(numero,base);

        System.out.print("Ese número en base " + base + " es: ");
        for (int i = restos.length-1; i >= 0; i--) {
            System.out.print(restos[i]);
        }

        teclado.close();

    }

    private static int [] convertir(int numero, int base)
    {
        int [] restos = new int [MAX];
        int numRestos; // indicará en cada momento el número de
restos obtenidos

        numRestos = 0;
        do {
            restos[numRestos] = numero % base;
            numero = numero / base;
            numRestos++;
        } while (numero != 0);

        return Arrays.copyOf(restos, numRestos);
    }

}

```

Problema 10.

```
import java.util.Scanner;

public class Ej10_RP7 {

    static final int MAX = 50;

    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);
        int cantidad;
        int [] coleccion1, coleccion2;

        System.out.print("Introduzca la cantidad de números que va
a introducir: ");
        cantidad = teclado.nextInt();

        System.out.print("Introduzca la primera colección de
números enteros: ");
        coleccion1 = leer(teclado,cantidad);

        System.out.print("Introduzca la segunda colección de
números enteros: ");
        coleccion2 = leer(teclado,cantidad);

        if (iguales(coleccion1,coleccion2)) {
            System.out.println("SI son iguales");
        } else {
            System.out.println("NO son iguales");
        }

        teclado.close();
    }

    private static int [] leer(Scanner teclado, int cantidad) {
        int [] coleccion = new int [cantidad];

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

        return coleccion;
    }

    private static boolean iguales(int [] c1, int [] c2)
    {
        boolean res;
```

```

        int pos;

        if (c1.length != c2.length) { // robusto
            res = false;
        } else {
            pos = buscar(c1[0],c2);
            if (pos == c2.length) { // no encontrada
                res = false;
            } else {
                pos = (pos +1) % c2.length;
                res = comparar(c1,1,c2,pos);
            }
        }

        return res;
    }

    private static int buscar(int num, int [] c) {
        int i = 0;

        while ((i < c.length) && (num != c[i])) {
            i++;
        }
        return i;
    }

    private static boolean comparar(int [] c1, int i, int [] c2, int
j) {

        while ((i < c1.length) && (c1[i] == c2[j])) {
            i++;
            j = (j + 1) % c2.length;
        }

        return i >= c1.length;
    }

}

```

Problema 11.

```

import java.util.Scanner;

public class Ej11_RP7 {

    static final int DIAS = 30; // noviembre

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

```

```

    int alumnos, dia;
    int [] diasComunes = new int [DIAS]; // posicion 0 => día 1

    System.out.print("Introduzca el número de alumnos: ");
    alumnos = teclado.nextInt();

    for (int i = 0; i < alumnos; i++) {
        System.out.println("Introduzca los días preferidos
por el alumno "
                           + (i+1) + " (introduzca un 0 para
terminar): ");
        dia = teclado.nextInt();
        while (dia != 0) {
            if ((1 <= dia) && (dia <= DIAS)) { // robusto
                diasComunes[dia-1]++;
            }
            dia = teclado.nextInt();
        }
    }

    System.out.print("Los días comunes son: ");
    for (int i = 0; i < DIAS; i++) {
        if (diasComunes[i] == alumnos) {
            System.out.print((i+1) + " ");
        }
    }

    teclado.close();

}

}

```

Problema 12.

```

import java.util.Scanner;

public class Ej12_RP7 {

    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);
        int cantidad;
        int [] valores, indices;

        System.out.print("Introduzca la cantidad de valores a leer:
");
        cantidad = teclado.nextInt();
    }
}

```

```

valores = new int [cantidad];

System.out.print("Introduzca los valores: ");
for (int i = 0; i < cantidad; i++) {
    valores[i] = teclado.nextInt();
}

indices = calcular(valores);

System.out.print("El orden de los indices es: ");
for (int i = 0; i < cantidad; i++) {
    System.out.print(indices[i] + " ");
}

teclado.close();
}

private static int [] calcular(int [] val) {
    boolean [] analizados = new boolean [val.length]; // se
inicializa a false
    int [] ind = new int [val.length];
    int indMenor;

    for (int i = 0; i < val.length; i++) {
        indMenor = buscarIndMenor(val, analizados);
        ind[i] = indMenor;
        analizados[indMenor] = true;
    }
    return ind;
}

private static int buscarIndMenor(int [] val, boolean [] ana) {
    int indMenor;

    indMenor = primerAnalizable(ana);

    for (int i = indMenor+1; i < val.length; i++) {
        if ((!ana[i]) && (val[i] < val[indMenor])) {
            indMenor = i;
        }
    }

    return indMenor;
}

private static int primerAnalizable(boolean [] ana) {
    int i = 0;

    while ((i < ana.length) && (ana[i])) {

```

```

        i++;
    }

    return i;
}
}

```

Problema 13.

```

import java.util.Scanner;

public class Ej13_RP7 {

    static final int N = 5;

    public static void main(String[] args) {
        Scanner teclado = new Scanner(System.in);
        int [] vector = new int [N];
        boolean existe = false;
        int centro = 1;

        System.out.print("Introduzca " + N + " números naturales:");

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

        while (centro <= N-2 && !existe)
        {
            if (sumaPesos(vector,0,centro-1,centro) ==
sumaPesos(vector,centro+1,N-1,centro)) {
                existe = true;
            } else {
                centro++;
            }
        }

        if (existe){
            System.out.println("El centro de este vector es el
indice " + centro);
        }else{
            System.out.println("Este vector no tiene centro ");
        }

        teclado.close();
    }
}

```



```
//Otra opción es hacerlo con dos funciones, una para cada  
sumatorio  
private static int sumaPesos(int [] v, int inicio, int fin, int  
centro)  
{  
    int suma = 0;  
  
    for(int i = inicio; i <= fin; i++)  
    {  
        suma = suma + Math.abs(centro-i) * v[i];  
    }  
    return suma;  
}  
  
}
```