

Assignment-10 (Sorting)

---

1. Selection sort iteratively selects the smallest element from the unsorted portion of the list and swaps it with the first element of the unsorted part. This process is repeated for the remaining unsorted portion until the entire list is sorted. Write a function ***Select\_sort()*** which recursively sorts the given array in increasing order and prints the resulting array after each iteration.

**Sample Input:**

Enter the number of elements: 6

Enter the elements: 4, 1, 9, 2, 3, 6

**Sample Output:**

Iteration 1: 1, 4, 9, 2, 3, 6

Iteration 2: 1, 2, 9, 4, 3, 6

Iteration 3: 1, 2, 3, 4, 9, 6

Iteration 4: 1, 2, 3, 4, 9, 6

Iteration 5: 1, 2, 3, 4, 6, 9

2. Write a function ***Quick\_Sort()*** which will sort an array of characters in descending order using *Quick* sort algorithm.

**Sample Input:**

Enter the length of array: 5

Enter the characters: C, A, D, B, E

**Sample Output:**

Sorted array is: E, D, C, B, A

3. Given two sorted arrays of integers, your task is to implement ***merge()*** procedure of merge sort algorithm which will merge those two-input array and also prints the number of comparisons required to merge.

**Sample Input:**

Enter the number of elements in the first array: 4

Enter 4 elements to be stored in first array: 2, 5, 8, 10

Enter the number of elements in the second array: 6

Enter 4 elements to be stored in second array: 3, 7, 9, 11, 20, 30

**Sample output:**

Number of comparisons required is: 7

Combined array after merging is: 2, 3, 5, 7, 8, 9, 10, 11, 20, 30

4. Heap Sort is a comparison-based sorting algorithm that works by transforming an arbitrary array into a binary heap. In the context of sorting in ascending order, the array is converted into a max-heap. Write a function ***Heap\_sort()*** which will sort an array into ascending order using heap sort algorithm.

**Sample Input:**

Enter the size of array: 5

Enter the elements: 8 4 2 1 6

**Sample Output:**

Sorted array is: 1 2 4 6 8