

Laboratory practice No. 2: Big O Notation

Laura María Giraldo Castrillón
Universidad Eafit
Medellín, Colombia
lmgiraldo1@eafit.edu.co

Andrés Felipe Oquendo Usma
Universidad Eafit
Medellín, Colombia
afoquendou@gmail.com

3) Practice for final project defense presentation

1. Insertion sort

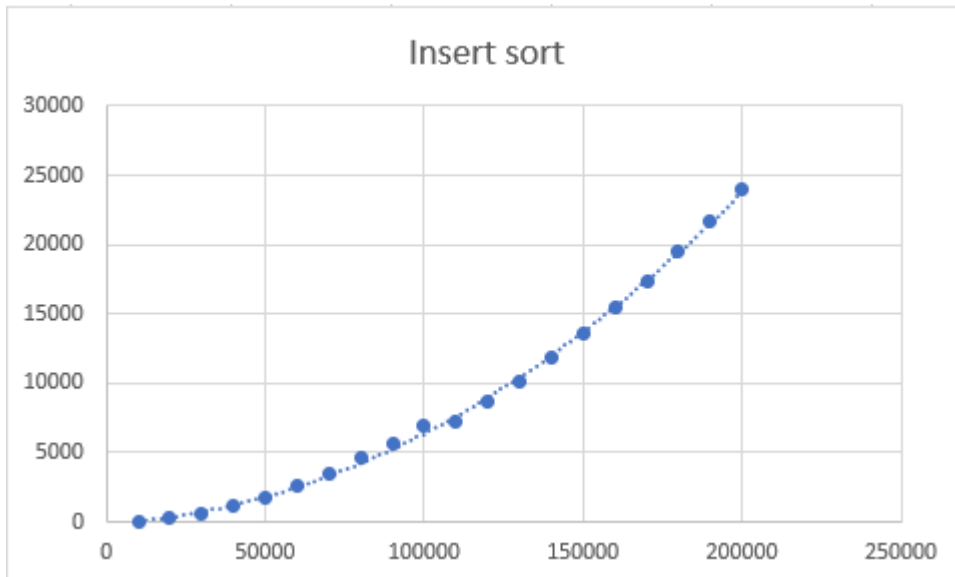
TIME	ARRAY LENGTH
73	10000
283	20000
552	30000
1121	40000
1744	50000
2539	60000
3452	70000
4565	80000
5650	90000
6970	100000
7271	110000
8661	120000
10114	130000
11842	140000
13588	150000
15400	160000
17328	170000
19473	180000
21682	190000
23969	200000

Merge sort

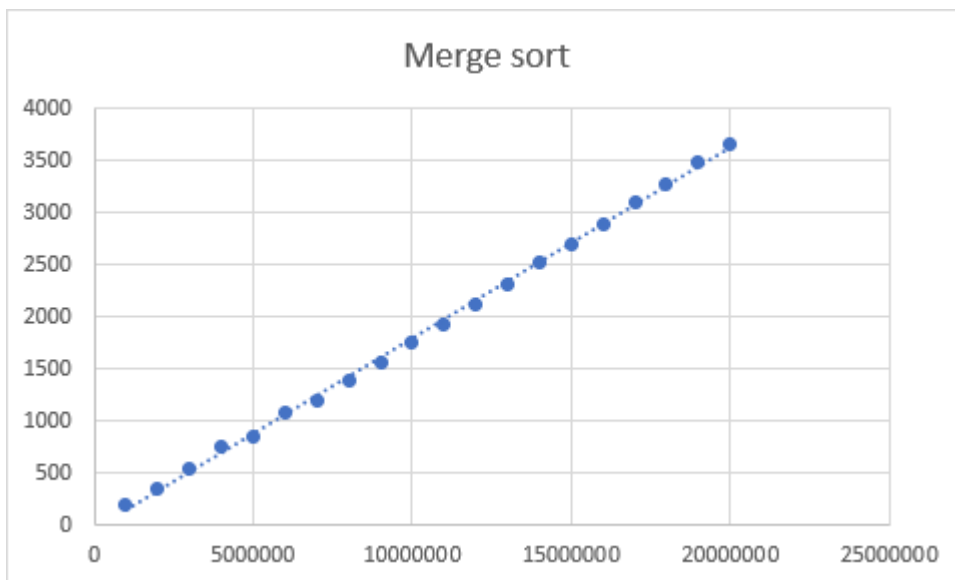
PROFESSOR MAURICIO TORO BERMÚDEZ
Phone: (+57) (4) 261 95 00 Ext. 9473. Office: 19 - 627
E-mail: mtorobe@eafit.edu.co

TIME	ARRAY LENGTH
188	1000000
343	2000000
541	3000000
760	4000000
847	5000000
1073	6000000
1202	7000000
1382	8000000
1560	9000000
1755	10000000
1934	11000000
2122	12000000
2322	13000000
2533	14000000
2700	15000000
2895	16000000
3102	17000000
3272	18000000
3482	19000000
3654	20000000

