Algorithmics	Student information	Date	Number of session
	UO: 275725		1.1
	Surname: Gómez Menéndez		
	Name: Laura		

Escuela de Ingeniería Informática
Universidad de Oviedo

Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Activity 1. [MEASURING EXECUTION TIMES]

Task 1 How many more years can we continue using this way of counting?

The biggest number we can use with the format long in java is (2^63)-1:

9.223.372.036.854.775.807

System.currentTimeMillis() shows how many milliseconds have passes since 1st of January of 1970.

So I think that we will be able to use this method until 292471208,67753601074 years from 1970.

Task 2 What does it mean that the time measured is 0?

It means that it took a really small amount of time so we can depreciate it.

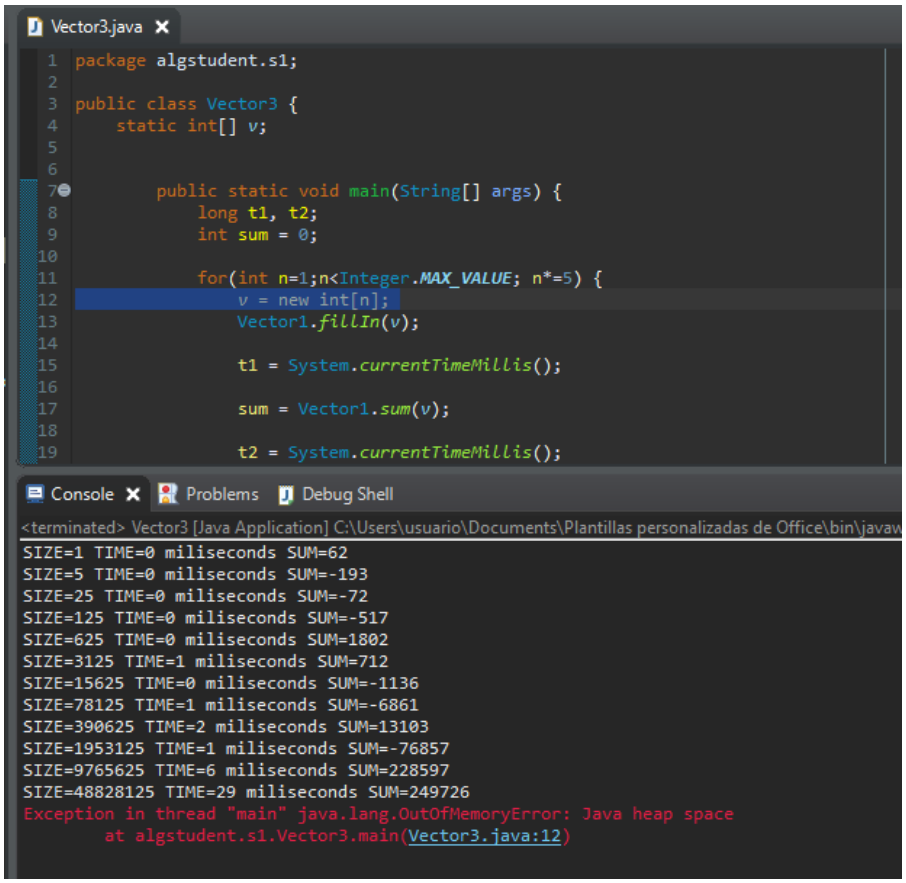
Task 3 From what size of problems (n) do we start to get reliable times?

50

Algorithmics	Student information	Date	Number of session
	UO: 275725		1.1
	Surname: Gómez Menéndez		
	Name: Laura		

Activity 2. [GROW OF THE PROBLEM SIZE]

Task 1 What happens with the time if the size of the problem is multiplied by 5?



The max size we can achieve is 48828125.

