```c
//Implement HeapSort
#include <stdio.h>

void swap(int &a, int &b)
{
    int t = a;
    a = b;
    b = t;
}

void max_heapify(int A[], int index, int heap_size)
{
    int left = 2*index+1, right = 2*index+2, largest=index;
    if (left <= heap_size && A[left] > A[index])
    {
        largest = left;
    }
    if (right <= heap_size && A[right] > A[index])
    {
        largest = right;
    }
    if(left <= heap_size && right <= heap_size)
    {
        if(A[left] > A[right] && A[left] > A[index])
            largest = left;
        else if(A[right] > A[left] && A[right] > A[index])
            largest = right;
    }
    if (largest != index)
    {
        swap(A[index], A[largest]);
        max_heapify(A, largest, heap_size);
    }
}

void build_maxheap(int A[], int heap_size)
{
    for(int i=heap_size/2 +1 ; i>=0 ; i--)
        max_heapify(A,i,heap_size);
}

void heapsort(int A[], int heap_size)
{
    build_maxheap(A, heap_size);
    for(int i=heap_size ; i>=1 ; i--)
    {
        swap(A[0], A[i]);
        heap_size--;
        max_heapify(A, 0, heap_size);
    }
}

int main()
{
    int a[10] = {4,1,3,2,16,9,10,14,8,7};
    printf("Original array\n");
    for(int i=0 ; i<10 ; i++)
    {
        printf("%6d", a[i]);
    }
    heapsort(a, 9);
    printf("\n");
    printf("Sorted Array\n");
    for(int i=0 ; i<10 ; i++)
    {
        printf("%6d", a[i]);
    }
    printf("\n");
}
```

```
/*
OUTPUT
Original array
     4     1     3     2    16     9    10    14     8     7
Sorted Array
     1     2     3     4     7     8     9    10    14    16
*/
```