HEAP SORT

#include<stdio.h>

void heapify(int\*,int, int);

void heapsort(int\*, int);

void print\_array(int\*, int);

int main()

{

int arr[] = { 10, 30, 5, 63, 22, 12, 56, 33 };

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

printf("\nArray before sorting:\n");

print\_array(arr, n);

heapsort(arr, n);

printf("\n\nArray after sorting:\n");

print\_array(arr, n);

return 0;

}

void heapsort(int\* arr, int n)

{

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

{

heapify(arr, n, i);

}

// sort the max heap

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

{

int temp = arr[i];

arr[i] = arr[0];

arr[0] = temp;

heapify(arr, i, 0);

}

}

void heapify(int\* arr, int n, int i)

{

int largest = i;

int left = 2 \* i + 1;

int right = 2 \* i + 2;

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

{

largest = left;

}

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

{

largest = right;

}

if (largest != i)

{

int temp = arr[i];

arr[i] = arr[largest];

arr[largest] = temp;

heapify(arr, n, largest);

}

}

void print\_array(int\* arr, int n)

{

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

{

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

}

}

OUTPUT:

