Algorithmics	Student information	Date	Number of session
	UO: 258220	11-02-21	1.1
	Surname: Cuesta Martínez		
	Name: Miguel		

Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Activity 1. Measuring execution times

1. By means of the use of Long.MAX_VALUE we obtain the exact moment:

```
Sun Aug 17 03:12:55 GMT-04:00 292278994
```

2. Measured time 0 means the value taken was below the millisecond, or somehow else ignored by the computer as not relevant

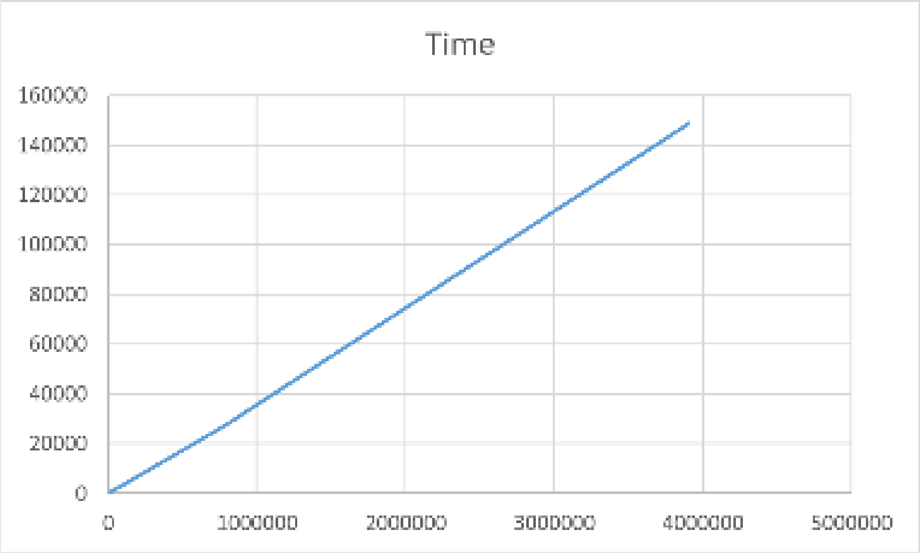
3. From n = 156250

Activity 2. Growth of the problem size

1. The time is also multiplied by 5

2. Yes, as O(n) means the relation between time and size is direct, that is, F(t) = n

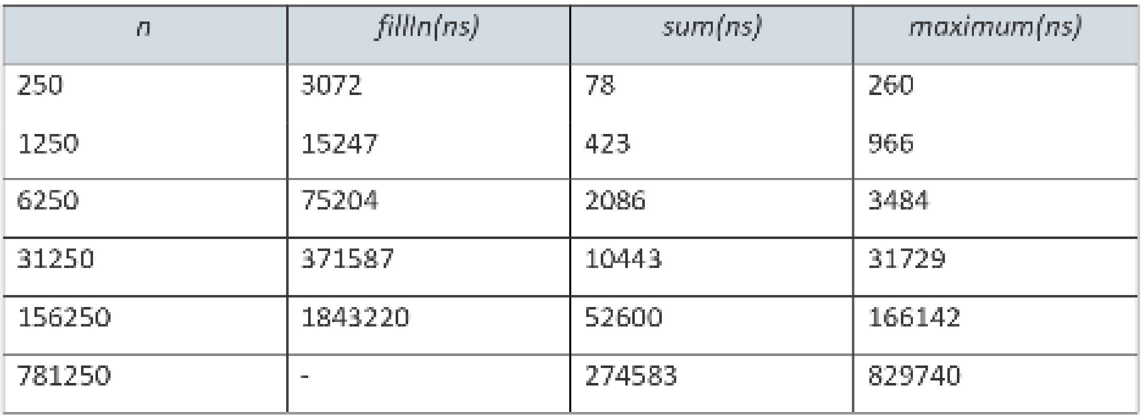
3. We obtain the following graph:



Algorithmics	Student information	Date	Number of session
	UO: 258220	11-02-21	1.1
	Surname: Cuesta Martínez		
	Name: Miguel		

Activity 3. Taking small execution times

Values under 50 have been eliminated:



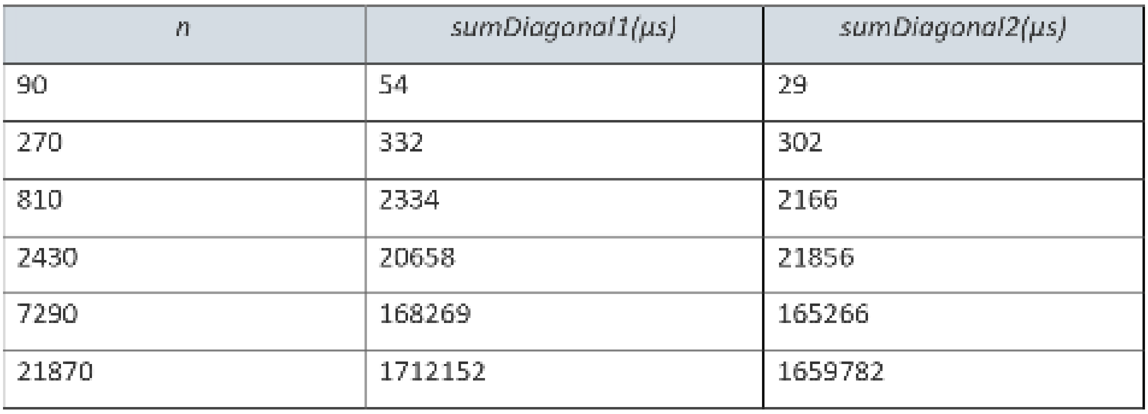
n	*fillIn(ns)*	*sum(ns)*	*maximum(ns)*
250	3072	78	260
1250	15247	423	966
6250	75204	2086	3484
31250	371587	10443	31729
156250	1843220	52600	166142
781250	-	274583	829740

While the main work is on the processor, for higher values it's the memory that becomes a problem. The values obtained meet the expectations:

$$\frac{t1}{n1} = \frac{t2}{n2} \; ; \; t2 = n2 * \frac{t1}{n1}$$

Activity 4. Operations on matrices

Values under 50 have been eliminated:



n	*sumDiagonal1(µs)*	*sumDiagonal2(µs)*
90	54	29
270	332	302
810	2334	2166
2430	20658	21856
7290	168269	165266
21870	1712152	1659782

The main work was on the processor for the whole process. The obtained values meet the expectations:

Algorithmics	Student information	Date	Number of session
	UO: 258220	11-02-21	1.1
	Surname: Cuesta Martínez		
	Name: Miguel		

$$\frac{n1^2}{t1} = \frac{n2^2}{t2}\;;\; t2 = t1\frac{n2^2}{n1^2}$$

Activity 5. Benchmarking

We get different results, due to how the machine compiles different code. Javac is faster than Python’s in this regard. Both versions follow the same trend though, so the different would be in the constant applied over the complexity, that they share.