**DAY 13 – HEAP SORT**

16.Write a C program to implement heap sort

**PROGRAM**

#include <stdio.h>

#define SIZE 20

int heap[SIZE], finalheap[SIZE], n, item;

void insert(int *i*)

{

    int val = heap[*i*];

    while(*i* > 1 && heap[*i* / 2] < val)

    {

    heap[*i*] = heap[*i* / 2];

*i* /= 2;

    }

    heap[*i*] = val;

}

void delete()

{

    int last, ptr = 1, left = 2, right = 3, temp;

    item = heap[1];

    last = heap[n];

    n = n - 1;

    heap[ptr] = last;

    while(left <= n)

    {

        if(heap[ptr] >= heap[left] && heap[ptr] >= heap[right])

        return;

        if(heap[right] <= heap[left])

        {

            temp = heap[ptr];

            heap[ptr] = heap[left];

            heap[left] = temp;

            ptr = left;

        }

        else

        {

            temp = heap[ptr];

            heap[ptr] = heap[right];

            heap[right] = temp;

            ptr = right;

        }

        left = 2 \* ptr;

        right = left + 1;

    }

}

void main()

{

    int i, temp, k;

    printf("Enter the no. of elements: ");

    scanf("%d", &n);

    printf("Enter the elements: ");

    for(i = 1; i <= n; i++)

    {

        scanf("%d", &heap[i]);

        insert(i);

    }

    printf("Heap array: ");

    for(i = 1; i <= n; i++)

    printf("%d ", heap[i]);

    temp = n;

    k = n;

    while(n >= 1)

    {

        delete();

        finalheap[k--] = item;

    }

    n = temp;

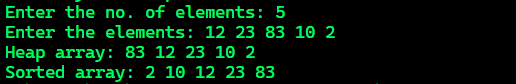
    printf("\nSorted array: ");

    for(i = 1; i <= n; i++)

        printf("%d ", finalheap[i]);

}

**OUTPUT**

****