**TITLE 17**

Write a C program to sort an array in descending order using Heapsort

**OBJECTIVE:**

By the end of this activity we will be able to Implement Heapsort in C

**PROBLEM STATEMENT:**

In this problem we aim to create an array using the numbers given as user input and then using Heapsort we arrange the elements in descending order. It requires input from the user:

Enter the numbers:

Once input data is collected and stored, the elements are stored in an array and are sorted.

**ALGORITHM:**

START

Define variables: n,i,last,temp

INPUT: Read input from user

COMPUTATION: An array is formed from the elements entered by the user

DISPLAY: Sorted array of elements in descending order

STOP

**PROGRAM:**

#include<stdio.h>

void create(int []);

void down\_adjust(int [],int);

void main()

{

int heap[100],n,i,last,temp;

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

scanf("%d",&n);

printf("Enter elements:\n");

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

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

//create a heap

heap[0]=n;

create(heap);

//sorting

while(heap[0] > 1)

{

last=heap[0];

temp=heap[1];

heap[1]=heap[last];

heap[last]=temp;

heap[0]--;

down\_adjust(heap,1);

}

//print sorted data

printf("Array after sorting:\n");

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

printf("%d ",heap[i]);

}

void create(int heap[])

{

int i,n;

n=heap[0]; //no. of elements

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

down\_adjust(heap,i);

}

void down\_adjust(int heap[],int i)

{

int j,temp,n,flag=1;

n=heap[0];

while(2\*i<=n && flag==1)

{

j=2\*i; //j points to left child

if(j+1<=n && heap[j+1] > heap[j])

j=j+1;

if(heap[i] > heap[j])

flag=0;

else

{

temp=heap[i];

heap[i]=heap[j];

heap[j]=temp;

i=j;

}

}

}

**CONCLUSION:**

The simulation of the above C program helped me to understand how we can use Heapsort to arrange elements in ascending order.

**OUTPUT:**

Enter no. of elements:

6

Enter the elements:

10

43

62

78

1

12

Array after sorting:

1

10

12

43

62

78