

WIA1002 Data Structure

Lab 8: Sorting

1. Create a program to sort N random numbers within 0-100 using merge sort in ascending order. However this number is special where the odd number is always smaller than the even number.

Example output:

```
Enter N Special Random Number within 0-100: 10
The Special Number are : 34 32 40 92 95 79 73 65 92 54
After Merge Sort : 65 73 79 95 32 34 40 54 92 92
```

2. Create a program to read the products from a text file (**lab8Q2.txt**). The product consists of PID, SID, price and quantity. Sort the product by price followed by PID in ascending order using quick sort.

Example output:

```
Reading data from product text file
P003 : S020 : 30.45 : 43
P006 : S022 : 10.45 : 22
P001 : S020 : 20.15 : 50
P013 : S018 : 42.5 : 13
P024 : S023 : 20.15 : 230
P004 : S020 : 42.5 : 21
P009 : S012 : 20.15 : 13
After Quick Sort (Price, PID) :
P006 : S022 : 10.45 : 22
P001 : S020 : 20.15 : 50
P009 : S012 : 20.15 : 13
P024 : S023 : 20.15 : 230
P003 : S020 : 30.45 : 43
P004 : S020 : 42.5 : 21
P013 : S018 : 42.5 : 13
```

3. Compare the performance of Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, and Quick Sort in the SortTest class using the 1000 sets of 100000 random numbers (Must use the same random set number for each sorting algorithm). Then, sort the sorting algorithms and identify the sorting algorithm with the fastest time.

Example output:

```

Performance Comparison of Sorting Algorithms
Running on 1000 sets of 100000 random numbers
Quick Sort : 3399
Selection Sort : 3774
Merge Sort : 5147
Insertion Sort : 5279
Bubble Sort : 6436
The fastest sorting algo is Quick Sort

```

4. Heap sort is a comparison based sorting technique based on Binary Heap data structure. It is similar to selection sort where we first **find the maximum element and place the maximum element at the end**. We repeat the same process for remaining element. A heap is an implementation of a priority queue using the BST. The tree must be a complete binary tree. In the array implementation of BST, if the root is **K**, the left child is **2K+1** and the right child is **2K+2**. The parent of K is **(K-1)/2**. Create a program to sort N random characters using Heap Sort in ascending order. The heapSort method can be added in the SortTest class.

Heap Sort Algorithm

Build a max heap. A max-heap has the property that every node stores a value that is greater than or equal to the value of either of its children. The root node contains the maximum value in the structure.

After that, swap this element with the last element.

Reheap the max heap excluding the last element then decrease the length of heap by one.

Example output:

```

Enter N characters : 10
The characters are : O E V X F M R N J E
Heap Sort - Ascending Order
E E F J M N O R V X

```