Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

 Лабораторная работа №15

По дисциплине «Основы алгоритмизации и программирования»

На тему «Анализ алгоритмов сортировок»

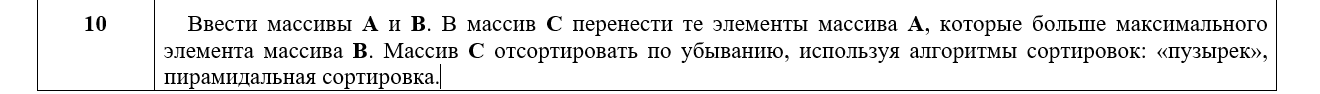
 Выполнила:

Студентка1 курса 6 группы

Литвинчук Дарья Валерьевна

Преподаватель: асс. Андронова М.В.

Минск, 2024



#include <iostream>

using namespace std;

void bubbleSort(int a[], int n) // пузырьковая сортировка

{

int i, j, t;

for (i = 1; i < n; i++)

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

if (a[j - 1] < a[j])

{

t = a[j - 1];

a[j - 1] = a[j];

a[j] = t;

}

}

void heapify(int A[], int pos, int n)

{

int t, tm;

while (2 \* pos + 1 < n)

{

t = 2 \* pos + 1;

if (2 \* pos + 2 < n && A[2 \* pos + 2] < A[t])

t = 2 \* pos + 2;

if (A[pos] > A[t])

{

tm = A[pos];

A[pos] = A[t];

A[t] = tm;

pos = t;

}

else

break;

}

}

void piramSort(int A[], int n) // пирамидальная сортировка

{

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

heapify(A, i, n);

while (n > 0)

{

int tm = A[0];

A[0] = A[n - 1];

A[n - 1] = tm;

n--;

heapify(A, 0, n);

}

}

int main()

{

setlocale(LC\_ALL, "rus");

int size;

cout << "Введите размеры массивов: ";

cin >> size;

int\* A = new int[size];

int\* B = new int[size];

int\* C = new int[size];

int max = INT\_MIN;

//заполняем массивы вручную так как это удобнее

cout << "Введите элементы в массив А: ";

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

{

cin >> A[i];

}

cout << endl << "Введите элементы в массив B: ";

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

{

cin >> B[i];

if (max < B[i]) // ищем максимальный

max = B[i];

}

int j = 0;

for (int i = 0; i < size; i++) // заполняем массив С

{

if (A[i] > max)

{

C[j] = A[i];

j++;

}

}

int\* C1 = new int[j];

for (int i = 0; i < j; i++) //дублируем массив для второй сортировки

{

C1[i] = C[i];

}

for (int i = 0; i < j; i++) // выводим до сортировки пузырьком

{

cout << C[i] << " ";

}

cout << endl;

bubbleSort(C, j); // сортируем пузырьком

for (int i = 0; i < j; i++) // выводим после сортировки

{

cout << C[i] << " ";

}

cout << endl<<endl;

for (int i = 0; i < j; i++) // выводим до сортировки

{

cout << C1[i] << " ";

}

cout << endl;

piramSort(C1, j); // пирамидальная сортировка

for (int i = 0; i < j; i++) // выводим после сортировки

{

cout << C1[i] << " ";

}

cout << endl;

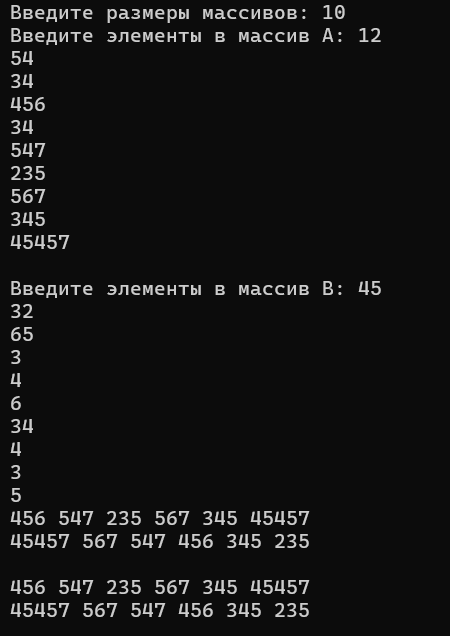
delete[] A;

delete[] B;

delete[] C;

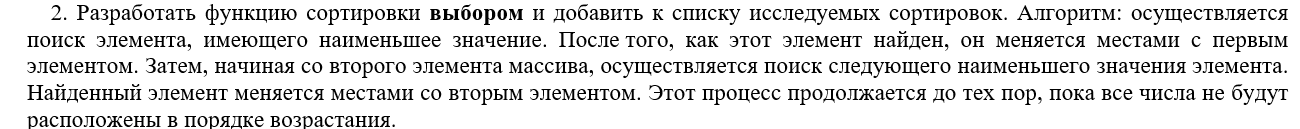
delete[] C1;

