**ADA LAB WEEK 5**

**Zayd Ahmed**

**1BM21CS254**

**18/07/23**

**Q1) Sort a given set of N integer elements using the Merge Sort technique. Run the program for different values of N.**

#include <stdio.h>

void merge(int a[], int low, int high, int mid) {

int c[high - low + 1];

int k = 0;

int i = low;

int j = mid + 1;

while (i <= mid && j <= high) {

if (a[i] < a[j]) {

c[k++] = a[i++];

} else {

c[k++] = a[j++];

}

}

while (i <= mid) {

c[k++] = a[i++];

}

while (j <= high) {

c[k++] = a[j++];

}

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

a[t] = c[t - low];

}

}

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

if (low < high) {

int mid = (low + high) / 2;

mergesort(a, low, mid);

mergesort(a, mid + 1, high);

merge(a, low, high, mid);

}

}

int main() {

int n;

printf("Enter the size of the array: ");

scanf("%d", &n);

int a[n];

printf("Enter the elements of the list:\n");

for (int i = 0; i < n; i++) {

scanf("%d", &a[i]);

}

mergesort(a, 0, n - 1);

printf("Sorted array: ");

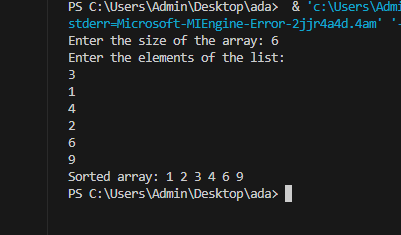
for (int i = 0; i < n; i++) {

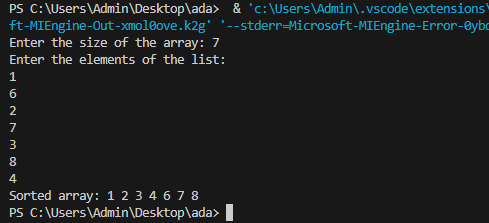
printf("%d ", a[i]);

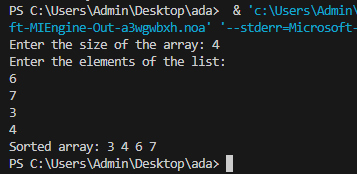
}

return 0; }

**Output:**

****

****

****

**Q2) Sort a given set of N integer elements using the Quick Sort technique.**

# include <stdio.h>

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

int mid;

if(low<high){

mid=partition(a,low,high);

QuickSort(a,low,mid);

QuickSort(a,mid+1,high);

}

}

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

int i=low+1;

int j=high;

int temp;

int pivot=a[low];

while(i<=j){

while(a[i]<pivot){

i++;

}

while(a[j]>pivot){

j--;

}

if(i<j){

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

}

temp=a[low];

a[low]=a[j];

a[j]=temp;

return j;

}

int main(){

int n;

printf("Enter the size of the array: ");

scanf("%d", &n);

int a[n];

printf("Enter the elements of the list:\n");

for (int i = 0; i < n; i++) {

scanf("%d", &a[i]);

}

QuickSort(a, 0, n-1);

printf("Sorted array: ");

for (int i = 0; i < n; i++) {

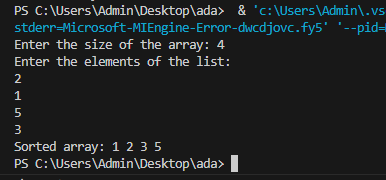
printf("%d ", a[i]);

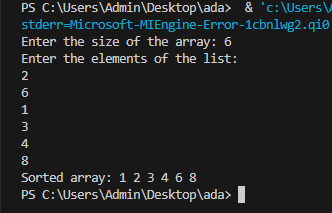
}

return 0;

}

**Output:**

****

****