6.Problem Statement: Heap Sort

**Problem Analysis:**

Heap sort is a comparison based sorting technique based on Binary Heap data structure. It is similar to selection sort where we first find the maximum element and place the maximum element at the end. We repeat the same process for remaining element.

Heap sort algorithm is divided into two basic parts:

*1.* Creating a Heap of the unsorted list/array.

*2.* Then a sorted array is created by repeatedly removing the largest/smallest element from the heap, and inserting it into the array. The heap is reconstructed after each removal.

Initially on receiving an unsorted list, the first step in heap sort is to create a Heap data structure (Max-Heap or Min-Heap). Once heap is built, the first element of the Heap is either largest or smallest (depending upon Max-Heap or Min-Heap), so we put the first element of the heap in our array. Then we again make heap using the remaining elements, to again pick the first element of the heap and put it into the array. We keep on doing the same repeatedly until we have the complete sorted list in our array.

**Algorithm:**

MaxHeapify(A, i)

l = left(i)

r = right(i)

if l <= heap-size[A] and A[l] > A[i]

then largest = l

else largest = i

if r <= heap-size[A] and A[r] > A[largest]

then largest = r

if largest != i

then swap A[i] with A[largest]

MaxHeapify(A, largest)

end func

BuildMaxHeap(A)

for i = |length[A]/2| downto 1

do MaxHeapify(A, i, n)

end func

HeapSort(A)

BuildMaxHeap(A)

for i = length[A] downto 2

do swap A[1] with A[i]

heap-size[A] = heap-size[A] – 1

MaxHeapify(A, 1)

end func

**Source Code:**

#include<bits/stdc++.h>

using namespace std;

void heapify(int a[],int n)

{

int k,i,j,item;

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

{

item = a[k];

i = k;

j = (i-1)/2;

while((i>0)&&(item>a[j]))

{

a[i] = a[j];

i = j;

j = (i-1)/2;

}

a[i] = item;

}

}

void adjust(int a[],int n)

{

int i,j,item;

j = 0;

item = a[j];

i = 2\*j+1;

while(i<=n-1)

{

if(i+1 <= n-1)

if(a[i] <a[i+1])

i++;

if(item<a[i])

{

a[j] = a[i];

j = i;

i = 2\*j+1;

}

else

break;

}

a[j] = item;

}

void heapsort(int a[],int n)

{

int i,t;

heapify(a,n);

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

{

t = a[0];

a[0] = a[i];

a[i] = t;

adjust(a,i);

}

}

int main()

{

int n,i,a[50];

printf("Enter the limit:\n");

scanf("%d",&n);

printf("Enter the elements:\n");

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

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

heapsort(a,n);

printf("The Sorted Elements Are:\n");

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

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

printf("\n");

}

**Sample Input:**

Enter the limit:

7

Enter the elements:

43 65 67 32 89 54 21

**Sample Output:**

The sorted elements are:

21 32 43 54 65 67 89