Algorithmics	Student information	Date	Number of session
	UO: 270534	2/17/2021	3
	Surname: Iglesias Préstamo		
	Name: Ángel		

Escuela de Ingeniería Informática
Universidad de Oviedo

Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Activity 1. [Two algorithms with the same complexity]

Hardware.

- **CPU:** Intel Core I7 4770K @ 4.0 GHz.
- **RAM:** 2x 8 GB Kingston Hyper X DDR3 1600 MHz == 16 GB.

Benchmarks.

N	*loop2(10^{-8}s)*	*loop3(10^{-8}s)*	*loop2(10^{-8}s) / loop3(10^{-8}s)*
1	11	11	1
2	11	7	1.57
4	19	14	1.36
8	66	38	1.74
16	248	133	1.86
32	989	399	2.49
64	3,862	1,954	1.97
128	15,467	7,785	1.99
256	61,859	30,863	2.00
512	246,609	123,609	1.99
1,024	984,576	492,890	1.99
Complexity:	O(n^2)	O(n^2)	

Conclusion.

As you can notice, as the values get higher and higher, the division between both times remains constant: 2. This means: when we achieve enough precision that foreign variables will not make such an impact, we can clearly see that both times are increased evenly.

Activity 2. [Two algorithms with different complexity]

Hardware.

- **CPU:** Intel Core I7 4770K @ 4.0 GHz

- **RAM:** 2x 8 GB Kingston Hyper X DDR3 1600 MHz == 16 GB.

Benchmarks.

n	*loop1(10^{-8}s)*	*loop2(10^{-8}s)*	*loop1(10^{-8}s) / loop2(10^{-8}s)*
1	12	11	1.09
2	10	11	0.91
4	15	19	0.79
8	38	66	0.57
16	86	248	0.35
32	187	989	0.19
64	438	3,862	0.11
128	988	15,467	0.06
256	2,241	61,859	0.03
512	5,006	246,609	0.02
1,024	10,893	984,576	0.01
Complexity:	O(n)	O(n^2)	

Conclusion.

As you can notice the difference between these two algorithms keeps increasing as the value is getting doubled. More in more, the first algorithm has a complexity of O(n) while the second's complexity is quadratic.

Activity 3. [Complexity of other algorithms]

Hardware.

- **CPU:** Intel Core I7 4770K @ 4.0 GHz.
- **RAM:** 2x 8 GB Kingston Hyper X DDR3 1600 MHz == 16 GB.

Benchmarks.

n	*loop4(10^{-6}s)*	*loop5(10^{-6}s)*	*loop4(10^{-6}s) / loop5(10^{-6}s)*
1	2	2	1
2	1	1	1
4	1	0	-
8	4	4	1
16	37	20	1.85
32	493	188	2.62

n	*loop4(10^{-6}s)*	*loop5(10^{-6}s)*	*loop4(10^{-6}s) / loop5(10^{-6}s)*
64	7,235	1,786	4.05
128	110,524	16,712	6.61
256	1,789,067	156,454	11.43
Complexity:	$O(n^4)$	$O(n^3 \log n)$	

Conclusion.

Something relative to the thing happening before is occurring now, however, the pace of this comparison is still lower than the one seen before.

Activity 4. [Two algorithms with different complexity]

Hardware.

- **CPU:** Intel Core I7 4770K @ 4.0 GHz.
- **RAM:** 2x 8 GB Kingston Hyper X DDR3 1600 MHz == 16 GB.

Task 1.

n	*Unknown(t)*
1	1
2	0
4	0
8	1
16	1
32	4
64	23
128	157
256	1,061
512	7,297
1,024	52,687
2,048	396,377
Complexity:	$O(n^3)$

Task 2. Does it make sense according to the theoretical complexity?

Unknown(t)
$t_1 = 1{,}061, n_1 = 256\ and\ n_2 = 1{,}024$
$t_3 = 52{,}687, n_3 = 1{,}024\ and\ n_4 = 7{,}290$
• $t_2 = \left(\frac{1{,}024}{256}\right)^3 \cdot 1{,}061 = 67{,}904$
• $t_4 = \left(\frac{2{,}048}{1{,}024}\right)^3 \cdot 52{,}687 = 421{,}496$

Values	*sumDiagonal1(t)*
Actual_1	52,687
Expected_1	67,904
Actual_2	396,377
Expected_2	421,496

Even if these results may seem to be a bit awkward, not going directly to the point, notice that considering that some values are still low, some of them are being used for me to calculate the expected complexity, I'm quite happy with them.