Name: Ankush jha

Roll no: BT22CSH023

Code:

*#include*<stdio.h>

*#include*<stdlib.h>

voidswap(int\**a*,int\**b*){

    inttemp=\**a*;

    \**a*=\**b*;

    \**b*=temp;

}

voidmaxHeapify(int*arr*[],int*n*,int*i*){

    intlargest=*i*;

    intleft=2\**i*+1;

    intright=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]);

        maxHeapify(*arr*,*n*,largest);

    }

}

voidbuildHeap(int*arr*[],int*n*){

    inti;

*for*(i=(*n*/2)-1;i>=0;i--){

        maxHeapify(*arr*,*n*,i);

    }

}

intdeleteMax(int*arr*[],int\**n*){

*if*(\**n*<=0){

        printf("Heap is empty\n");

*return*-1;

    }

    intmaxElement=*arr*[0];

*arr*[0]=*arr*[\**n*-1];

    (\**n*)--;

*// Modify maxHeapify to maintain the max heap property*

    inti=0;

*while*(1){

        intlargest=i;

        intleft=2\*i+1;

        intright=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]);

            i=largest;

        }*else*{

*break*;

        }

    }

*return*maxElement;

}

voidheapSort(int*arr*[],int*n*){

    buildHeap(*arr*,*n*);

    inti;

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

        swap(&*arr*[0],&*arr*[i]);

        maxHeapify(*arr*,i,0);

    }

}

voidprintArray(int*arr*[],int*n*){

    inti;

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

        printf("%d ",*arr*[i]);

    }

    printf("\n");

}

intmain(){

    intarr[]={1,5,6,8,9,7,3};

    intn=sizeof(arr)/sizeof(arr[0]);

    printf("Original array: ");

    printArray(arr,n);

    buildHeap(arr,n);

    printf("Max Heap: ");

    printArray(arr,n);

    intmax=deleteMax(arr,&n);

    printf("Deleted Max Element: %d\n",max);

    printf("Max Heap after deletion: ");

    printArray(arr,n);

    heapSort(arr,n);

    printf("Sorted array using Heap Sort: ");

    printArray(arr,n);

*return*0;

}

Output:

