Assignment: 10

Name: Ujjwal Kumar Jha

Registration No. 20194196

Group: C_2

Q1. WAP to implement the following:

a) Bubble sort.

b) Selection sort.

c) Insertion sort.

d) Merge sort.

e) Quick sort.

f) Heap sort.

```
#include <stdio.h>
#include <stdlib.h>
void insertion(int a[], int n){
    int i,j,x;

    for(i=1;i<n;i++){
        j=i-1;
        x=a[i];
        while(j>-1&&a[j]>x){
            a[j+1]=a[j];j--;
        }
  a[j+1]=x;
        }

    }

void bubbleSort(int a[], int n){
int i,j,x;
for(i=0;i<n-1;i++){
```

```
  int flag=0;
  for(j=0;j<n-1-i;j++){
      if(a[j]>a[j+1]){
          x=a[j];
          a[j]=a[j+1];
          a[j+1]=x;
       flag=1;
       }
       }
      if(flag==0){

          break;
      }
  }



}

void selectionSort(int a[],int n){
    int i,j,k,x;
    for(i=0;i<n-1;i++){
        for(j=k=i;j<n;j++){
            if(a[j]<a[k])
            k=j;
        }
        x=a[i];
        a[i]=a[k];a[k]=x;
    }

}
void swap(int *a, int *b){
    int x;
    x=*a;
    *a=*b;
    *b=x;

}

int partition(int a[],int l, int h){
    int pivot,i,j;
    pivot=a[l];
    i=l;j=h;
    do{
        do
```

```
        {
            i++;
        } while (a[i]<=pivot);
        do
        {
            j--;
        } while (a[j]>pivot);
        if (i<j)
        {
            swap(&a[i],&a[j]);
        }


    }while(i<j);
    swap(&a[l],&a[j]);
    return j;
}
void quickSort(int a[],int l,int h){
    int i,n,j;
    if(l<h){
        j=partition(a,l,h);
        quickSort(a,l,j);
        quickSort(a,j+1,h);
    }

}
void Merge(int a[], int l,int mid, int h){
    int i,j,k;
    i=l;
    j=mid+1;
    k=l;
    int b[100];
    while(i<=mid&&j<=h){
        if(a[i]<a[j])
        b[k++]=a[i++];
        else
        {
            b[k++]=a[j++];
        }}

        for(;i<=mid;i++)
        b[k++]=a[i];
        for(;j<=h;j++)
        b[k++]=a[j];
```

```
        for(i=l;i<=h;i++)
        a[i]=b[i];

}

void ImergSort(int a[], int n){
    int p,i,l,h,mid;
    for(p=2;p<=n;p=p*2){
        for(i=0;i+p-1<=n;i=i+p){
            l=i;
            h=i+p-1;
            mid=(l+h)/2;
            Merge(a,l,mid,h);
        }
    }
    if(p/2<n)
    Merge(a,0,p/2-1,n);
}


void heapify(int arr[], int n, int i) {

   int largest = i;
   int left = 2 * i + 1;
   int right = 2 * i + 2;

   if (left < n && arr[left] > arr[largest])
     largest = left;

   if (right < n && arr[right] > arr[largest])
     largest = right;


   if (largest != i) {
     swap(&arr[i], &arr[largest]);
     heapify(arr, n, largest);
   }
 }


 void heapSort(int arr[], int n) {

   for (int i = n / 2 - 1; i >= 0; i--)
     heapify(arr, n, i);
```

```
    for (int i = n - 1; i >= 0; i--) {
      swap(&arr[0], &arr[i]);


      heapify(arr, i, 0);
    }
  }
  void printarray(int a[],int n){
      printf("\nPlease Find the sorted elements:\n");
      for(int i=0;i<n;i++)
      printf("%d\t",a[i]);
  }
void main(){
        int n,i,x;
        int a[100];

        do{

        printf("\n\n**********MAIN MENU********\n1.Bubble sort\n2.Selection Sort\
n3.Insertion sort\n4.Merge\n5.qick\n6.Heap\n7.Exit\nPlease select from menu above
:\n");

        scanf("%d",&x);


        switch (x){
case 3:
printf("Please Enter the size of array:\t");
        scanf("%d",&n);

        printf("Please Enter the elements of un sorted array:\t");
        for(i=0;i<n;i++)
        scanf("%d",&a[i]);
         insertion(a,n); printarray(a,n);break;
         case 1:
         printf("Please Enter the size of array:\t");
        scanf("%d",&n);

        printf("Please Enter the elements of un sorted array:\t");
        for(i=0;i<n;i++)
        scanf("%d",&a[i]);
        bubbleSort(a,n);printarray(a,n);break;
        case 2:
        printf("Please Enter the size of array:\t");
        scanf("%d",&n);
```

