

Programación II - 2011/2012 - Universitat Jaume I

Evaluación continua - Módulo 1 de teoría - Grupo B

4 de abril de 2012

La duración máxima de esta prueba es de 45 minutos. No puedes consultar libros ni apuntes.

Ejercicio 1 (5 puntos)

Analiza el tiempo de ejecución del siguiente programa y exprésalo empleando la notación O . Justifica tu respuesta indicando el coste del método `enigma` y el coste total de cada bloque de código donde se pide.

¿Qué problema resuelve el método `enigma`? ¿Debe cumplir alguna condición el vector para que el método funcione correctamente? ¿Sabrías resolver más eficientemente el mismo problema con las mismas condiciones?

```
import java.util.Scanner;
public class EnigmaCostes {

    public static boolean enigma(int [] vector, int dato) { // COSTE: -----
        return enigma(vector, dato, 0, vector.length - 1);
    }
    public static boolean enigma(int [] vector, int dato, int a, int b) {
        if (a > b)
            return false;
        else {
            int c = (a + b) / 2;
            if (vector[c] == dato)
                return true;
            else
                return enigma(vector, dato, a, c - 1) || enigma(vector, dato, c + 1, b);
        }
    }
    public static void main (String [] args) {
        Scanner entrada = new Scanner(System.in);
        System.out.print("Digame un numero entero positivo: ");
        int n = entrada.nextInt(), cont = 0;

        int [] v = new int[n]; // COSTE: -----

        for (int i = 1; i < n; i = i * 2) // COSTE: -----
            v[i] = 2 * i + 1;

        for (int i = 1; i <= 100; i = i * 2) // COSTE: -----
            if ( enigma(v, i) )
                cont++;

        for (int i = 2; i <= n / 2; i = i + 2) // COSTE: -----
            if ( enigma(v, i) )
                cont++;

        for (int i = n / 2; i < n; i = i + 2) // COSTE: -----
            for (int j = n - 2; j < n; j = j + 1)
                if ( enigma(v, i * j) )
                    cont++;

        System.out.println(cont);
    }
}
```

Ejercicio 2 (5 puntos)

1. ¿Cuál es el resultado del siguiente programa si el usuario introduce el número 97531? ¿Y si introduce el número 9?

```
import java.util.Scanner;
import java.util.Arrays;
public class MisterioRecursion {

    public static int[] misterio1(int numero) {
        int[] v = new int[ misterio2(numero) ];
        misterio3(numero, v, 0);
        return v;
    }
    public static int misterio2(int numero) {
        if (numero == 0) return 0;
        else return 1 + misterio2(numero / 10);
    }
    public static void misterio3(int numero, int[] vector, int i) {
        if (numero > 0) {
            vector[i] = numero % 10;
            misterio3(numero / 10, vector, i + 1);
        }
    }
    public static void main (String [] args) {
        Scanner entrada = new Scanner(System.in);
        System.out.print("Digame un numero entero positivo: ");
        System.out.println( Arrays.toString( misterio1( entrada.nextInt() ) ) );
    }
}
```

2. Describe brevemente para qué sirve el método `misterio1` y para qué sirve el método `misterio2`.
3. Expresa empleando la notación O el tiempo de ejecución de cada uno de los métodos `misterio1`, `misterio2` y `misterio3`.
4. Escribe un programa que produzca de forma iterativa y con el mismo coste temporal, sin emplear recursión para nada, el mismo resultado que el programa anterior.
5. ¿Cuál es el coste espacial de cada uno de los dos programas? ¿Cuál es preferible teniendo en cuenta eso?