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

#include <stdio.h>

#include <stdlib.h>

#include <time.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);

}

}

int main() {

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

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

clock\_t start = clock();

heapSort(arr, n);

clock\_t end = clock();

printf("Sorted array: ");

for (int i = 0; i < n; i++)

printf("%d ", arr[i]);

printf("\n");

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

printf("Time taken: %f seconds\n", time\_taken);

return 0;

}

Output:

A screen shot of a black background

AI-generated content may be incorrect.