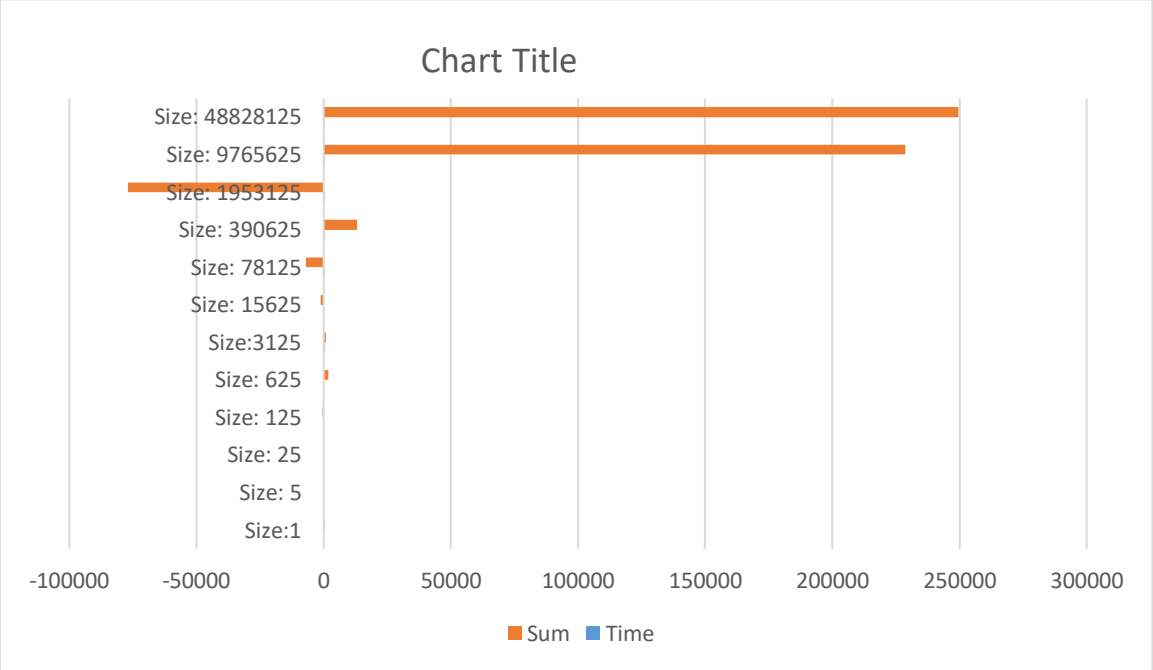
Task 2 Are the times obtained those that were expected from linear complexity O(n)?

No, because we can not trust on them due to the fact that are really small times.

Algorithmics	Student information	Date	Number of session
	UO: 275725		1.1
	Surname: Gómez Menéndez		
	Name: Laura		

Task 3 Use a spreadsheet to draw a graph with Excel. On the X axis we can put the time and on the Y axis the size of the problem.

Size	Time (ms)	Sum
1	0	62
5	0	-193
25	0	-72
125	0	-517
625	0	1802
3125	1	712
15625	0	-1136
78125	1	-6861
390625	2	13103
1953125	1	-76857
9765625	6	228597
48828125	29	249726



Algorithmics	Student information	Date	Number of session
	UO: 275725		1.1
	Surname: Gómez Menéndez		
	Name: Laura		

Activity 3. [TAKING SMALL EXECUTION TIMES]

As the argument that I used is 100000 the results are in hundreds of micros.

N	fillIn(t)	Sum(t)	Maximum(t)
10	23	0	0
30	7	0	0
90	20	0	0
270	56	0	0
810	165	0	0
2430	494	16	31
7290	1492	38	63
21870	5031	131	122
65610	14539	332	386
196830	40272	1087	1051
590490	120169	3900	3129
1771470	397341	16361	11457
5314410		47699	33109

- **What are the main components of the computer in which you did the work(process, memory…)**

  Process.

Algorithmics	Student information	Date	Number of session
	UO: 275725		1.1
	Surname: Gómez Menéndez		
	Name: Laura		

- **Do the values obtained meet the expectations? For that, you should calculate and indicate the theoretical values(a couple of examples per column) of the time complexity. Explain briefly the results.**

FillIn(), sum() and maximum() have a complexity of O(n).

FillIn()

If t1 = 1492, n1 = 7290 and n2 = 21870

K = n2/n1 = 21870/7290 = 3

T2= k * t1 = 3*1492 = 4476 is the expected value for t2 and the one I measured is 5031. There is a bit difference between both but it is not big.

