Aim: write a program in c to perform heap sort

Algorithm:

1. Build a max heap from the input data.
2. At the root of the heap, exchange the first element with the last element.
3. Reduce the size of the heap by 1 and heapify the root element.
4. Repeat steps 2 and 3 until all elements are sorted.

Source Code:

#include <stdio.h>

#include <stdlib.h>

*void* heapify(*int* arr[], *int* n, *int* i)

{

*int* largest = i;

*int* l = 2\*i + 1;

*int* r = 2\*i + 2;

    if (l < n && arr[l] > arr[largest])

        largest = l;

    if (r < n && arr[r] > arr[largest])

        largest = r;

    if (largest != i)

    {

*int* temp = arr[i];

        arr[i] = arr[largest];

        arr[largest] = temp;

        heapify(arr, n, largest);

    }

}

*void* heapSort(*int* arr[], *int* n)

{

    for (*int* i = n / 2 - 1; i >= 0; i--)

        heapify(arr, n, i);

    for (*int* i=n-1; i>=0; i--)

    {

*int* temp = arr[0];

        arr[0] = arr[i];

        arr[i] = temp;

        heapify(arr, i, 0);

    }

}

*void* printArray(*int* arr[], *int* n)

{

    for (*int* i=0; i < n; ++i)

        printf("%d ", arr[i]);

    printf("\n");

}

*int* main()

{

*int* arr[] = {12, 11, 13, 5, 6, 7};

*int* n = sizeof(arr)/sizeof(arr[0]);

    heapSort(arr, n);

    printf("Sorted array is \n");

    printArray(arr, n);

}

Output:

Sorted array is

5 6 7 11 12 13