

Laboratorio Nro. 1 Recursión

Juan Pablo Madrid Florez
Universidad Eafit
Medellín, Colombia
jpmadridf@eafit.edu.co

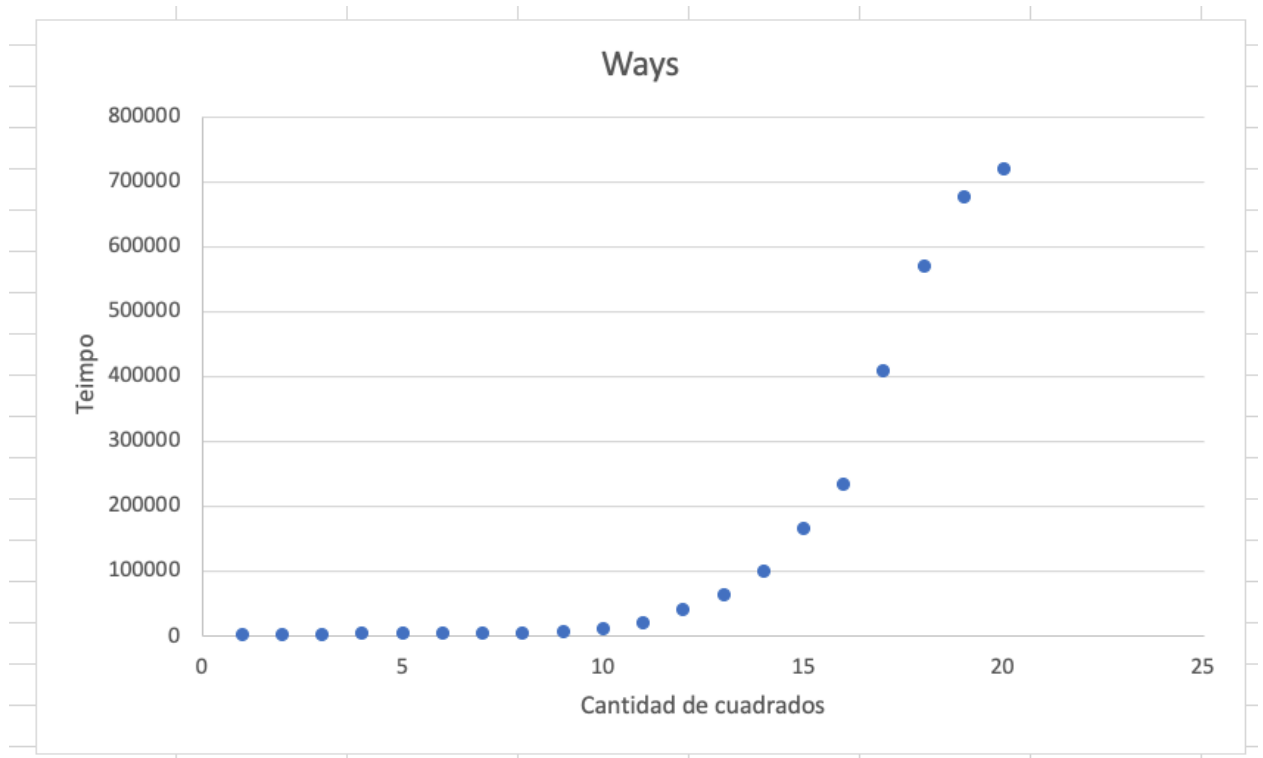
Abelino Sepúlveda Estrada
Universidad Eafit
Medellín, Colombia
asepulvede@eafit.edu.co

Andrés Gómez Arango
Universidad Eafit
Medellín, Colombia
agomez10@eafit.edu.co

3) Simulacro de preguntas de sustentación de Proyectos

3.1 $T(n) = T(n-1) + T(n-2) + c$ que es $O(2^n)$.

3.2



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

El tiempo esta en nanosegundos y a pesar de que al principio los datos eran de una pequeña medida al ir creciendo se fue elevando la cantidad de tiempo casi de una forma exponencial, también al momento de evaluar 50 ocurre que el programa nos da un valor negativo y se demora mas de lo previsto y su tiempo fue de 4.0711256996E10 nanosegundos.

3.3 No es viable ya que al calcular áreas muy grandes el sistema fallaría y no sería muy eficaz.

3.5 y 3.6

Recursión 1:

$T(n) = T(n-1) + c$ que es $O(n)$ (donde n es el número al que se le quiere hallar factorial)

$T(n) = T(n-1) + T(n-2) + c$ que es $O(2^n)$ (n es el número que se le quiere hallar su valor más la sucesión Fibonacci)

$T(n) = T(n-1) + c$ que es $O(n)$ (donde n es el número de filas de un triángulo hecho de bloques)

$T(n) = T(n-1) + c$ que es $O(n)$ (donde n es valor de la potencia, es decir el número por el que está elevada la base)

$T(n) = T(n-1) + c$ que es $O(n)$ (donde n es la longitud del string)

Recursión 2:

$T(n) = 2T(n-1) + c$ que es $O(2^n)$ (Donde n es la longitud del arreglo menos start)

$T(n) = 2T(n-1) + c$ que es $O(2^n)$ (Donde n es la longitud del arreglo menos start)

$T(n) = 2T(n-1) + c$ que es $O(2^n)$ (Donde n es la longitud del arreglo menos start)

$T(n) = 2T(n-1) + c$ que es $O(2^n)$ (Donde n es la longitud del arreglo menos start)

$T(n) = 2T(n-1) + c$ que es $O(2^n)$ (Donde n es la longitud del arreglo menos start)

4) Simulacro de Parcial

4.1.1 return true

4.1.2 if(s.substring(0,1).equals(s.substring(s.length()-1))).

4.2 d.

4.3.1 int res= solucionar(n-a,a,b,c)+1;

4.3.2 res= Math.max(res,solucionar(n-b,a,b,c)+1);

4.3.3 res= Math.max(res,solucionar(n-c,a,b,c)+1);

4.5.1 if(T==0) return 1;

4.5.2 if(T<0) return 0;

4.5.3 return f1+f2+f3;

4.6.1 0.

4.6.2 n.charAt(i+1) - '0'.

4.8.1 return 0.

4.8.2 ni+nj.

4.10 b.

4.11.1 Return lucas(n-1)+lucas(n-2);

4.11.2 C.

4.12.1 0.

4.12.2 Math.max(fi,fj).

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

4.12.3 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

