Activity 1. Bubble algorithm

|  |  |  |  |
| --- | --- | --- | --- |
| n (10000) | t ordered | t reversed | t random |
| 1 | 319 | 1512 | 1072 |
| 2 | 1271 | 5962 | 4252 |
| 4 | 5064 | 23892 | 17041 |
| 8 | 20172 | 96432 | 67629 |
| 16 | 80878 | OoT | OoT |

t ordered follows a quadratic O(n2) time as each time you double the n, the time increases in 22

t reversed follows also a quadratic O(n2) time

t random follows also a quadratic time

The 3 follows a quadratic time as the bubble algorithm is O(n2) for sorted, unsorted, average,…

Activity 2. Selection algorithm

|  |  |  |  |
| --- | --- | --- | --- |
| n (10000) | t ordered | t reversed | t random |
| 1 | 319 | 283 | 310 |
| 2 | 1249 | 1124 | 1215 |
| 4 | 4995 | 4486 | 4816 |
| 8 | 19915 | 17867 | 19242 |
| 16 | 78723 | 71469 | 77405 |

t ordered follows a quadratic O(n2) time

t reversed follows also a quadratic O(n2) time

t random follows also a quadratic time

The 3 follows a quadratic time as the selection algorithm is O(n2) as the bubble, for sorted, unsorted, average,…

Activity 3. Insertion algorithm

|  |  |  |  |
| --- | --- | --- | --- |
| n (10000) | t ordered | t reversed | t random |
| 1 | LoR | 295 | 153 |
| 2 | LoR | 1161 | 580 |
| 4 | LoR | 4625 | 2341 |
| 8 | LoR | 18550 | 9300 |
| 16 | LoR | 75583 | 37311 |
| 32 | LoR | OoT | OoT |
| 64 | LoR | OoT | OoT |
| 128 | LoR | OoT | OoT |
| 256 | LoR | OoT | OoT |
| 512 | 91 | OoT | OoT |
| 1024 | 182 | OoT | OoT |
| 1024 | 361 | OoT | OoT |
| 2048 | 723 | OoT | OoT |
| 4096 | 1442 | OoT | OoT |
| 8192 | 2872 | OoT | OoT |

t ordered is so fast as this algorithm checks if it is sorted in each iteration so if it is sorted in x iterations it stops

t reversed and t random are as normal with this algorithm O(n2)

Activity 4. Quicksort algorithm

|  |  |  |  |
| --- | --- | --- | --- |
| n (250000) | t ordered | t reversed | t random |
| 1 | LoR | 295 | 153 |
| 2 | LoR | 1161 | 580 |
| 4 | LoR | 4625 | 2341 |
| 8 | LoR | 18550 | 9300 |
| 16 | LoR | 75583 | 37311 |
| 32 | LoR | OoT | OoT |
| 64 | LoR | OoT | OoT |

t ordered is so fast as this algorithm checks if it is sorted in each iteration so if it is sorted in x iterations it stops

t reversed and t random are as normal with this algorithm O(n2)