ESTRUCTURA DE DATOS 1
Código ST0245

Laboratorio Nro. 2
Complejidad algorítmica y arreglos

Isabella Echeverri Villa
Universidad Eafit
Medellín, Colombia
iecheverrv@eafit.edu.co

Samuel Arturo Flórez Rincón
Universidad Eafit
Medellín, Colombia
saflorezr@eafit.edu.co

3) Simulacro de preguntas de sustentación de Proyectos

3.1

Grafica InsertionSort

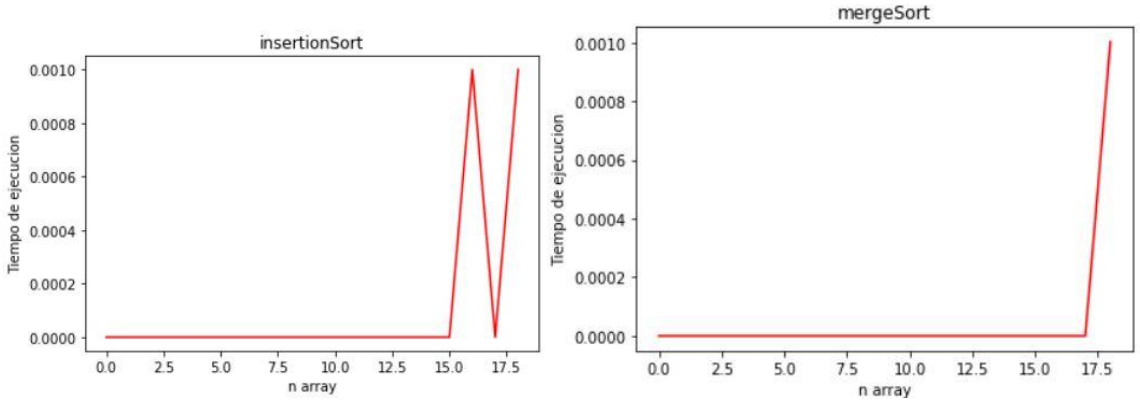
X	Y
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0
10	0
11	0
12	0
13	0
14	0
15	0
16	0.00099897
17	0
18	0
19	0
20	0

Grafica MergeSort

X	Y
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0
10	0
11	0
12	0
13	0
14	0
15	0
16	0
17	0
18	0
19	0
20	0.00100255

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

3.2



Eje X: El eje X corresponde al tamaño del problema
Eje Y: El eje x corresponde al tiempo de ejecución
La variable independiente es el tamaño del problema, puesto que de ella depende el tiempo de ejecución.

3.3 Según la complejidad en tiempo de InsertionSort, no es apropiado utilizar este algoritmo para un videojuego con millones de elementos en una escena y demandas de tiempo real, pues este algoritmo es un poco lento y complejo para esto. Su complejidad es polinómica y crece cada vez que aumenta la cantidad de datos.

3.4 En la complejidad del algoritmo mergesort aparece un logaritmo, pues al poner la ecuación en Wolfram Alpha (cosa que se debe hacer debido a que este algoritmo utiliza recursión) la ecuación de recurrencia asi lo muestra. Este logaritmo se debe al hecho de que al declarar las variables Y y Z, se hace el llamado recursivo pero con la mitad del arreglo esta vez, lo que genera el logaritmo.

3.5 Para que InsertionSort sea más rápido que mergeSort es conveniente que los datos esten ordenados de menor a mayor y, de hecho, si fueran todos iguales no haría que InsertionSort fuera más lento.

3.7 Complejidad ejercicios de codingBat

3.8

CountEvens

T(n)=c1+c2*n+c3+c4+c5+c6
O(c1+c2*n+c3+c4+c5+c6)
O(c2*n)
O(n)

Donde n es la longitud del arreglo que se recibe en los parámetros.

bigDiff

T(n)=c1+c2+c3*n+c4*n+c5*n+c6
O(c1+c2+c3*n+c4*n+c5*n+c6)
O(c3*n+c4*n+c5*n)
O(c5*n)
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

Donde n es la longitud del arreglo que se recibe en los parámetros
centeredAverage
T(n)=c1+c2+c3+c4*n+c5*n+c6*n+c7
O(c1+c2+c3+c4*n+c5*n+c6*n+c7)
O(c4*n+c5*n+c6*n)
O(c5*n)
O(n)
Donde n es la longitud del arreglo que se recibe en los parámetros

Sum13

T(n)=c1+c2*n+c3*n+c4*n+c5
O(c1+c2*n+c3*n+c4*n+c5(
O(c2*n)
O(n)

Donde n es la longitud del arreglo que se recibe en los parámetros

Sum67

T(n)=c1+c2+c3*n+c4*n+c5*n+(c6*n)*n)+c7+c8+c9
O(c1+c2+c3*n+c4*n+c5*n+(c6*n^2)+c7+c8+c9
O(c6*n^2)
O(n^2)

Donde n es la longitud del arreglo que se recibe en los parámetros

ARRAY3

maxSpan

T(n)=c1+c2+c3+c4+c5+c6*n+c7
O(c1+c2+c3+c4+c5+c6*n+c7)
O(c6*n)
O(n)

canBalance

T(n)=c1+T(n+1)+c2
T(n) = c1 - c^3 n En wólfram Alpha
O(c1-c3*n)
O(c3*n)
O(n)

