```cpp
#include<iostream>
#include<cmath>
using namespace std;

void swap(int arr[], int n, int i)
{
    int temp = arr[n];
    arr[n] = arr[i];
    arr[i] = temp;
}

void heapify(int arr[], int n, int i)
{
    int l_c = 2 * i + 1;
    int r_c = 2 * i + 2;
    int large = i;

    if(l_c < n && arr[l_c] > arr[large])
    {
        large = l_c;
    }
    if(r_c < n && arr[r_c] > arr[large])
    {
        large = r_c;
    }

    if(large != i)
    {
        swap(arr, i, large);
        heapify(arr, n, large);
    }
}

void build_heap(int arr[], int n)
{
    for(int i = (n - 1) / 2; i >= 0; i--)
    {
        heapify(arr, n, i);
    }
}

int height(int arr[], int n)
{
    int i = 0;
    int height = 0;

    while(i < n)
    {
        height++;
        i = 2 * i + 1;
    }
    return height;
}

void heapTree(int arr[], int n)
{
    int h = height(arr, n);
    int blocks = pow(2, h);
    int temp_blocks = blocks;
    int index = 0;
    int k = 3;

    for(int i = 0; i < h; i++)
    {
        for(int j = 0; j < blocks; j++)
        {
            if(j == temp_blocks / 2 && index < n)
            {
                cout << arr[index];
                index++;
            }
            else
            {
                if(j == (temp_blocks * k) / 2 && index < n)
                {
                    cout << arr[index];
                    index++;
                    k += 2;
                }
                else
                {
                    cout << " ";
                }
            }
        }
        cout << "\n";
        temp_blocks = temp_blocks / 2;
        k = 3;
    }
}

int main()
```

```
{
    int n;
    cout << "Enter the length:\n";
    cin >> n;

    int arr[n];
    cout << "Enter elements: \n";
    for(int i = 0; i < n; i++)
    {
        cin >> arr[i];
    }

    build_heap(arr, n);
    heapTree(arr, n);

/*
    int h = height(arr, n);
    int blocks = pow(2, h);
    int temp_blocks = blocks;
    int index = 0;
    int printed_index;
    int k = 3;

    for(int i = 0; i < h; i++)
    {
        for(int j = 0; j < blocks; j++)
        {
            if(j == temp_blocks / 2 && index < n)
            {
                cout << arr[index];
                index++;
                printed_index = j;
            }
            else
            {
                if(j == (temp_blocks * k) / 2 && index < n)
                {
                    cout << arr[index];
                    index++;
                    k += 2;
                }
                else
                {
                    cout << " ";
                }
            }
        }
        cout << "\n";
        printed_index = 0;
        temp_blocks = temp_blocks / 2;
        k = 3;
    }*/

    return 0;
}
```