**PROGRAM 10**

**Sort a given set of N integer elements using Heap Sort technique and compute its time taken**

**CODE:**

#include <stdio.h>

#include <time.h>

void main()

{

int heap[20], no, i, j, c, root, temp;

clock\_t start, end;

double time;

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

scanf("%d", &no);

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

for(i=0;i<no;i++)

{

heap[i] = rand()%100;

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

}

start = clock();

for (i = 1; i < no; i++)

{

c = i;

do

{

root = (c - 1) / 2;

if (heap[root] < heap[c]) /\* to create MAX heap array \*/

{

temp = heap[root];

heap[root] = heap[c];

heap[c] = temp;

}

c = root;

} while (c != 0);

}

end = clock();

printf("\n\nHeap array : ");

for (i = 0; i < no; i++)

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

for (j = no - 1; j >= 0; j--)

{

temp = heap[0];

heap[0] = heap[j]; /\* swap max element with rightmost leaf element \*/

heap[j] = temp;

root = 0;

do

{

c = 2 \* root + 1; /\* left node of root element \*/

if ((heap[c] < heap[c + 1]) && c < j-1)

c++;

if (heap[root]<heap[c] && c<j) /\* again rearrange to max heap array \*/

{

temp = heap[root];

heap[root] = heap[c];

heap[c] = temp;

}

root = c;

} while (c < j);

}

printf("\n\n The sorted array is : ");

for (i = 0; i < no; i++)

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

time= ((double) (end - start)) / CLOCKS\_PER\_SEC;

printf("\nThe time required to compute the heap sort =%f",time);

getch();

}

**Output:**