LinearIn

T(n)=c1+c2*n+(c3*m)*n+c4*n*m+c5*n*m+c6
O(c1+c2*n+(c3*m)*n+c4*n*m+c5*n*m+c6)
O(c2*n+(c3*m)*n+c4*n*m+c5*n*m)
O(c4*n*m)
O(nxm)

Donde n y m son las longitudes de los dos arreglos que el método recibe al principio

CountClumps

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

O(n)

SeriesUp

O(n)

4) Simulacro de Parcial

4.1 *a*

4.2 *b.* $O(m \times n \times \sqrt{n})$

4.3 *a.* $O(n^3)$

4.4 $O(m \times n)$ Complejidad en tiempo

Complejidad en memoria:

4.5 $\boldsymbol{T(n) = T\left(\frac{n}{10}\right) + c,\ que\ es\ O(log_{10} n)}$

4.5.2 B. No

4.6 a. $\boldsymbol{O(n^3 + (log(log(m)) + m \times \sqrt{n})}$

4.7 1. Falso

2. Verdadero

3.

4. Verdadero

4.8 a. Esta ejecuta T(n)=c+T(n-1) pasos, que es O(n)

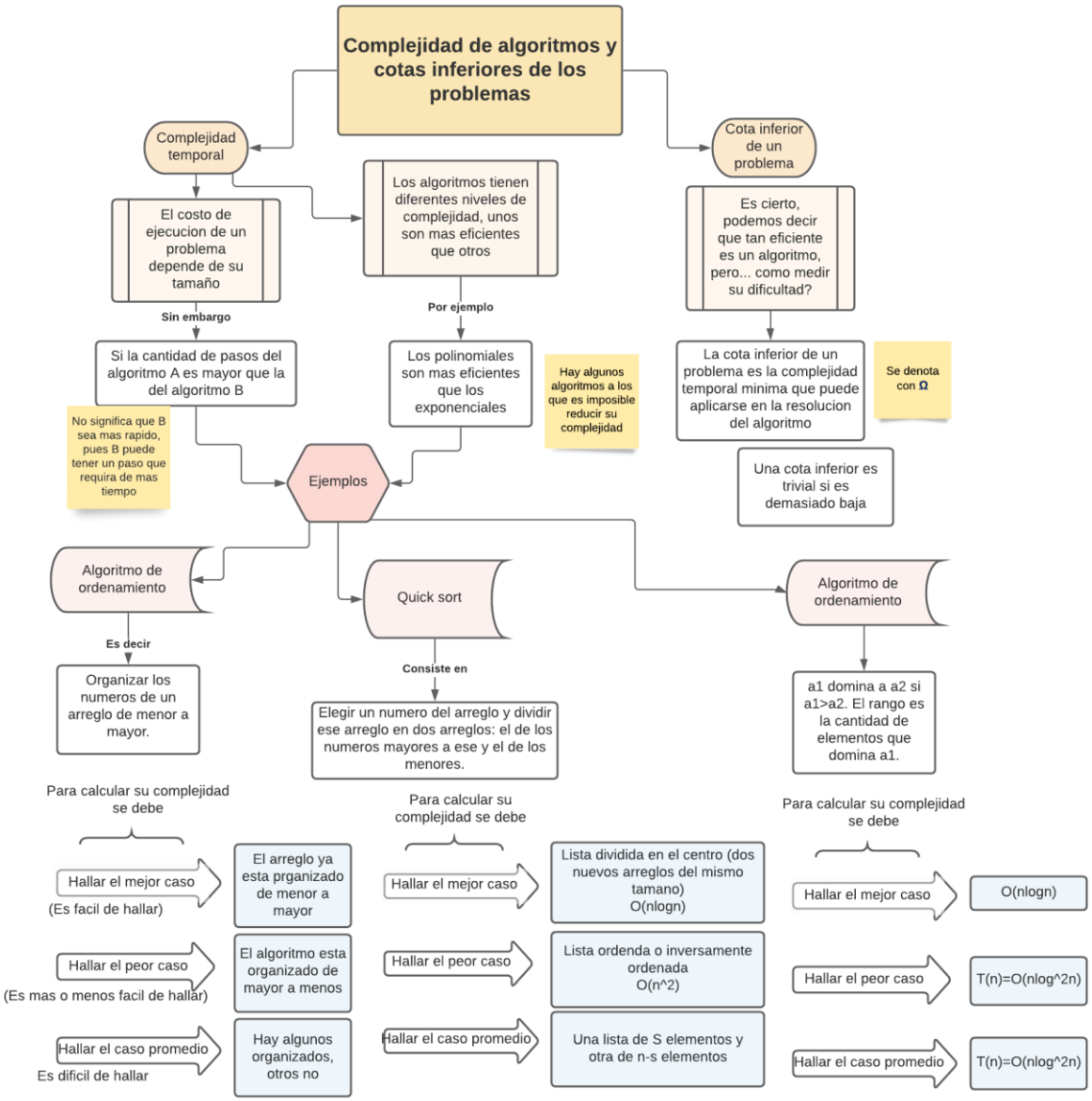
4.9 b. T(n)=2T(n/2)+ $\boldsymbol{n^2}$***)***

5) Lectura recomendada (opcional)

Mapa conceptual

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



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