**Practical-5**

***AIM:* Implementation of max-heap sort algorithm**

**Input:**

#include<stdio.h>

#include<stdlib.h>

#include<conio.h>

#define MAX 10

void main()

{

int i,n;

int arr[MAX];

void makeheap(int arr[MAX],int n);

void heapsort(int arr[MAX],int n);

void display(int arr[MAX],int n);

clrscr();

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

arr[i]=0;

printf("\n How many elements you want to insert?");

scanf("%d",&n);

printf("\n Enter the elements");

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

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

printf("\n The Elements are ...");

display(arr,n);

makeheap(arr,n);

printf("\n Heapified");

display(arr,n);

heapsort(arr,n);

printf("\nElements sorted by Heap sort... ");

display(arr,n);

getch();

}

void makeheap(int arr[MAX],int n)

{

int i,val,j,father;

for(i=1;i<n;i++)

{

val=arr[i];

j=i;

father=(j-1)/2;//finding the parent of node j

while(j>0&&arr[father]<val)//creating a MAX heap

{

arr[j]=arr[father];//preserving parent dominance

j=father;

father=(j-1)/2;

}

arr[j]=val;

}

}

void heapsort(int arr[MAX],int n)

{

int i,k,temp,j;

for(i=n-1;i>0;i--)

{

temp=arr[i];

arr[i]=arr[0];

k=0;

if(i==1)

j=-1;

else

j=1;

if(i>2&&arr[2]>arr[1])

j=2;

while(j>=0&& temp <arr[j])

{

arr[k]=arr[j];

k=j;

j=2\*k+1;

if(j+1<=i-1&&arr[j]<arr[j+1])

j++;

if(j>i-1)

j=-1;

}

arr[k]=temp;

}

}

void display(int arr[MAX],int n)

{

int i;

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

printf("\n %d",arr[i]);

}