Activity 1. [Some iterative models]

Loop1: O(nlog(n))

Loop2: O(n2 log(n))

Loop3: O(n2log(n))

Loop4: O(n3)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N | tLoop1 | tLoop2 | tLoop3 | tLoop4 |
| 100 | 0,007 | 0,166 | 0,86 | 0,69 |
| 200 | 0,013 | 0,592 | 3,51 | 4,85 |
| 400 | 0,0302 | 2,692 | 14,76 | 36,51 |
| 800 | 0,0664 | 12,318 | 63,19 | 285 |
| 1600 | 0,1509 | 49,5 | 269 | 2219 |
| 3200 | 0,3187 | 221,6 | 1146 | 16749 |
| 6400 | 0,6935 | 868,4 | 4630 | Oot |
| 12800 | 1,5296 | 3948 | 19466 | Oot |
| 25600 | 3,16 | 17337 | Oot | Oot |
| 51200 | 6,342 | Oot | Oot | Oot |

Taking into account the theoretical complexities, the results obtained after the execution of the programs are the expected ones.

Activity 2. [Creation of iterative models]

Loop5: O(n2log^2(n))

Loop6: O(n3log(n))

Loop7: O(n4)

|  |  |  |  |
| --- | --- | --- | --- |
| N | tLoop5 | tLoop6 | tLoop7 |
| 100 | 4,09 | 57 | 394 |
| 200 | 19,5 | 451 | 6145 |
| 400 | 92,8 | 3899 | Oot |
| 800 | 432,5 | 32641 | Oot |
| 1600 | 2018,6 | Oot | Oot |
| 3200 | 9167 | Oot | Oot |
| 6400 | 41854 | Oot | Oot |
| 12800 | Oot | Oot | Oot |
| 25600 | Oot | Oot | Oot |
| 51200 | Oot | Oot | Oot |

As expected, the growth tendency of the complexities is reflected in the times obtained.

Activity 3. [Compare algorithms with different complexities]

|  |  |  |  |
| --- | --- | --- | --- |
| N | tLoop1 | tLoop2 | t1/t2 |
| 100 | 0,007 | 0,166 | 0,04216867 |
| 200 | 0,013 | 0,592 | 0,02195946 |
| 400 | 0,0302 | 2,692 | 0,01121842 |
| 800 | 0,0664 | 12,318 | 0,00539049 |
| 1600 | 0,1509 | 49,5 | 0,00304848 |
| 3200 | 0,3187 | 221,6 | 0,00143818 |
| 6400 | 0,6935 | 868,4 | 0,0007986 |
| 12800 | 1,5296 | 3948 | 0,00038744 |
| 25600 | 3,16 | 17337 | 0,00018227 |
| 51200 | 6,342 | Oot | #¡VALOR! |

As the complexity of the loop1 program is much better than the complexity of the loop2 program, the division ratio tends to 0 as “n” increases.

Activity 4. [Compare algorithms with same complexities]

|  |  |  |  |
| --- | --- | --- | --- |
| N | tLoop3 | tLoop2 | t3/t2 |
| 100 | 0,86 | 0,166 | 5,18072289 |
| 200 | 3,51 | 0,592 | 5,92905405 |
| 400 | 14,76 | 2,692 | 5,48291233 |
| 800 | 63,19 | 12,318 | 5,12989122 |
| 1600 | 269 | 49,5 | 5,43434343 |
| 3200 | 1146 | 221,6 | 5,17148014 |
| 6400 | 4630 | 868,4 | 5,3316444 |
| 12800 | 19466 | 3948 | 4,93059777 |
| 25600 | Oot | 17337 | #¡VALOR! |
| 51200 | Oot | Oot | #¡VALOR! |

Both complexities are not exactly the same because we are omitting the bases of the logarithms, this explains the variations of the division ratio. However, this “error” can be ignored because the division ratio tends to be constant, so this indicates that both complexities have the same growth tendency.

Activity 5. [Compare algorithms with same complexities, but different environments]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| N | tLoop4(python) (t41) | tLoop4(java without op) (t42) | tLoop4(java with op) (t43) | t42/t41 | t43/t42 |
| 100 | 4 | 0,69 | 0,088 | 0,1725 | 0,12753623 |
| 200 | 29 | 4,85 | 0,54 | 0,16724138 | 0,11134021 |
| 400 | 221 | 36,51 | 3,364 | 0,16520362 | 0,09213914 |
| 800 | 1692 | 285 | 25,096 | 0,16843972 | 0,08805614 |
| 1600 | 14907 | 2219 | 185,14 | 0,14885624 | 0,08343398 |
| 3200 | Oot | 16749 | 1362,5 | #¡VALOR! | 0,08134814 |
| 6400 | Oot | Oot | 10753 | #¡VALOR! | #¡VALOR! |

The division ratio between the program executed in python and the one executed in java has a slightly tendency to 0, we can suppose that the cause of this behavior, even though the program is the same, is the different approaches of the languages python is an interpreted language and java a compiled one and it seems that compiled languages are faster.

Executing the program in java with optimization and without it, reflect a slight improvement when executing with optimization which we can see in the tendency of the division ratio (to 0). This make sense because the optimization of java, improve the performance of the repeated operations and as the size of the problem increases this improvement in the times becomes bigger.