2. Insertion sort



Merge sort



3. Merge sort is faster than insertion sort, because mergeSort's complexity is $O(n)$ and insertionSort's complexity is $O(n^2)$. That means that use insertion sort for one million length array will take about 31 years to sort the array.
4. MaxSpan problem defines a "span" as the quantity of numbers between two equal numbers, these ones included. The aim is to find the max span. In order to solve it, we use two loops and a variable called

“maxSpan” initialized to 0(it will be used as an auxiliary variable). The first loop is used to choose one element of the array and we also initialize a variable called “sum” with 0. Into the internal loop, it compares the number in the position selected by the external-loop’s index with the others elements in the array determined by the second-loop’s index. At the same time the sum variable is increased one by one, until find two equal elements. When two equal elements have been found, sum variable becomes into a span and then it compares if that new span is bigger than current maxSpan variable. If it is bigger, maxSpan variable gets sum’s value, if it is not, sum’s value does not matter and gets useless. After that, the loop begins again with another number of the array, and so on until all the array numbers are use in the external loop.

5. Array2:**countEvens**

$$T(n) = c1 * n + c2 * n$$

$$T(n) = O(n)$$

bigDiff

$$T(n) = c1 * n + c2 * n$$

$$T(n) = O(n)$$

centeredAverage

$$T(n) = c1 * n + c2 * n + c3 * n$$

$$T(n) = O(n)$$

sum13

$$T(n) = c1 * n + c2 * n$$

$$T(n) = O(n)$$

sum67

$$T(n) = c1 * n$$

$$T(n) = O(n)$$

Array 3:**maxSpan**

$$T(n) = c1 * (n - 1) * n + c2 * (n - 1) * n$$

$$T(n) = O(n^2)$$

fix34

$$T(n) = c1 * (n - 1) * n + c2 * (n - 1) * n$$

$$T(n) = O(n^2)$$

fix45

$$T(n) = c1 * (n - 1) * n + c2 * (n - 1) * n$$

$$T(n) = O(n^2)$$

canBalance

$$T(n) = c1 * (n - 1) * n$$

$$T(n) = O(n^2)$$

linearIn

$$T(n) = c1 * (n - 1) * n$$

$$T(n) = O(n^2)$$

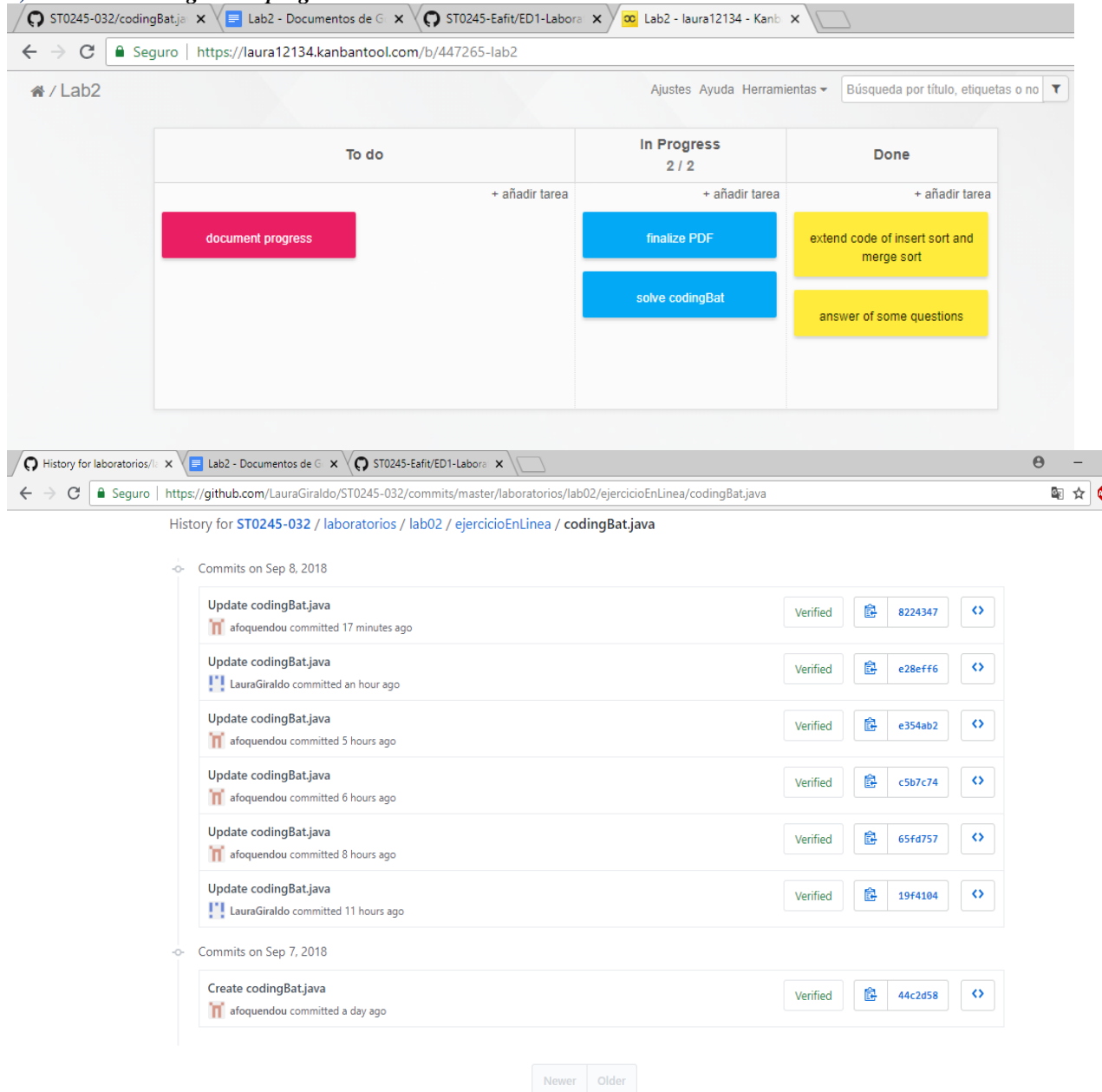
6. n represents one variable of which depends on the algorithm and with it we calculate the complexity. Thanks to that we know the execution time.

