

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



## 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 1<sup>st</sup> 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?

```

Vector3.java x
1 package algstudent.s1;
2
3 public class Vector3 {
4     static int[] v;
5
6
7     public static void main(String[] args) {
8         long t1, t2;
9         int sum = 0;
10
11         for(int n=1;n<Integer.MAX_VALUE; n*=5) {
12             v = new int[n];
13             Vector1.fillIn(v);
14
15             t1 = System.currentTimeMillis();
16
17             sum = Vector1.sum(v);
18
19             t2 = System.currentTimeMillis();
20
21         }
22     }
23 }

```

```

<terminated> Vector3 [Java Application] C:\Users\usuario\Documents\Plantillas personalizadas de Office\bin\javaw
SIZE=1 TIME=0 milliseconds SUM=62
SIZE=5 TIME=0 milliseconds SUM=-193
SIZE=25 TIME=0 milliseconds SUM=-72
SIZE=125 TIME=0 milliseconds SUM=-517
SIZE=625 TIME=0 milliseconds SUM=1802
SIZE=3125 TIME=1 milliseconds SUM=712
SIZE=15625 TIME=0 milliseconds SUM=-1136
SIZE=78125 TIME=1 milliseconds SUM=-6861
SIZE=390625 TIME=2 milliseconds SUM=13103
SIZE=1953125 TIME=1 milliseconds SUM=-76857
SIZE=9765625 TIME=6 milliseconds SUM=228597
SIZE=48828125 TIME=29 milliseconds SUM=249726
Exception in thread "main" java.lang.OutOfMemoryError: Java heap space
    at algstudent.s1.Vector3.main(Vector3.java:12)

```

The max size we can achieve is 48828125.

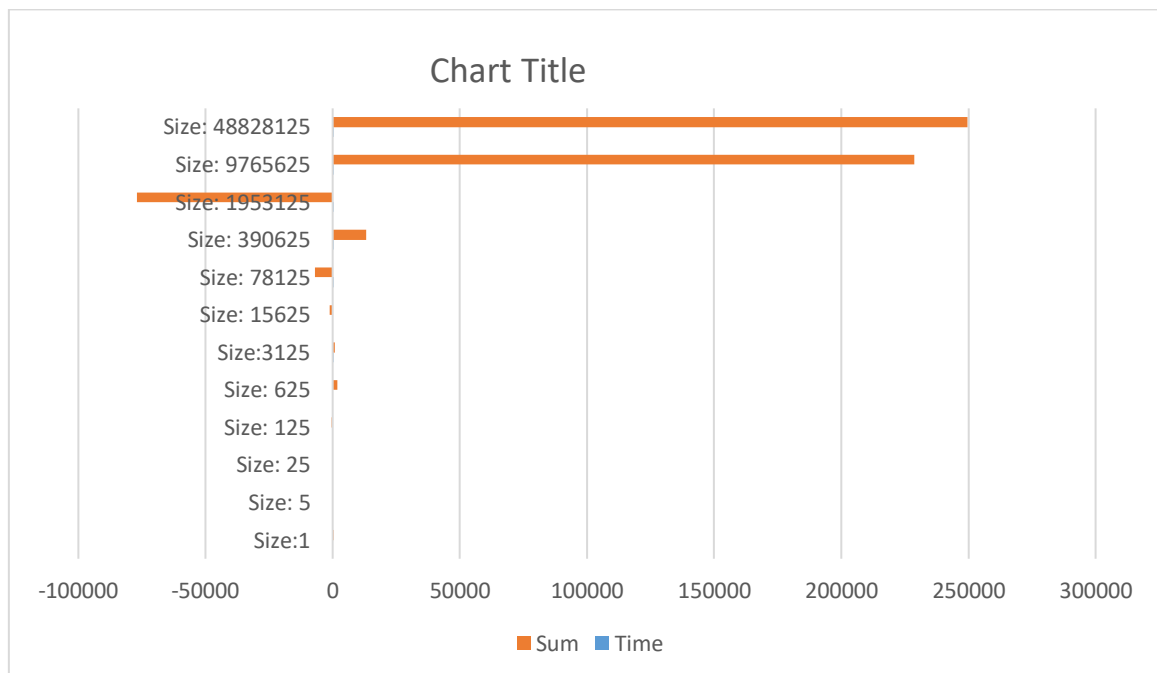
**Task 2** Are the times obtained those that were expected from linear complexity  $O(n)$ ?

