Activity 1. Direct exchange or bubble algorithm

Table 1, Bubble algorithm (WITHOUT OPTIMIZATION)

|  |  |  |  |
| --- | --- | --- | --- |
| N | T ordered | T reverse | T random |
| 10000 | 335 ms | 1776 ms | 1024 ms |
| 2\*10000 | 1278 ms | 6986 ms | 4121 ms |
| 2\*\*2\*10000 | 5129 ms | 28005 ms | 16709 ms |
| 2\*\*3\*10000 | 20841 ms | OoT(80K+) | OoT(66k) |
| 2\*\*4\*10000 | OoT | OoT | OoT |

This algorithm has a best complexity of O(n) therefore the ordered one is the fastest.The worst complexity of it is O(n^2) that is why we can see that the reverse is the slowest and the random is in between the values.

Activity 2. Selection algorithm

Table 2, Selection algorithm (WITHOUT OPTIMIZATION)

|  |  |  |  |
| --- | --- | --- | --- |
| N | T ordered | T reverse | T random |
| 10000 | 315 | 288 | 323 |
| 2\*10000 | 1253 | 1128 | 1275 |
| 2\*\*2\*10000 | 5012 | 4479 | 5101 |
| 2\*\*3\*10000 | 20115 | 18272 | 20272 |
| 2\*\*4\*10000 | OoT(70k+) | OoT(74k) | OoT(79k) |

This algorithm has always the same complexity O(n^2) that is why in the three cases the timings are the same.

Activity 3. Insertion algorithm

Table 3, Insertion algorithm (WITHOU OPTIMIZATION)

|  |  |  |  |
| --- | --- | --- | --- |
| N | T ordered | T reverse | T random |
| 10000 | LoR | 304 | 157 |
| 2\*10000 | LoR | 1173 | 606 |
| 2\*\*2\*10000 | LoR | 4816 | 2433 |
| 2\*\*3\*10000 | LoR | 19402 | 9675 |
| 2\*\*4\*10000 | LoR | OoT(76k) | 38466 |
| 2\*\*5\*10000 | LoR | OoT | OoT |
| 2\*\*6\*10000 | LoR | OoT | OoT |
| 2\*\*7\*10000 | LoR | OoT | OoT |
| 2\*\*8\*10000 | 49 | OoT | OoT |
| 2\*\*9\*10000 | 91 | OoT | OoT |
| 2\*\*10\*10000 | 185 | OoT | OoT |
| 2\*\*11\*10000 | 376 | OoT | OoT |
| 2\*\*12\*10000 | 752 | OoT | OoT |
| 2\*\*13\*10000 | 1502 | OoT | OoT |

Activity 4. Quicksort algorithm

Table 4, Quicksort algorithm (WITHOU OPTIMIZATION)

|  |  |  |  |
| --- | --- | --- | --- |
| N | T ordered | T reverse | T random |
| 250000 | LoR | LoR | LoR |
| 2\*250000 | 63 | 72 | LoR |
| 2\*\*2\*250000 | 123 | 142 | LoR |
| 2\*\*3\*250000 | 255 | 288 | LoR(29) |
| 2\*\*4\*250000 | 532 | 596 | 59 |
| 2\*\*5\*250000 | 1144 | 1265 | 122 |
| 2\*\*6\*250000 | 2283 | 2577 | 261 |

Activity 5. Quicksort + Insertion algorithm

Table 5, Quicksort + Insertion (WITHOUT OPTIMIZATION)

|  |  |
| --- | --- |
| n | T random |
| Quicksort | 2283 |
| Quicksort + Insertion(K=5) |  |
| Quicksort + Insertion(K=10) |  |
| Quicksort + Insertion(K=20) |  |
| Quicksort + Insertion(K=30) |  |
| Quicksort + Insertion(K=50) |  |
| Quicksort + Insertion(K=100) |  |
| Quicksort + Insertion(K=200) |  |
| Quicksort + Insertion(K=500) |  |
| Quicksort + Insertion(K=1000) |  |