

Laboratorio Nro. 1

Escribir el tema del laboratorio

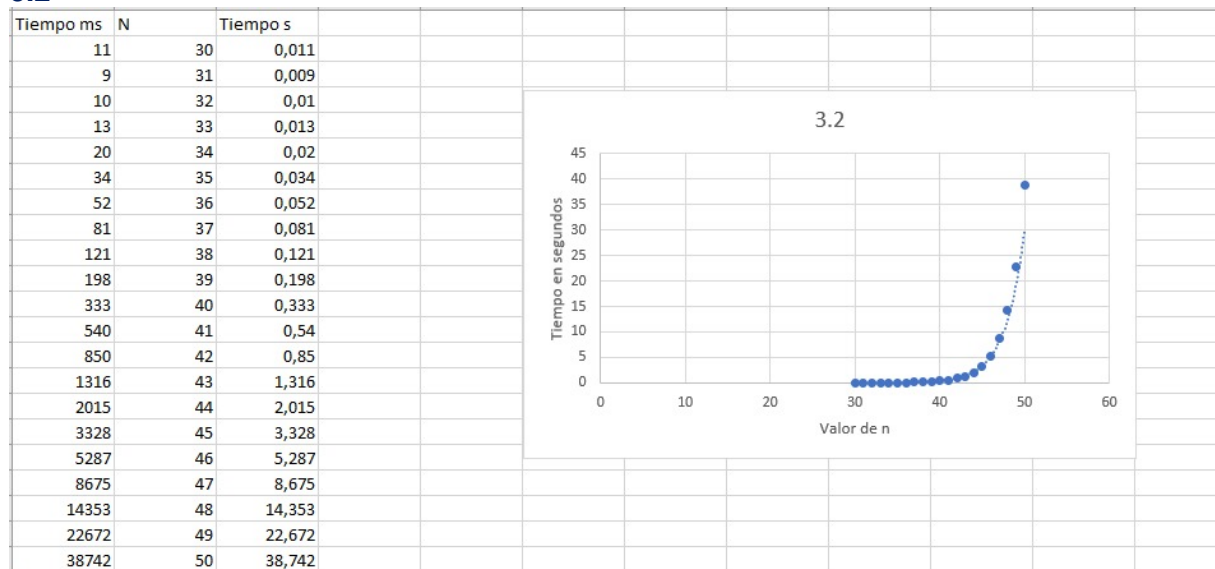
Juan Jose Madrigal Palacio
Universidad Eafit
Medellín, Colombia
jjmadrigap@eafit.edu.co

Luis Ángel Jaimes Mora
Universidad Eafit
Medellín, Colombia
lajaimesm@eafit.edu.co

3) Simulacro de preguntas de sustentación de Proyectos

3.1 //Wolfram $T(n) = C_3 \cdot 2^n + (C_1 + C_2) \Rightarrow T(n) = 4 \cdot (2^n) + 4$
 // $T(n)$ es $O(4 \cdot (2^n) + 4)$
 // $T(n)$ es $O(4 \cdot (2^n))$
 // $T(n)$ es $O(2^n)$

3.2



Para calcular la cantidad de formas que existen de llenar un rectángulo de 50x2 cm² con rectángulos de 1x2 cm² el algoritmo se demora 38,742 segundos para realizar este calculo

3.3 No, ya que al ser de $O(2^n)$ no es eficiente para números muy grandes como sería el caso del Puerto Antioquia

PhD. Mauricio Toro Bermúdez
Docente | Escuela de Ingeniería | Informática y Sistemas
Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
Tel: (+57) (4) 261 95 00 Ext. 9473

ESTRUCTURA DE DATOS 1

Código ST0245

3.4 -El ejercicio groupSum6 se trata de un problema/ejercicio en el cual nos dan un grupo o arreglo de números enteros llamado 'nums', junto con un número el cual en nuestro caso llamamos 'Inicio' el cual usaremos para saber desde cual número de 'nums' partir para encontrar si existe o no una suma de los números en 'nums' la cual sea igual a otro número entero que nos dan que en nuestro caso llamamos 'objetivo', pero tenemos que asegurarnos que para llegar a 'objetivo' debemos usar todos los números 6 que hallan en 'nums'.