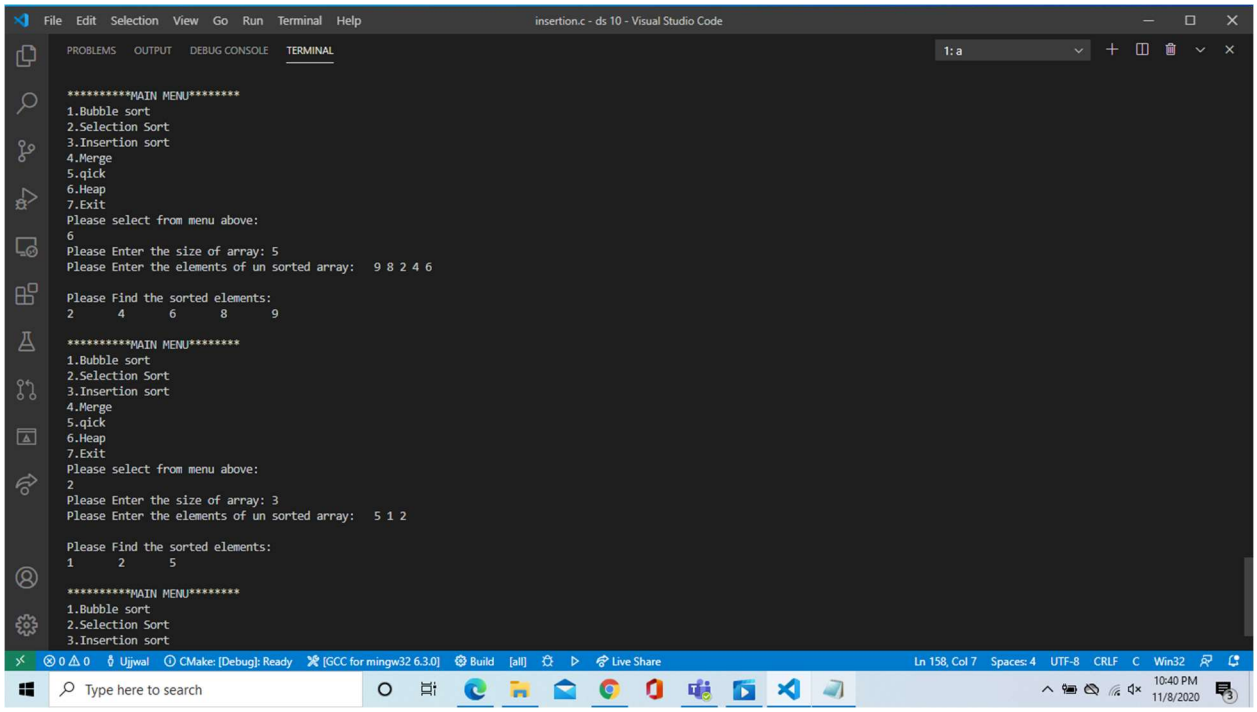
```
     printf("Please Enter the elements of un sorted array:\t");
     for(i=0;i<n;i++)
     scanf("%d",&a[i]);
     selectionSort(a,n);printarray(a,n);break;
     case 5:
     printf("Please Enter the size of array:\t");
     scanf("%d",&n);

     printf("Please Enter the elements of un sorted array:\t");
     for(i=0;i<n;i++)
     scanf("%d",&a[i]);
     quickSort(a,0,n);printarray(a,n);break;
     case 4:
     printf("Please Enter the size of array:\t");
     scanf("%d",&n);

     printf("Please Enter the elements of un sorted array:\t");
     for(i=0;i<n;i++)
     scanf("%d",&a[i]);
    ImergSort(a,n);printarray(a,n);break;
    case 6:
    printf("Please Enter the size of array:\t");
     scanf("%d",&n);

     printf("Please Enter the elements of un sorted array:\t");
     for(i=0;i<n;i++)
     scanf("%d",&a[i]);
  heapSort(a,n);printarray(a,n);break;
  case 7:
  exit(0);
  default:
  printf("\nInvalid option provided:");
    }}while(x!=7);

}
```


File Edit Selection View Go Run Terminal Help
insertion.c - ds 10 - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: a
*********MAIN MENU********
1.Bubble sort
2.Selection Sort
3.Insertion sort
4.Merge
5.qick
6.Heap
7.Exit
Please select from menu above:
6
Please Enter the size of array: 5
Please Enter the elements of un sorted array: 9 8 2 4 6
Please Find the sorted elements:
2 4 6 8 9
*********MAIN MENU********
1.Bubble sort
2.Selection Sort
3.Insertion sort
4.Merge
5.qick
6.Heap
7.Exit
Please select from menu above:
2
Please Enter the size of array: 3
Please Enter the elements of un sorted array: 5 1 2
Please Find the sorted elements:
1 2 5
*********MAIN MENU********
1.Bubble sort
2.Selection Sort
3.Insertion sort
Ujjwal
CMake: [Debug]: Ready
[GCC for mingw32 6.3.0]
Build
[all]
Live Share
Ln 158, Col 7
Spaces: 4
UTF-8
CRLF
C
Win32
Type here to search
10:40 PM
11/8/2020