EJERCICIOS TEMA 1

Algoritmia y Complejidad

Realizado por:

Gabriel López Cuenca
Miguel Ángel Losada Fernández
Sergio Sanz Sacristán
Álvaro Zamorano Ortega

Ejercicio 3

$$T(n) = 1 + 1 + \underbrace{\sum_{n=x}^{y} (1)}_{n} + 1 + \max\left(1,\ 1+1+\underbrace{\sum_{n=x}^{y}\left(1+\sum_{j=3x}^{3y}(1)\right)}_{n(1+3n)}\right) + 1 + T\left(\frac{n}{2}\right) + 1$$

$$T\left(\frac{n}{2}\right) \Rightarrow x + \frac{y-x}{2} \Rightarrow \frac{2x+y-x}{2} = \frac{x+y}{2} \longrightarrow \text{es constante}$$

$$T(n) = 2 + n + 1 + \left(4 + T\left(\frac{n}{2}\right) + n + 3n^2\right) = 7 + 2n + 3n^2 + T\left(\frac{n}{2}\right)$$

$$n = 2^K \Rightarrow T(2^K) = 7 + 2\cdot 2^K + 3\cdot(2^K)^2 + T\left(\frac{2^K}{2}\right)$$

$$T(2^K) = 7 + 2\cdot 2^K + 3\cdot 4^K + T(2^{K-1})$$

$$T(2^K) = x^K \Rightarrow x^K = 7 + 2\cdot 2^K + 3\cdot 4^K + x^{K-1}$$

$$x^K - x^{K-1} = 7 + 2\cdot 2^K + 3\cdot 4^K$$

$$x^{K-1}(x-1) = 7 + 2\cdot 2^K + 3\cdot 4^K$$

HOMOGENEA: $x^{K-1}(x-1) = 0 \Rightarrow 1$ RAIZ $\Rightarrow \boxed{x^{(H)} = A}$

PARTICULAR: $x^{(P)} = B + C\cdot 2^K + D\cdot 4^K$

$$x^{(P)} = BK + C\cdot 2^K + D\cdot 4^K$$

$\uparrow$

La raiz de B coincide con la raiz de la homogénea (1^K) $\Rightarrow$ Necesitamos multiplicarla por K para que tenga importancia

$$x = x^{(H)} + x^{(P)} = A + BK + C\cdot 2^K + D\cdot 4^K$$

$$2^K = n \Rightarrow x = A + B\cdot\log_2(n) + Cn + Dn^2 = T(n)$$

$$\boxed{O(n^2)}$$

Ejercicio 5

Realiza una función que determine si un número recibido como parámetro es primo. Realiza un análisis de eficiencia y de complejidad.

```java
public class Tema1_EJ5 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        System.out.println("Introduzca un número: ");
        Scanner s = new Scanner (System.in);
        int n = s.nextInt();
        boolean primo = numeroPrimo(n);
        if (primo) {
            System.out.println(n + " es " + " primo\n");
        } else {
            System.out.println(n + " no es " + " primo\n");
        }
    }

    /**
     * Ejercicio 5
     * @param num
     * @return
     */
    public static boolean numeroPrimo(int num) {
        boolean primo = true;
        int i=2;
        //Mientras el numero no sea divisible por i, e i sea menor que el
        //numero se hace el bucle
        while ((primo) && (i<num)) {
            if (num%i==0){
                //Si la división por i es exacta, el numero no es primo
                primo = false;
            }
            //La i avanza hasta llegar al número
            i++;
        }

        //Devolver si el número es primo o no
        return primo;
    }
}
```

$$T(n) = 1 + 1 + \underbrace{\underbrace{1 + \max(1,0)}_{if} + \underbrace{1}_{i++}}_{\text{instruccion del while}} + \underbrace{1}_{\text{Condición while}} + \sum (1 + \max(1,0) + 1 + 1)$$

$$T(n) = 6 + \sum_{i=1}^{n} (4) = 6 + 4n = \boxed{O(n)}$$

Ejercicio 9

Realizar una función recursiva que calcule el siguiente sumatorio: S= 1+2+3+4+….+n-1+n. Realiza un análisis de eficiencia y de complejidad.

```java
public class Tema1_EJ9 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args){
        System.out.println("Introduzca un número:");
        Scanner s = new Scanner (System.in);
        int n = s.nextInt();
        int suma = sumatorio(n);
        System.out.println("La suma de 1 a " + n + " es: " + suma + "\n");
    }

    /**
     * Ejercicio 9
     * @param num
     * @return
     */
    public static int sumatorio(int num){
        //Caso base: numero=1
        if(num==1){
            //Su suma es 1
            return 1;
        }else{
            //Suma del anterior + llamada recursiva del siguiente valor
            return num+sumatorio(num-1);
        }
    }
}
```

$$T(n) = 1 + \max(1,\ 1 + T(n-1)) = 1 + 1 + T(n-1)$$

$$T(n) = 2 + T(n-1)$$

$$T(n) - T(n-1) = 2$$

$$T(n) = x^n \Rightarrow x^n - x^{n-1} = 2$$

$$x^{n-1}(x-1) = 2$$

Homogenea: $x^{n-1}(x-1)=0 \rightarrow$ RAIZ: $x=1 \rightarrow \boxed{x^{(H)} = A \cdot 1^n = A}$

Particular: $x^{(P)} = B \rightarrow$ Al tener el mismo grado que la raiz de la homogenea, se multiplica por n

$$\boxed{x^{(P)} = B \cdot n}$$

$$x = x^{(H)} + x^{(P)} \rightarrow x = A + Bn = T(n) \rightarrow \boxed{O(n)}$$