If t1 = 5031, n1 = 21870 and n2 = 65610

K = n2/n1 = 65610/21870= 3

T2= k * t1 = 3*5031 = 15093 is the expected value for t2 and the one I measured is 14539. There is also a bit difference between both but it is not too big.

Sum()

If t1 = 131, n1 = 21870 and n2 = 65610

K = n2/n1 = 65610/21870= 3

T2= k * t1 = 3*131 = 393 is the expected value for t2 and the one I measured is 332. There is a small difference between both

If t1 = 332, n1 = 65610 and n2 = 196830

K = n2/n1 = 65610/21870= 3

T2= k * t1 = 3*332 = 996 is the expected value for t2 and the one I measured is 1087. There is also a bit difference between both but it is not too big.

Algorithmics	Student information	Date	Number of session
	UO: 275725		1.1
	Surname: Gómez Menéndez		
	Name: Laura		

Maximum()

If t1 = 122, n1 = 21870 and n2 = 65610

K = n2/n1 = 65610/21870= 3

T2= k * t1 = 3*122 = 366 is the expected value for t2 and the one I measured is 386.

There is practically no difference between both

If t1 = 386, n1 = 65610 and n2 = 196830

K = n2/n1 = 65610/21870= 3

T2= k * t1 = 3*386 = 1158 is the expected value for t2 and the one I measured is 1051.

There is also a bit difference between both but it is not too big.

Algorithmics	Student information	Date	Number of session
	UO: 275725		1.1
	Surname: Gómez Menéndez		
	Name: Laura		

Activity 4. [OPERATIONS ON MATRICES]

As the argument that I used is 100000 the results are in hundreds of micros.

N	sumDiagonal(t)	sumDiagonal2(t)
10	12	
30	21	
90	110	
270	890	
810	8410	
2430	71712	
7290	629612	

- **What are the main components of the computer in which you did the work(process, memory..)**

  Process.

- **Do the values obtained meet the expectations? For that, you should calculate and indicate the theoretical values(a couple of examples per column) of the time complexity. Explain briefly the results.**

The complexity of the method is O(n^2)

If t1 = 890, n1 = 270 and n2 = 810

K = n2/n1 = 810/270= 3

T2= k^2 * t1 = (3^2)*890 = 8010 is the expected value for t2 and the one I measured is 8410. There is a not so big difference between both.

