

Laboratorio Nro. 1

Recursión

Camilo Cañas Jaramillo

Universidad Eafit

Medellín, Colombia

ccanasj@eafit.edu.co

Laura Marin Velez

Universidad Eafit

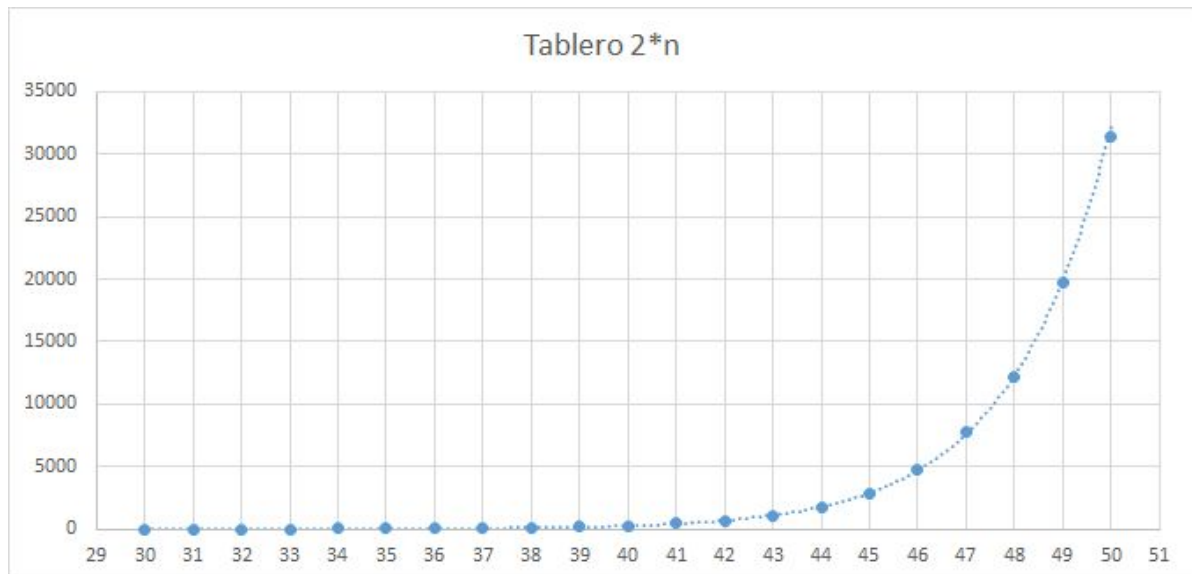
Medellín, Colombia

lmarinv2@eafit.edu.co

3) Simulacro de preguntas de sustentación de Proyectos

3.1 $T(n) = c3 + T(n-1) + T(n-2)$ Lo cual da una $O(2^n)$

3.2 El algoritmo se demora alrededor de 32 segundos en calcular la cantidad de rectángulos de $1*2\text{cm}^2$ necesarios para llenar un rectángulo de $50*2\text{cm}^2$



3.3 No, ya que el algoritmo crece exponencialmente con el aumento del tamaño del contenedor lo cual lleva a tiempos más largos de ejecución del algoritmo haciéndolo menos eficiente y práctico para grandes medidas.

3.4 El algoritmo de groupSum5 se encarga de sumar los números de un arreglo de enteros, en el cual se ingresa la posición de inicio del arreglo y el “Objetivo” al que se desea llegar con la suma de los números del arreglo, con la condición de que todos los números que sean múltiplos de 5 se tendrán en cuenta para la suma de cualquier objetivo mientras que el número que siga de un múltiplo de 5 no le siga un 1, si esto ocurre el número no se contará para la suma. El algoritmo compara cada posición del arreglo recursivamente para comprobar y sumar los valores de cada posición ya sean, múltiplos de 5 y no tengan un 1 al lado, como los demás números.

4) Simulacro de Parcial

4.1 *start + 1, nums , target*

4.2 *b*

4.3

4.3.1 *n - 1, a, b, c*

4.3.2 *res, a*

4.3.3 *b , c*

4.4 *e*

4.5

4.5.1 *Línea 2: return n*

Línea 3: n-1

Línea 4: n-2

4.5.2 *b*

4.6

4.6.1 *1*

4.6.2 *sumaAux(n, n.charAt(i + 1))*

4.8

4.8.1 *return 0;*

4.8.2 *ni + nj*

4.9 *c*

4.10 *b*

4.11

4.11.1 *n-1 , Lucas(n-2)*

4.11.2 *c*

4.12

4.12.1 0

4.12.2 *Math.max(fi,fj)*

4.12.3 sat