Algorithmics	Student information	Date	Number of session
	UO: 300895	13-2-25	3
	Surname: Franco Martinez		
	Name: Ivan		

Escuela de Ingeniería Informática
Universidad de Oviedo

Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Activity 1. [Some iterative models-1]

Table1:

n	TLoop1	TLoop2	TLoop3	TLoop4
100	0.0067	0.166	0.85	0.7
200	0.0118	0.594	3.4	4.76
400	0.0302	2.708	14.21	35.88
800	0.0630	12.437	61.38	280
1600	0.1544	49.387	263	2171
3200	0.3033	221.2	1113	17435
6400	0.6928	887.44	4653	OoT
12800	1.49	4055	19968	OoT
25600	3.098	27729	OoT	OoT
51200	6.07	OoT	OoT	OoT

Complexities:

loop 1 = O(n*log(n))

loop2 = O(n^2*log(n))

loop3 = O(n^2*log(n))

loop4 = O(n^3)

Although Loops 2 and 3 have the same theoretical complexity nut their execution times differ because they are written in different ways. As for the relationship between the complexities, as the complexity of the loops increases, the execution times also increase as expected.

Activity 2. [Some iterative models-2]

Algorithmics	Student information	Date	Number of session
	UO: 300895	13-2-25	3
	Surname: Franco Martinez		
	Name: Ivan		

Table2:

n	TLoop5	TLoop6	TLoop7
100	5.5	68	474
200	24.4	538	7262
400	115.8	4654	OoT
800	561.2	39740	OoT
1600	2557.5	OoT	OoT
3200	11790	OoT	OoT
6400	53505	OoT	OoT
12800	OoT	OoT	OoT
25600	OoT	OoT	OoT
51200	OoT	OoT	OoT

Complexities:

Loop5 = O(n^2 log^2(n))

Loop6 = O(n^3 log(n))

Loop7 = O(n^4)

As the complexity increases, the execution times also increase, which is to be expected. Additionally, the times are greater than those in Table 1

Activity 3. [Comparison of two algorithms]

Table3:

n	TLoop1	TLoop2	t1/t2
100	0,00670	0,16600	0,0404
200	0,01180	0,59400	0,0199
400	0,03020	2,70800	0,0112
800	0,06300	12,43700	0,0051
1600	0,15440	49,38700	0,0031
3200	0,30330	221,20000	0,0014
6400	0,69280	887,44000	0,0008
12800	1,49000	4055,00000	0,0004
25600	3,09800	27729,00000	0,0001
51200	6,07000	OoT	OoT

As the complexities for loop1 is O(n*log(n)) and for loop2 is O(n^2*log(n)), the time of execution must be greater for the loop2 , so the results of the division (t1/t2) must be < 1, what matches with the results obtained.

Algorithmics	Student information	Date	Number of session
	UO: 300895	13-2-25	3
	Surname: Franco Martinez		
	Name: Ivan		

Activity 4. [Two algorithms with the same complexity]

Table4:

n	TLoop3	TLoop2	t3/t2
100	0,85	0,166	5,1205
200	3,4	0,594	5,7239
400	14,21	3	5,2474
800	61,38	12	4,9353
1600	263	49	5,3253
3200	1113	221,2	5,0316
6400	4653	887,44	5,2432
12800	19968	4055	4,9243
25600	OoT	27729	OoT
51200	OoT	OoT	OoT

Instead the both program have the same complexity , loop2 was wrote in a different way than loop3 this factors makes that loop2 is faster than loop3 what means that its division (t3/t2) will be >0, what matches the results optined

Activity 5. [Same algorithm in different development environments]

Table5:

n	tLoop4(Python)-t41	tLoop4(java without optimization)-t42	tLoop4(Java with optimization)-t43	t42/t41	t43/t42
100	2	0,9	0,07	0,4500	0,0778
200	26	5,54	0,48	0,2131	0,0859
400	199	38	3,288	0,1910	0,0865
800	1730	269	23,0	0,1555	0,0855
1600	13995	1926	167,0	0,1376	0,0867
3200	OoT	14826	1243,0	OoT	0,0838
6400	OoT	OoT	OoT	OoT	OoT

Instead we are running the same program, python it would be always slower than java and running java with the optimization command would be always faster than running without

it. That's the reasons of why the execution times are different so as t41 would be slower than t42 we will expect that the division (t42/t41) is < 0, and as t43 is faster that t42 we will expect the division(t43/t42) to be > 1. Both measurements coincide with the results