

**DATA STRUCTURES LAB**

**WEEK-7**

**NAME : ABHISHEK KUMAR JHA**

**ROLL NO : 19R21A05C2**

**DATE : 6/1/2021**

**PROBLEM STATEMENT:**

Write a C program to implement Quick sort, Merge sort and Heap Sort

**CODE:**

**QUICK SORT:**

#include<stdio.h>

void qsort(int a[10],int first,int last)

{

int i,j,t,pivot,n;

if(first<last)

{

i=first;

j=last;

pivot=first;

while(i<j)

{

while(a[i]<=a[pivot]&& i<last)

i++;

while(a[j]>a[pivot])

j--;

if(i<j)

{

t=a[i];

a[i]=a[j];

a[j]=t;

}

}

t=a[pivot];

a[pivot]=j;

a[j]=t;

qsort(a,first,j-1);

qsort(a,j+1,last);

}

}

int main()

{

int i,n,a[50];

printf("Enter no of elements\n");

scanf("%d",&n);

printf("Enter elements\n");

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

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

printf("Elements before sorting\n");

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

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

qsort(a,0,n-1);

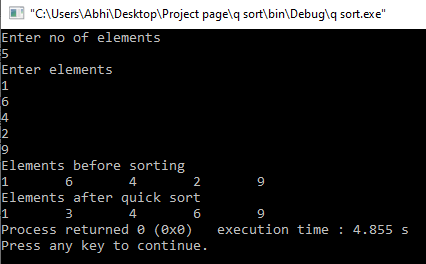
printf("\nElements after quick sort\n");

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

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

}

**Output:**



**CODE:**

**MERGE SORT:**

#include <stdio.h>

void mergesort(int a[100],int low,int high)

{

int mid;

if(low<high)

{

mid=(low+high)/2;

mergesort(a,low,mid);

mergesort(a,mid+1,high);

merge(a,low,high,mid);

}

}

void merge(int a[100],int low,int high,int mid)

{

int i,j,k,c[50];

i=low;j=mid+1;k=low;

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

{

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

{

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

}

else

{

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

}

}

while(i<=mid)

{

c[k]=a[i];

i++;

k++;

}

while(j<=high)

{

c[k]=a[j];

j++;

k++;

}

for(i=0;i<k;i++)

a[i]=c[i];

}

int main()

{

int i,n,a[50];

printf("Enter no of elements\n");

scanf("%d",&n);

printf("Enter elements\n");

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

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

printf("Elements before sorting\n");

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

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

mergesort(a,0,n-1);

printf("\nElements after merge sort\n");

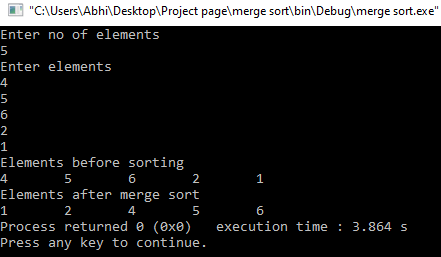
for(i=0;i<n;i++)

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

return 0;

}

**Output:**



**CODE:**

**HEAP SORT:**

#include<stdio.h>

void adjust(int i,int n,int a[])

{

int j,item;

j=2\*i;

item=a[i];

while(j<=n)

{

if((j<n)&&(a[j]<a[j+1]))

j++;

if(item>=a[j])

break;

else

{

a[j/2]=a[j];

j=2\*j;

}

}

a[j/2]=item;

}

void heapify(int n,int a[])

{

int i;

for(i=n/2;i>0;i--)

adjust(i,n,a);

}

void heapsort(int n,int a[])

{

int temp,i;

heapify(n,a);

for(i=n;i>0;i--)

{

temp=a[i];

a[i]=a[1];

a[1]=temp;

adjust(1,i-1,a);

}

}

int main()

{

int i,n,a[50];

printf("Enter no of elements\n");

scanf("%d",&n);

printf("Enter elements\n");

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

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

printf("Elements before sorting\n");

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

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

heapsort(n,a);

printf("\nElements after heap sort\n");

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

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

}

**Output:**

