19.HEAP SORT:-

Code:-

```
#include<stdio.h>
#include <stdlib.h>
#define MAX 25
void random_shuffle(int arr[])
{
    int i, j, temp;
    srand(time(NULL));
    for (i = MAX - 1; i > 0; i--) {
        j = rand()%(i + 1);
        temp = arr[i];
        arr[i] = arr[j];
        arr[j] = temp;
    }
}
void  max_heapify(int a[], int i, int heapsize)
{
    int tmp, largest;
    int l = (2 * i) + 1;
    int r = (2 * i) + 2;
    if ((l <= heapsize) && (a[l] > a[i]))
        largest = l;
    else
        largest = i;
    if ((r <= heapsize) && (a[r] > a[largest]))
        largest = r ;
    if (largest != i)
    {
        tmp = a[i];
```

```
        a[i] = a[largest];
        a[largest] = tmp;
        max_heapify(a, largest, heapsize);
    }
}
void  build_max_heap(int a[], int heapsize)
{
    int i;
    for (i = heapsize/2; i >= 0; i--)
    {
        max_heapify(a, i, heapsize);
    }
}
void heap_sort(int a[], int heapsize)
{
    int i, tmp;
    build_max_heap(a, heapsize);
    for (i = heapsize; i > 0; i--)
    {
        tmp = a[i];
        a[i] = a[0];
        a[0] = tmp;
        heapsize--;
        max_heapify(a, 0, heapsize);
    }
}
int main()
{
    int i, r, heapsize;
    int a[MAX];
```

```
    for (i = 0; i < MAX; i++)
        a[i] = i;
    heapsize = MAX - 1;
    random_shuffle(a);
    printf("\n");
    heap_sort(a, heapsize);
    for (i = 0; i < MAX; i++)
        printf("%d ", a[i]);
    return 0;
}
```

OUTPUT:-

```
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
--------------------------------
Process exited after 0.02709 seconds with return value 0
Press any key to continue . . .
```