**Instituto Politécnico Nacional**

**Escuela Superior de Cómputo**

**Análisis de Algoritmos**

**Ejercicio 5: Análisis de Algoritmos recursivos.**

**Sampayo Hernández Mauro**



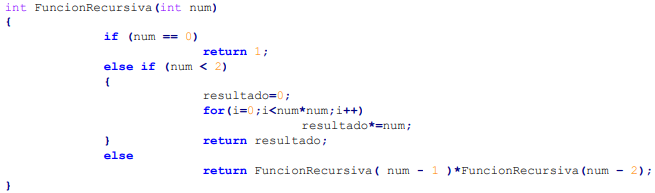
**Grupo:** 3CM2

**Profesor:** Edgardo Adrián Franco Martínez

**Fecha de entrega:** 22 de octubre de 2018

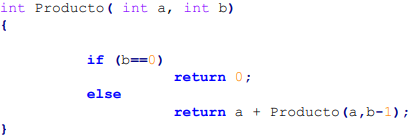
**Ejercicio 5: Análisis de Algoritmos recursivos.**

**1. Calcular la cota de complejidad para el algoritmo de la siguiente función recursiva:**



Análisis: Contando retornos e iteraciones:

**2. Calcular la complejidad de la implementación recursiva del producto.**



Contando comparaciones, aritméticas y retornos:

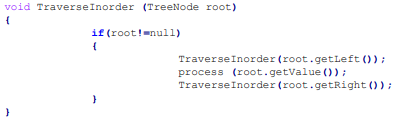
Condiciones Iniciales:

T (0) = 2

T (1) = 4

Análisis:

**3. Calcular el costo de un recorrido in-orden en un árbol binario completamente lleno.**



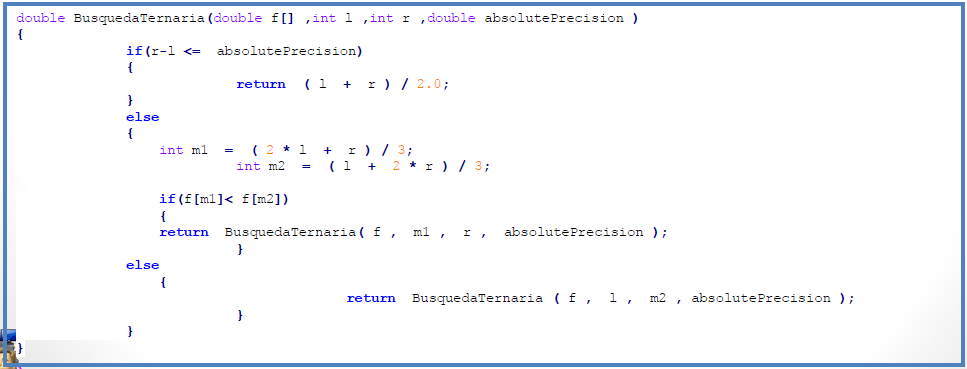
Considerando comparaciones y retornos:

T (0) = 1

T (1) = 3

Análisis:

**4.-Calcular la cota de complejidad del algoritmo de búsqueda ternaria.**



T(n) = 1 + T()

\*Reordenando términos

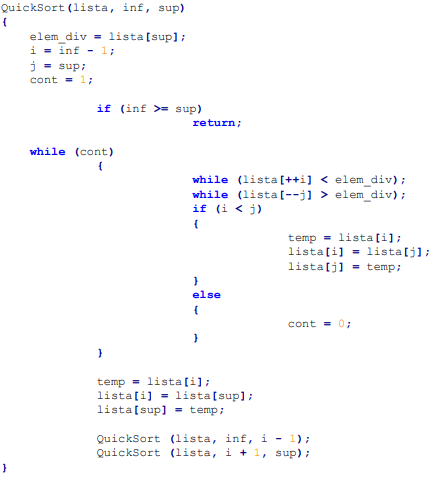
T() + 1 -> O(c)

\*a=1,b=3…Con el teorema maestro

nlog31 -> O(c)

(nlog31lgn)=(lgn)

**5. Calcula la cota de complejidad del algoritmo de ordenamiento Quicksort**



T(n) = 2T() + n

\*Reordenando términos

2T() + n -> O(n)

a=2 , b=2 Con el teorema maestro

nlog22 -> O(n)

(nlog22log(n))=(nlog(n))

**6. Resolver las siguientes ecuaciones y dar su orden de complejidad:**

𝑇𝑛=3(𝑛–1)+4𝑇(𝑛–2)⇒𝑛>1; 𝑇(0)=0;𝑇(1)=1.