4) Practice for midterms

1. c
2. d
3. b
4. a
5. d
6. a
7. $t(n) = c + (n - 1)$, $O(n)$
8. b
9. d
10. b

- 11. c
- 12. b
- 13. a

5) Team work and gradual progress



The image shows two screenshots. The top screenshot is a Kanban board for 'Lab2' on the website <https://laura12134.kanbantool.com/b/447265-lab2>. It has three columns: 'To do', 'In Progress' (2 / 2), and 'Done'. The 'To do' column has a pink card 'document progress'. The 'In Progress' column has two blue cards: 'finalize PDF' and 'solve codingBat'. The 'Done' column has two yellow cards: 'extend code of insert sort and merge sort' and 'answer of some questions'. The bottom screenshot is a GitHub commit history for the repository `ST0245-032 / laboratorios / lab02 / ejercicioEnLinea / codingBat.java`. It shows commits from September 7, 2018, and September 8, 2018. The commits are: 'Create codingBat.java' (afoquendou, 1 day ago), and six 'Update codingBat.java' commits (afoquendou and LauraGiraldo, ranging from 17 minutes ago to 11 hours ago).

Column	Task
To do	document progress
In Progress	finalize PDF
In Progress	solve codingBat
Done	extend code of insert sort and merge sort
Done	answer of some questions

Commit Message	Author	Time Ago	SHA-1
Create codingBat.java	afoquendou	committed a day ago	44c2d58
Update codingBat.java	afoquendou	committed 17 minutes ago	8224347
Update codingBat.java	LauraGiraldo	committed an hour ago	e28eff6
Update codingBat.java	afoquendou	committed 5 hours ago	e354ab2
Update codingBat.java	afoquendou	committed 6 hours ago	c5b7c74
Update codingBat.java	afoquendou	committed 8 hours ago	65fd757
Update codingBat.java	LauraGiraldo	committed 11 hours ago	19f4104



UNIVERSIDAD EAFIT
SCHOOL OF ENGINEERING
DEPARTMENT OF INFORMATICS AND SYSTEMS

Code: ST245
Data structures
1

History for laboratorios/... x Lab2 - Documentos de G... x ST0245-Eafit/ED1-Labor... x Traductor de Google x

Seguro | <https://docs.google.com/document/d/1w8uWQ5mD5Xu-t3yQ8RjYRp-suLaH9jUKjp6NrYOYnJ4/edit?ts=5b943618>

Hoy, 22:47

100% Total: 11 cambios

Historial de versiones

Mostrar solo las versiones con nombre

Hoy

8 de septiembre, 22:47
Versión actual
Laura Maria Giraldo Castrillon
Andrés Felipe Oquendo

8 de septiembre, 17:36
Andrés Felipe Oquendo

8 de septiembre, 15:48
Andrés Felipe Oquendo
Archivo .doc importado - Ver el original

Mostrar cambios

UNIVERSIDAD EAFIT
SCHOOL OF ENGINEERING
DEPARTMENT OF INFORMATICS AND SYSTEMS

Code: ST245
Data structures
1

Laboratory practice No. 2: Big O Notation

Laura Maria Giraldo Castrillon
Universidad Eafit
Medellin, Colombia
lmgiraldol@eafit.edu.co

Andrés Felipe Oquendo Usma
Universidad Eafit
Medellin, Colombia
afoquendou@gmail.com

3) Practice for final project defense presentation

1. Insertion sort

TIME	ARRAY LENGTH
73	10000
283	20000
552	30000

10:48 p. m.
8/09/2018

PROFESSOR MAURICIO TORO BERMÚDEZ
Phone: (+57) (4) 261 95 00 Ext. 9473. Office: 19 - 627
E-mail: mtorobe@eafit.edu.co