**Q1.**

**1] Stable(Merge Sort):-**

#include <stdio.h>

#include <stdlib.h>

#define SIZE 10

void mergelist(int a[], int lb1, int ub1, int lb2, int ub2);

void mergesort(int a[], int low, int high) {

    if (low == high)

        return;

    else {

        int mid = (low + high) / 2;

        mergesort(a, low, mid);     // left subarray

        mergesort(a, mid + 1, high); // right subarray

        mergelist(a, low, mid, mid + 1, high);

    }

}

void mergelist(int a[], int lb1, int ub1, int lb2, int ub2) {

    int i = lb1;

    int j = lb2;

    int k = 0;

    int c[SIZE];

    while (i <= ub1 && j <= ub2) {

        if (a[i] < a[j])

            c[k++] = a[i++];

        else

            c[k++] = a[j++];

    }

    while (j <= ub2) {

        c[k++] = a[j++];

    }

    while (i <= ub1)

        c[k++] = a[i++];

    for (i = lb1, k = 0; i <= ub2; i++, k++)

        a[i] = c[k];

}

int main() {

    int arr[] = {12, 11, 13, 5, 6, 7};

    int arr\_size = sizeof(arr) / sizeof(arr[0]);

    printf("Given array is \n");

    for (int i = 0; i < arr\_size; i++)

        printf("%d ", arr[i]);

    printf("\n");

    mergesort(arr, 0, arr\_size - 1);

    printf("Sorted array is \n");

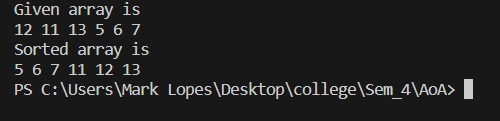
    for (int i = 0; i < arr\_size; i++)

        printf("%d ", arr[i]);

    printf("\n");

    return 0;

}



**2] Fast(Quick Sort):-**

#include <stdio.h>

void swap(int \*a, int \*b) {

    int temp = \*a;

    \*a = \*b;

    \*b = temp;

}

int partition(int arr[], int low, int high) {

    int pivot = arr[high];

    int i = low;

    for (int j = low; j < high; j++) {

        if (arr[j] < pivot) {

            i++;

            swap(&arr[i - 1], &arr[j]);

        }

    }

    swap(&arr[i], &arr[high]);

    return i;

}

void quickSort(int arr[], int low, int high) {

    if (low < high) {

        int pi = partition(arr, low, high);

        quickSort(arr, low, pi - 1);

        quickSort(arr, pi + 1, high);

    }

}

void printArray(int arr[], int size) {

    for (int i = 0; i < size; i++)

        printf("%d ", arr[i]);

    printf("\n");

}

int main() {

    int arr[] = {1,45,34,87,56,23};

    int arr\_size = 6;

    printf("Given array is \n");

    printArray(arr, arr\_size);

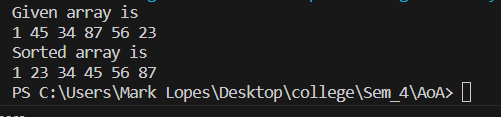
    quickSort(arr, 0, arr\_size - 1);

    printf("Sorted array is \n");

    printArray(arr, arr\_size);

    return 0;

}

****

**Postlab:-**

