Algorithmics	Student information	Date	Number of session
	UO: 300895	7-2-25	2
	Surname: Franco Martinez		
	Name: Ivan		

Escuela de Ingeniería Informática
Universidad de Oviedo

Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Activity 1. [Measuring execution times-1]

Calculate how many more years we can continue using this way of counting. Explain what you did to calculate it.

The currentTimeMillis() method returns the number of milliseconds since January 1, 1970, stored as a 64-bit signed integer. The maximum value of a 64-bit signed integer is 2^63 - 1.

To find out how many more years we can use this method:

- Current value: As of now, it has been around 55 years since 1970 ≈ 1.73 × 10^12 milliseconds.
- Maximum value: 2^63 - 1 ≈ 9.22 × 10^18 milliseconds.
- Remaining time: (9.22 × 10^18 - 1.73 × 10^12) milliseconds.

Converting this to years ≈ 292 million years.

Activity 2. [Measuring execution times-2]

Why does the measured time sometimes come out as 0?

Because the n of repetitions is too small and theoperation takes less than 1 millisecond, the start and end

From what size of problem (n) do we start to get reliable times?

Approximately from `12910000`

Activity 3. [Taking small execution times]

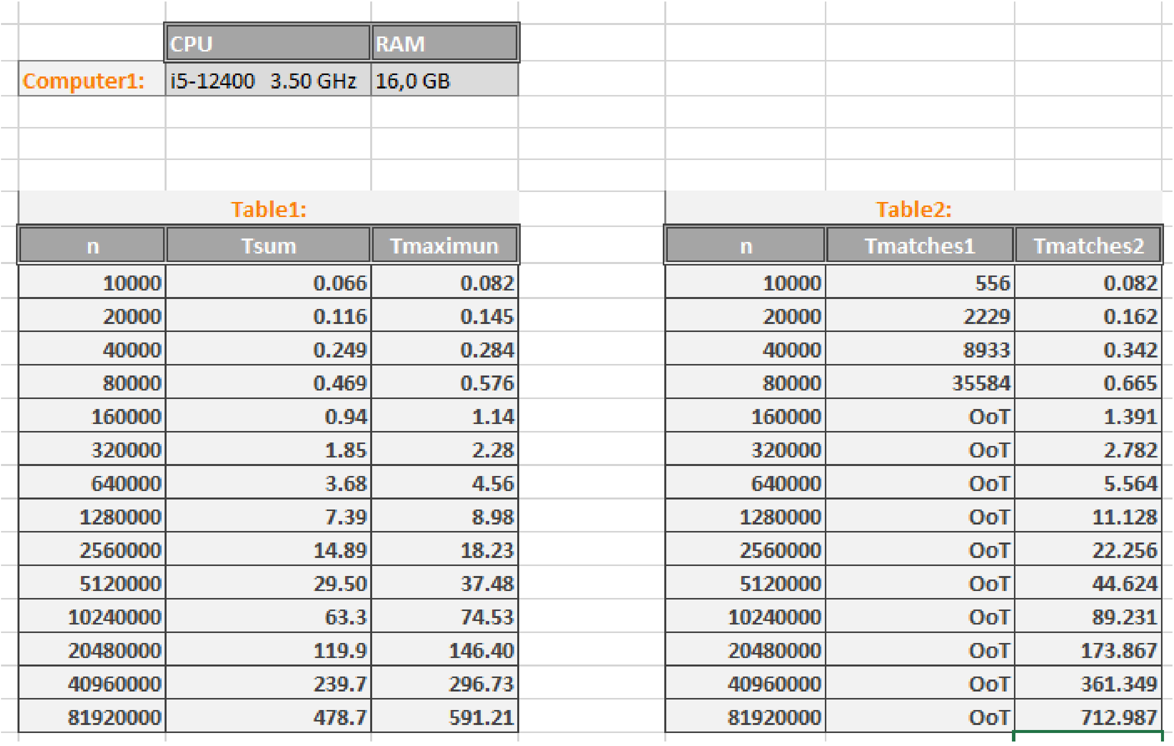
What happens with time if the problem size is multiplied by 2?

The times get multiplied by 2 also

Algorithmics	Student information	Date	Number of session
	UO: 300895	7-2-25	2
	Surname: Franco Martinez		
	Name: Ivan		

What happens with time if the problem size is multiplied by a value k other than 2? (try it, for example, for k=3 and k=4 and check the times obtained)

The times get multiplied by the constant k also



	CPU	RAM
Computer1:	i5-12400 3.50 GHz	16,0 GB

Table1:

n	Tsum	Tmaximun
10000	0.066	0.082
20000	0.116	0.145
40000	0.249	0.284
80000	0.469	0.576
160000	0.94	1.14
320000	1.85	2.28
640000	3.68	4.56
1280000	7.39	8.98
2560000	14.89	18.23
5120000	29.50	37.48
10240000	63.3	74.53
20480000	119.9	146.40
40960000	239.7	296.73
81920000	478.7	591.21

Table2:

n	Tmatches1	Tmatches2
10000	556	0.082
20000	2229	0.162
40000	8933	0.342
80000	35584	0.665
160000	OoT	1.391
320000	OoT	2.782
640000	OoT	5.564
1280000	OoT	11.128
2560000	OoT	22.256
5120000	OoT	44.624
10240000	OoT	89.231
20480000	OoT	173.867
40960000	OoT	361.349
81920000	OoT	712.987