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  $t_1 = 1492$ ,  $n_1 = 7290$  and  $n_2 = 21870$

$K = n_2/n_1 = 21870/7290 = 3$

$T_2 = k * t_1 = 3 * 1492 = 4476$  is the expected value for  $t_2$  and the one I measured is 5031.

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

If  $t_1 = 5031$ ,  $n_1 = 21870$  and  $n_2 = 65610$

$K = n_2/n_1 = 65610/21870 = 3$

$T_2 = k * t_1 = 3 * 5031 = 15093$  is the expected value for  $t_2$  and the one I measured is 14539. There is also a bit difference between both but it is not too big.

Sum()

If  $t_1 = 131$ ,  $n_1 = 21870$  and  $n_2 = 65610$

$K = n_2/n_1 = 65610/21870 = 3$

$T_2 = k * t_1 = 3 * 131 = 393$  is the expected value for  $t_2$  and the one I measured is 332.

There is a small difference between both

If  $t_1 = 332$ ,  $n_1 = 65610$  and  $n_2 = 196830$

$K = n_2/n_1 = 196830/65610 = 3$

$T_2 = k * t_1 = 3 * 332 = 996$  is the expected value for  $t_2$  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  $t_1 = 122$ ,  $n_1 = 21870$  and  $n_2 = 65610$

$K = n_2/n_1 = 65610/21870 = 3$

$T_2 = k * t_1 = 3 * 122 = 366$  is the expected value for  $t_2$  and the one I measured is 386.

There is practically no difference between both

If  $t_1 = 386$ ,  $n_1 = 65610$  and  $n_2 = 196830$

$K = n_2/n_1 = 196830/65610 = 3$

$T_2 = k * t_1 = 3 * 386 = 1158$  is the expected value for  $t_2$  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  $t_1 = 890$ ,  $n_1 = 270$  and  $n_2 = 810$

$$K = n_2/n_1 = 810/270 = 3$$

$T_2 = k^2 * t_1 = (3^2) * 890 = 8010$  is the expected value for  $t_2$  and the one I measured is 8410. There is a not so big difference between both.

If  $t_1 = 8410$ ,  $n_1 = 810$  and  $n_2 = 2730$

$$K = n_2/n_1 = 2730/810 = 3$$

$T_2 = k^2 * t_1 = (3^2) * 8410 = 24570$  is the expected value for  $t_2$  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]

The screenshot shows the Eclipse IDE with a Java file named `JavaTimes.java`. The code measures the execution time of linear and quadratic operations for various input sizes `n`. The console output shows the following results:

Operation	n	Time (ms)
Linear	1000000	7
Linear	2000000	7
Linear	4000000	4
Linear	8000000	12
Linear	16000000	16
Linear	32000000	28
Linear	64000000	35
Linear	128000000	71
Linear	256000000	157
Linear	512000000	320
Linear	1024000000	774
Linear	2048000000	1445
Linear	4096000000	2764
Linear	8192000000	5614
Quadratic	10000	1
Quadratic	40000	1
Quadratic	160000	1
Quadratic	640000	1
Quadratic	2560000	11
Quadratic	10240000	5
Quadratic	40960000	21
Quadratic	163840000	94
Quadratic	655360000	439
Quadratic	2621440000	1658
Quadratic	10485760000	6286

The screenshot shows the Python.org website with a terminal window displaying benchmarking results for linear and quadratic operations. The results are as follows:

Operation	n	Time (ms)
Linear	1000000	64
Linear	2000000	127
Linear	4000000	254
Linear	8000000	519
Linear	16000000	1015
Linear	32000000	2032
Linear	64000000	4064
Linear	128000000	8201
Quadratic	10000	1
Quadratic	40000	1
Quadratic	160000	1
Quadratic	640000	11
Quadratic	2560000	439
Quadratic	10240000	1658
Quadratic	40960000	6286



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)**