Algorithmics	Student information	Date	Number of session
	UO:277653	18/02/2021	3
	Surname: Stanci		
	Name: Stelian Adrian		

Escuela de Ingeniería Informática
Universidad de Oviedo

Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Activity 1. Two algorithms with the same complexity

CPU: Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz 2.59 GHz

RAM: 16GB

N	loop2(t) 10^{-5}s	loop3(t) 10^{-5}s	loop2(t)/loop3(t)
8	1	1	1
16	1	1	1
32	3	2	1,50
64	14	9	1,56
128	32	25	1,28
256	128	70	1,83
512	388	274	1,42
1024	1478	778	1,90
2048	5515	2642	2,09
4096	21338	10325	2,07

In order to decide if the results make sense we have to make use of the following formula:

$$t2 = \frac{f(n2)}{f(n1)} * t1$$

Taking into account that the complexities of the algorithms are quadratic in both cases, the division would result in either 1, less than 1 or more than 1, depending on the constants. Applying the formula for loop2 with n1 = 2048, t1 = 5515, n2 = 4096 we get that t2 is equal to 22060, which is close to the value we obtained.

We repeat the same for loop3, with n1 = 2048, t1 = 2642 and n2 = 4096. In this case t2 = 10568, which is also close to what we obtained, 10325

Algorithmics	Student information	Date	Number of session
	UO:277653	18/02/2021	3
	Surname: Stanci		
	Name: Stelian Adrian		

Activity 2. Two algorithms with different complexity

CPU: Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz   2.59 GHz

RAM: 16GB

N	Loop1(t) 10^{-5}s	loop2(t) 10^{-5}s	Loop1(t)/loop2(t)
8	1	1	1,00
16	0	1	0,00
32	1	3	0,33
64	1	14	0,07
128	3	32	0,09
256	8	128	0,06
512	11	388	0,03
1024	25	1478	0,02
2048	51	5515	0,01
4096	129	21338	0,01

In order to decide if the results make sense we have to make use of the following formula:

$$t2 = \frac{f(n2)}{f(n1)} * t1$$

The algorithm in loop1 has a complexity of O(nlogn). Taking that into consideration we can apply the formula stated with the values n1 = 2048, t1 = 51, n2 = 4096, and we obtain that t2=126, which is very close to the empirically obtained value, 129, so we can conclude that the value makes sense.

If we do the same analysis for the algorithm in loop 2, with the same values, we obtain that t2=22060, which is near the obtained value as well.

Note that, as the complexity of loop1 is better than the one of loop2, the division ratio tends to 0 as we increase the n.

Activity 3. Complexity of other algorithms

CPU: Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz 2.59 GHz

RAM: 16GB

N	Loop4(t) 10^{-3}s	Loop5(t) 10^{-3}s	Loop4(t)/loop5(t)
8	1	1	1,00
16	4	0	-
32	23	4	5,75
64	309	17	18,18
128	3785	137	27,63
256	59004	1157	51,00
512	911401	10345	88,10
1024			
2048			
4096			

In order to decide if the results make sense we have to make use of the following formula:

$$t2 = \frac{f(n2)}{f(n1)} * t1$$

The algorithm in loop4 has a complexity of $O(n^4)$. Taking that into consideration we can apply the formula stated with the values n1 = 256, t1 = 59004, n2 = 512, and we obtain that t2=944064, which is close to the empirically obtained value, 911401, so we can conclude that the value makes sense.

We do the same for loop5, with the values n1 = 256, t1 = 1157 and n2 = 512, but this time taking into account that the complexity is O(n^3logn). The value we obtain for t2 is 10413, which is very close to the obtained value, 10345, so we conclude that this value also makes sense.

Activity 4. Study of `Unknown.java`

N	Execution time (in 10^{-5}s)
8	1
16	1
32	3
64	3
128	5
256	32
512	251
1024	1193
2048	7095
4096	52015

The complexity of the algorithm is O(n^3) but taking into account that the last two loops iterate through the previous variable instead of n the computational cost is reduced, we may take this into consideration when computing the theoretical values. In order to see if the results make sense, we are going to make use of the following formula:

$$t2 = \frac{f(n2)}{f(n1)} * t1$$

With t1 = 7095, n1 = 2048, n2 = 4096. And we obtain that t2 = 56760, which approximates to the value we obtained, 52015, so we can conclude that the result does make sense.

N	Execution time (in 10^{-5}s)
8	0
16	0
32	3
64	2
128	5
256	33
512	250
1024	1239
2048	7752
4096	50621

We repeat the operation again, this time with t1 = 7752, n1 = 2048 and n2 = 4096. We obtain that t2 = 62016. This result is a bit far from the one we obtained, 50621, but that is due to the variables used in the loops, which divide the complexity by a constant.