-El método llamado groupSum6 con parámetros 'Inicio' un número entero, 'nums' un arreglo de enteros, y 'objetivo' un número entero; se hace una condición de parada en la cual si 'Inicio' es = al largo de 'nums' es true devuelve el booleano para sí 'objetivo' es = 0; Luego si el entero en posición 'Inicio' en 'nums' es != de 6 y se llama a groupSum6 con 'Inicio+1', 'nums', 'objetivo'; si todo esto es true se devuelve true y acaba el código; o también si groupSum6 con 'Inicio'+1, 'nums', 'objetivo'-'nums' referente a 'Inicio' da true se devuelve true; sino es ninguna se devuelve false y se termina el código.

3.5 Recursión 1

- bunnyEars2: $T(n) = C_1 + C_2 * n$
- triangle: $T(n) = C_1 + C_2 * n$
- count7: $T(n) = C_1 + (C_2 * \text{Log}(n)) / \text{Log}(10)$
- count8: $T(n) = C_1 + (C_2 * \text{Log}(n)) / \text{Log}(10)$
- countX: $T(n) = C_1 + C_2 * n$

Recursión 2

- groupSum6: $T(n) = c_1 2^{(n-1)} + C_2 (2^n - 1)$
- groupNoAdj: $T(n) = 2^{(n/2)} (c_2 (-1)^n + c_1) - C_2$
- groupSum5: $T(n) = c_1 (1/2 (3 - \sqrt{13}))^n + c_2 (1/2 (3 + \sqrt{13}))^n - C_2/3$
- groupSumClump: $T(n) = C_1 + C_2 * n$
- splitArray: $T(n) = c_1 2^{(n-1)} + C_2 (2^n - 1)$

3.6 - $T(n)$; se refiere a la función o punto del algoritmo en cuestión evaluado en n.

- n ; se trata de la variable que se ingresa como parámetro al método/algoritmo desde la cual se procede a trabajar.

- c_1 , c_2 ; son las operaciones fijadas en el código del algoritmo, las cuales pueden ser tanto la condición de parada, sumas, restas, etc.

4) Simulacro de Parcial

4.1 Start+1 , nums , target

4.2 $T(n) = T(n/2) + C$

4.3.1 Línea 4.

.2 Línea 5.

.3 Línea 6.

4.4 d) El máximo valor de un arreglo a y es $O(n)$

PhD. Mauricio Toro Bermúdez

Docente | Escuela de Ingeniería | Informática y Sistemas
 Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
 Tel: (+57) (4) 261 95 00 Ext. 9473



ESTRUCTURA DE DATOS 1
Código ST0245

- 4.5.1** Línea 2. return n
 Línea 3. n-1
 Línea 4. n-2
.2 b) $T(n)=T(n-1) + T(n-2) + C$
- 4.6.1** Línea 10. sumaAux(n,i+2)
.2 Línea 12. sumaAux(n,i+1)
- 4.7.1** Línea9.S, i + 2, t - S[i]
.2 Línea10.S, i + 1, t
- 4.8.1** Línea9. Return 0
.2 Línea13.nj+ni
- 4.9** c) 22
- 4.10** b) 6
- 4.11** **.1** Línea4. n-1 , lucas(n-2)
.2 c) $T(n)=T(n-1)+T(n-2)+c$, que es $O(2^n)$
- 4.12** **.1** Línea13. Math.max(k,d)
.2 Línea17. fi+fj
.3 Línea18. sat

PhD. Mauricio Toro Bermúdez

Docente | Escuela de Ingeniería | Informática y Sistemas
 Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
 Tel: (+57) (4) 261 95 00 Ext. 9473

