#include <iostream>

#include <sys/time.h>

#include <sys/resource.h>

void siftDown(int a[], int root, int bottom);

void heapSort(int a[], int n) {

int i, temp;

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

siftDown(a, i, n - 1);

for (i = n - 1; i >= 1; i--) {

temp = a[0];

a[0] = a[i];

a[i] = temp;

siftDown(a, 0, i - 1);

}

}

void siftDown(int a[], int root, int bottom) {

int done, maxChild, temp;

done = 0;

while ((root \* 2 <= bottom) && (!done)) {

if (root \* 2 == bottom)

maxChild = root \* 2;

else if (a[root \* 2] > a[root \* 2 + 1]) // left child greater than right child

maxChild = root \* 2;

else

maxChild = root \* 2 + 1;

if (a[root] < a[maxChild]) { // exchange root with maxchild

temp = a[root];

a[root] = a[maxChild];

a[maxChild] = temp;

root = maxChild;

} else

done = 1;

}

}

int main() {

int a[100], n, i;

struct timeval tv1, tv2;

struct rusage r\_usage;

std::cout << "Enter the number of elements: " << std::endl;

std::cin >> n;

std::cout << "Enter the elements: " << std::endl;

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

std::cin >> a[i];

gettimeofday(&tv1, nullptr);

heapSort(a, n);

gettimeofday(&tv2, nullptr);

std::cout << "Sorted elements are: " << std::endl;

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

std::cout << a[i] << " ";

std::cout << "\nTime taken by Heap sort = " << static\_cast<double>(tv2.tv\_usec - tv1.tv\_usec) << " microseconds" << std::endl;

getrusage(RUSAGE\_SELF, &r\_usage);

std::cout << "Memory usage: " << r\_usage.ru\_maxrss << " kilobytes" << std::endl;

return 0;

}