}



**Дополнительны задания**

Доп. 2



#include <iostream>

using namespace std;

void swap(int& a, int& b) // функция для обмена местами

{

int temp = a;

a = b;

b = temp;

}

void selectionSort(int arr[], int n) // функция сортировки

{

for (int i = 0; i < n - 1; i++)

{

int minIndex = i;

for (int j = i + 1; j < n; j++)

{

if (arr[j] < arr[minIndex])

{

minIndex = j;

}

}

// обмен значениями

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

}

}

int main()

{

setlocale(LC\_ALL, "rus");

int arr[] = { 123, 2565, 1672, 3422, 611, 76, 34, 67, 23457, 547345, 5574, 235, 4 };

int n = sizeof(arr) / 4;

cout << "Исходный массив: ";

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

{

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

}

cout << endl;

// вызов функции сортировки

selectionSort(arr, n);

// вывод отсортированного массива

cout << "Отсортированный массив: ";

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

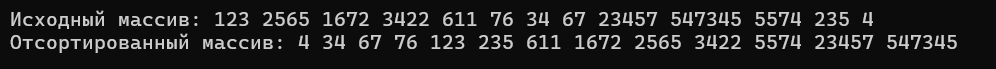
cout << arr[i] << " ";

}

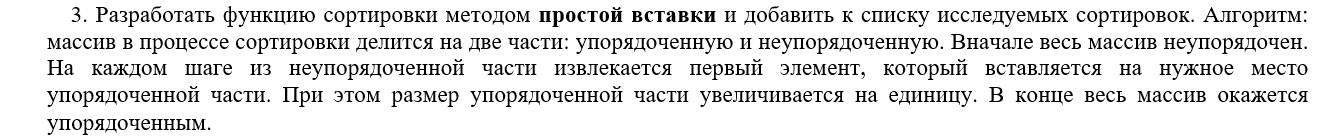
cout << endl;

return 0;

}



Доп. 3



#include <iostream>

using namespace std;

void insertionSort(int arr[], int n) // функция сортировки

{

for (int i = 1; i < n; i++)

{

int key = arr[i];

int j = i - 1;

while (j >= 0 && arr[j] > key)

{

arr[j + 1] = arr[j];

j = j - 1;

}

arr[j + 1] = key;

}

}

int main()

{

setlocale(LC\_ALL, "rus");

int arr[] = { 6, 4, 25, 52, 7, 132, 6, 34, 76, 34, 76, 3473, 7457, 23, 46, 23, 4, 6 };

int n = sizeof(arr) / 4;

cout << "Исходный массив: ";

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

{

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

}

cout << endl;

// вызов функции сортировки

insertionSort(arr, n);

// вывод отсортированного массива

cout << "Отсортированный массив: ";

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

{

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

}

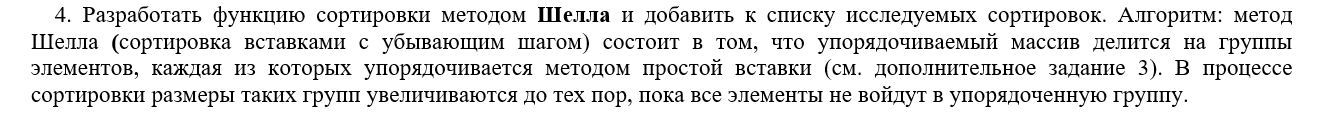
cout << endl;

return 0;

}



Доп. 4



#include <iostream>

using namespace std;

void shellSort(int arr[], int n) // функция сортировки

{

for (int gap = n / 2; gap > 0; gap /= 2) {

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

int temp = arr[i];

int j;

for (j = i; j >= gap && arr[j - gap] > temp; j -= gap) {

arr[j] = arr[j - gap];

}

arr[j] = temp;

}

}

}

int main()

{

setlocale(LC\_ALL, "rus");

int arr[] = { 12, 45, 96, 32, 47, 5, 4, 213, 645, 32, 646, 23, 122, 4657, 235, 325, 32, 125, 66};

int n = sizeof(arr) / 4;

cout << "Исходный массив: ";

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

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

}

cout << endl;

// вызов функции сортировки

shellSort(arr, n);

// вывод отсортированного массива

cout << "Отсортированный массив: ";

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

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

}

cout << endl;

return 0;

}