T(n) = 3T(n-1) +4T(n-2)

\*Reordenando términos

T(n) – 3T(n-1) – 4T(n-2) = 0

\*Obtenemos la ecuación característica

(x -4)(x+1)

\*por lo tanto

R1 = 4 C14n + C2(-1)2

R2 = -1 Si T(0)=0 y T(1)=1 C1=1/5 y C2=-1/5

T(n)= (1/5)4n –(1/5)(-1)2 -> O(4n)

𝑇𝑛=3(𝑛–1)+4𝑇(𝑛–2)+(𝑛+5)2⇒𝑛>1; 𝑇0=5,𝑇(1)=27

T(n) = 3T(n–1)+4T(n–2)+(n+5)2n

Reordenando términos

T(n) – 3T(n–1)- 4T(n–2)=(n+5)2n

b=2, d=1…Obtenemos la ecuación característica

(x -4)(x+1)(x-2)2

por lo tanto

R1 = 4 C14n + C2(-1)2 + C32n + C4n22

R2 = -1

R3 = 2

R3 = 2 Si C4!=0

T(n)= C14n + C2(-1)2 + C32n + C4n22 -> O(n2n)

(𝑛)−2𝑇(𝑛−1)=3𝑛⇒𝑛≥2; 𝑇(0)=0,𝑇(1)=1

T(n) – 2T(n–1)=3n

b=3,d=0 Obtenemos la ecuación característica

(x -3)(x-2)

\*por lo tanto

R1 = 3 C13n + C22n

R2 = 2 Si T(0)=0 y T(1)=1 C1=1 y C2=-1

T(n)= C13n + C222 -> O(3n)

**7. Calcula la cota de complejidad que tendrían los algoritmos con los siguientes modelos recurrentes:**

𝑇𝑛=3𝑇+4𝑇+2𝑛2+𝑛

T(n) = n +3 T()

Reordenando términos

3T() + n -> O(n)

a=3,b=3…Con el teorema maestro

nlog33 -> O(n)

(nlog33lgn)=(nlogn)

T(n) = n2 + 4T()

\*Reordenando términos

4T() + n2 -> O(n2)

\*a=4,b=2…Con el teorema maestro

nlog24 -> O(n2)

(nlog24lgn)=(n2logn)

𝑇𝑛=(𝑛–1)+𝑇(𝑛–2)+𝑇 𝑠𝑖 𝑛>1; 𝑇(0)=1,𝑇(1)=1

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

\*Reordenando términos

T(n) – T(n-1)- T(n-2) = 0

\*b=1,d=0…Obtenemos la ecuación característica

(x2 – x -1)

\*por lo tanto

R1 = C12 + C22

R2 = Si C1!=0 o C2!=0

2 -> (cn)

T(n) = T()

\*Reordenando términos

T() + 0 -> O(c)

\*a=1,b=2…Con el teorema maestro

nlog21 -> O(c)

(nlog21lgn)=(lgn)

𝑇𝑛=𝑇 +2𝑇 +2

T(n) = T() + 0

\*Reordenando términos

T() + 0 -> O(c)

\*a=1,b=2…Con el teorema maestro

nlog21 -> O(c)

(nlog21lgn)=(lgn)

T(n) = 2 + 2T()

\*Reordenando términos

2T() + 2 -> O(c)

\*a=2,b=4…Con el teorema maestro

nlog42- -> O(nlog42)

(nlog42)

𝑇𝑛=2𝑇 +4𝑇 +10𝑛2+5𝑛

T(n) = 5n + 4T()

\*Reordenando términos

4T() + 5n -> O(n)

\*a=4,b=4…Con el teorema maestro

nlog44 -> O(n)

(nlog44lgn)=(nlgn)

T(n) = 10n2 + 2T()

\*Reordenando términos

2T() + n2 -> O(n2)

\*a=2,b=2…Con el teorema maestro

nlog22+ -> O(n2)

(n2)