Сравнить скорость двух Пирамидальных сортировок и выбрать подходящую, так как одна из них работает за О(n), а другая за О(nlogn).

1)

#include <iostream>

#include <vector>

using namespace std;

typedef int Date\_t;

void heap\_sort(int \*A, int num);

int main() {

Date\_t num;

cin >> num;

Date\_t mas[num];

for (Date\_t i = 0; i < num; i++){

cin >> mas[i];

}

heap\_sort(mas, num);

for (Date\_t i = 0; i < num; i++){

cout << mas[i] << " ";

}

cout << "\n";

return 0;

}

void heap\_sort(Date\_t \*A, Date\_t num){

for (size\_t k = 1; k < num; k++){

size\_t i = k;

size\_t parent\_index = (i - 1) / 2;

while(i != 0 and A[i] > A[parent\_index]){

swap(A[i], A[parent\_index]);

i = parent\_index;

parent\_index = (i - 1) / 2;

}

}

for (size\_t top = num - 1; top > 0; top--){

swap(A[0], A[top]);

size\_t i = 0;

size\_t left = 2 \* i + 1;

size\_t right = 2 \* i + 2;

while (((left < top) && (A[left] > A[i])) || ((right < top) && (A[right] > A[i]))) {

if (right >= top || (A[left] >= A[right])){

swap(A[left], A[i]);

i = left;

}else{

swap(A[right], A[i]);

i = right;

}

left = 2 \* i + 1;

right = 2 \* i + 2;

}

}

}

2)

#include <iostream>

using namespace std;

void heap(int arr[], int len, int i) {

int larget = i;

int left = 2\*i + 1;

int right = 2\*i + 2;

if(right < len && arr[right] > arr[larget]){

larget = right;

}

if(left < len && arr[left] > arr[larget]){

larget = left;

}

if(larget != i){

swap(arr[larget], arr[i]);

heap(arr, len, larget);

}

}

void buildHeap(int arr[], int len){

for(int i = len/2; i >=0; i--){

heap(arr, len, i);

}

}

void heapSort(int arr[], int len){

buildHeap(arr, len);

while(len > 0){

swap(arr[0], arr[len-1]);

len--;

heap(arr, len, 0);

}

}

int main() {

ios::sync\_with\_stdio(false);

cin.tie(nullptr);

int n;

cin >> n;

int arr[n];

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

cin >> arr[i];

}

heapSort(arr, n);

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

cout << arr[i] << " ";

}

return 0;

}