**Assignment-5**

**Problem Statement:-**    
Develop functions to insert and delete into/from a max heap.  
  
**Course Objective:-** To know the basics of computational complexity analysis and various algorithm design strategies.  
  
**Course Outcomes:-** Students will be able to Build the various algorithmic design paradigms.  
  
**Program:-**

#include<stdio.h>

#define MAX 100

int h[MAX];

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

int c=\*a;

\*a=\*b;

\*b=c;

}

void heap\_up(int h[],int i){

int parent=(i-1)/2;

if(h[i]>h[parent] && i>0){

swap(&h[i],&h[parent]);

heap\_up(h,parent);

}

}

void display(int h[],int size){

printf("\nThe Elements of MAX Heap are:-\n");

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

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

}

printf("\n");

}

void create(int h[],int size){

int n;

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

printf("Enter %d Element: ",i+1);

scanf("%d",&n);

h[i]=n;

heap\_up(h,i);

}

}

void heap\_down(int h[],int i,int last){

int l,r;

l=2\*i+1;

r=2\*i+2;

int largest=i;

if(l<=last && h[i]<h[l]){

largest=l;

}

if(r<=last && h[r]>h[l]){

largest=r;

}

if(largest!=i){

swap(&h[i],&h[largest]);

heap\_down(h,largest,last);

}

}

int delete(int h[],int size){

swap(&h[0],&h[size-1]);

size--;

heap\_down(h,0,size-1);

return size;

}

int main(){

int size,c;

do{

printf("--------------------------\nEnter Your Choice \n1.Create/Insert\n2.Delete\n3.Display\n4.Exit\n--------------------------\n");

scanf("%d",&c);

switch(c){

case 1:

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

scanf("%d",&size);

create(h,size);

break;

case 2:

size=delete(h,size);

break;

case 3:

display(h,size);

break;

}

}while(c!=4);

return 0;

}

**Output:-**