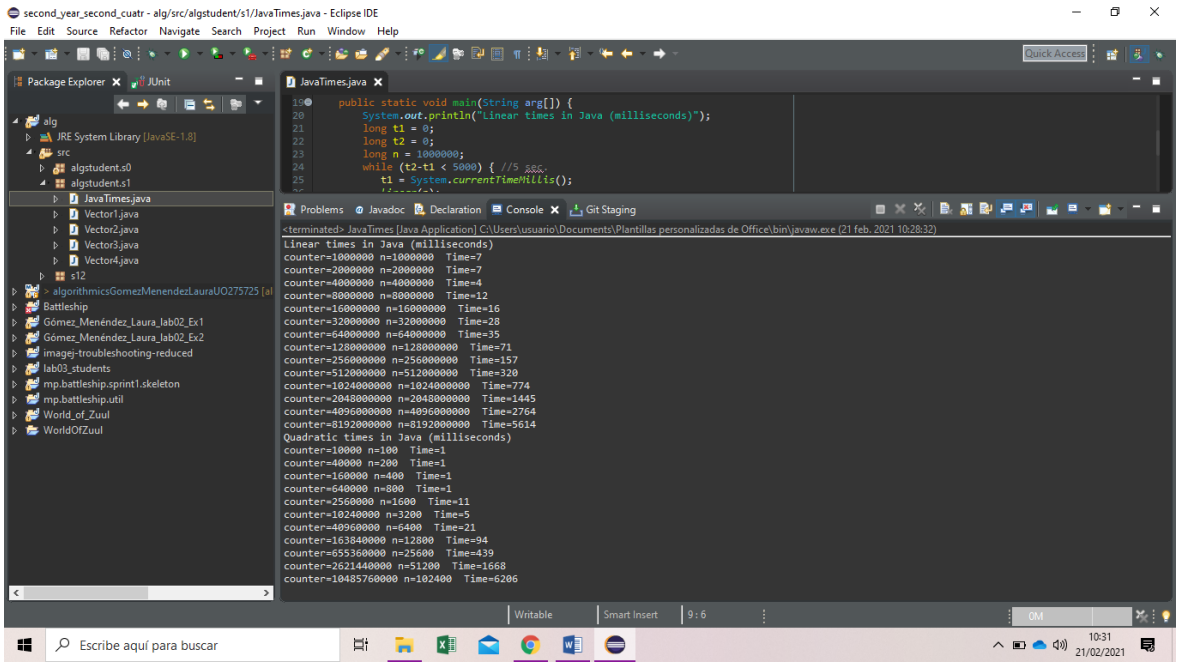
If t1 = 8410, n1 = 810 and n2 = 2730

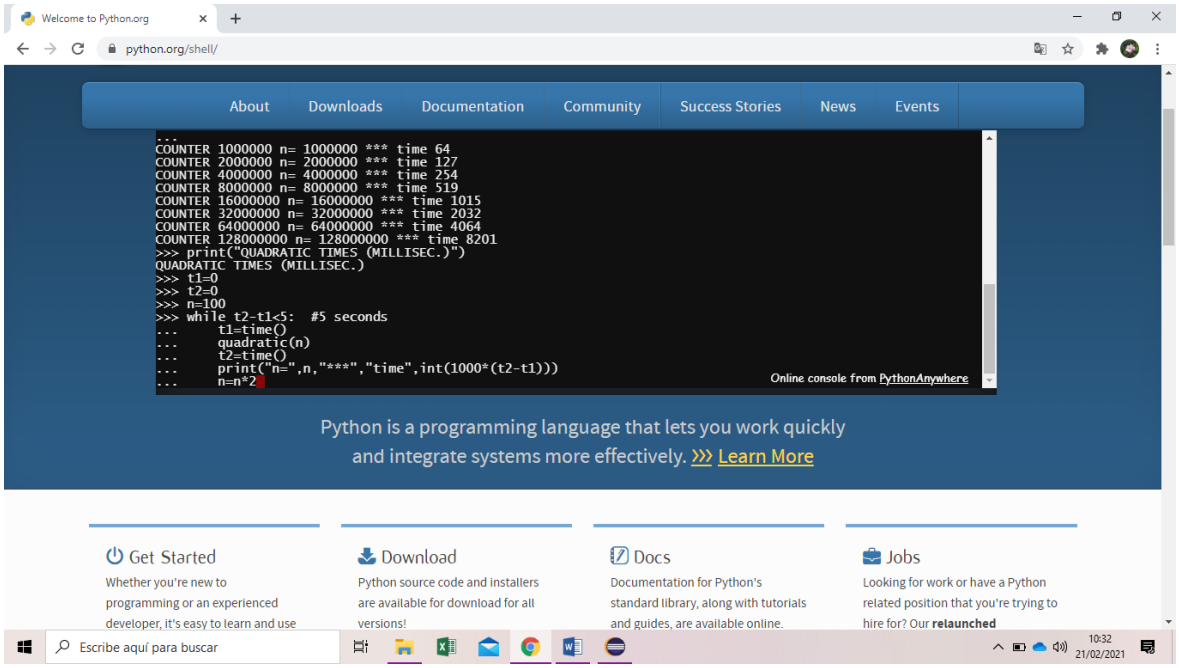
K = n2/n1 = 2730/810= 3

T2= k^2 * t1 = (3^2)*2730 = 24570 is the expected value for t2 and the one I measured is 71712. The difference is too big so it might be a problem or maybe when it was measured the PC was doing another thing.

Algorithmics	Student information	Date	Number of session
	UO: 275725		1.1
	Surname: Gómez Menéndez		
	Name: Laura		

Activity 5. [BENCHMARKING]





Algorithmics	Student information	Date	Number of session
	UO: 275725		1.1
	Surname: Gómez Menéndez		
	Name: Laura		

Task 1 Why you get two differences in execution time between the two programs?

Because the main components that each one is using are different.

Task 2 Regardless of the specific times. Is there any analogy in the behavior of the two implementations)