Name: Nguyễn Hữu Khang

Student id: 2011365

**LAB 6 REPORT**

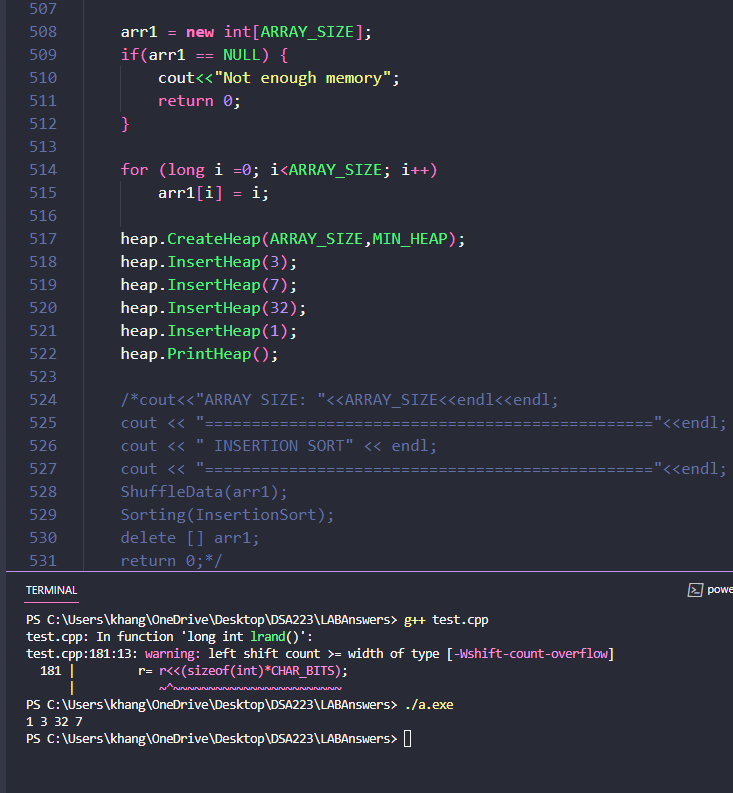
**Complete Heap class and implement min – heap**

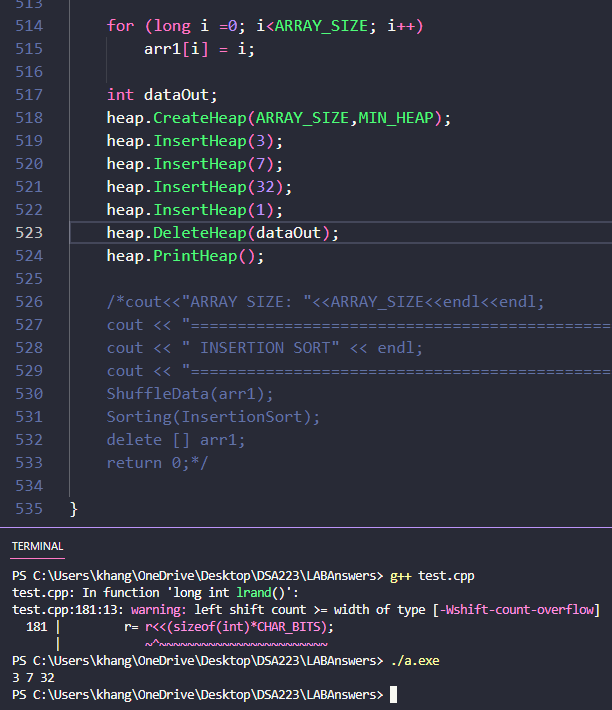
**Code:**

****

****

**Running test result:**

****

****

**MEASURE TIME OF SORTING:**

ARRAY\_SIZE = 50000

**First measurement:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type of Sorting** | **Random array** | **Increasing array** | **Decreasing array** |
| Insertion Sort | 1.422 | 0 | 2.578 |
| Shell Sort | 0 | 0 | 0.016 |
| Selection Sort | 2.25 | 2.281 | 2.172 |
| Heap Sort | 0.015 | 0 | 0.016 |
| Bubble Sort | 6.531 | 0 | 5.578 |
| Quick Sort | 0 | 0 | 0.015 |
| Merge Sort | 0 | 0 | 0.016 |

**Second measurement:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type of Sorting** | **Random array** | **Increasing array** | **Decreasing array** |
| Insertion Sort | 1.36 | 0 | 2.577 |
| Shell Sort | 0.016 | 0 | 0 |
| Selection Sort | 2.25 | 2.234 | 2.157 |
| Heap Sort | 0 | 0 | 0.015 |
| Bubble Sort | 6.656 | 0 | 5.329 |
| Quick Sort | 0 | 0 | 0 |
| Merge Sort | 0 | 0 | 0.016 |

In conclusion, the time of sorting that has been measured is close to the theory. Giving the average amount of numbers, the time of Shell Sort is quite close to Heap Sort, Quick Sort and Merge Sort which have O(log n) complexity.