**Министерство науки и высшего образования Российской Федерации**Федеральное государственное бюджетное образовательное учреждение высшего образования

**«Владимирский государственный университет**

**имени Александра Григорьевича и Николая Григорьевича Столетовых»**

**(ВлГУ)**

Институт прикладной математики, физики и информатики

Кафедра физики и прикладной математики

**ЛАБОРАТОРНАЯ РАБОТА №7**

по дисциплине  
«Основы программирования»  
на тему:  
**«Способы сортировки массивов»**

Выполнил:

ст. гр. ПМИ-123

Рушев А.М.

Принял:  
ст. преподаватель

каф. ФиПМ

Черников А.С.

Владимир, 2023 г.

**Цель работы:** изучение алгоритмов сортировки данных.

**Задание:**

Составить блок-схему и реализовать программное решение по представленной блок-схеме следующих способов сортировки данных.

1. Методом пузырька без улучшений.
2. Методом пузырька с тремя улучшениями: флаг перестановки, запоминание индекса последнего обмена, шейкер сортировка.
3. Простыми вставками.
4. Выбором.

**Практическая часть:**

**Задание 1.**

Листинг программы:

void task1() {

cout << "Задание 1. (Метод пузырька без улучшений)" << endl;

//инициализация

const int n = 5;

int arr[n];

//заполнение массива

cout << "Исходный массив:" << "\t";

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

arr[i] = 1 + rand() % 100;

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

}

cout << endl;

//сортировка пузырьком

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

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

if (arr[j] > arr[j + 1]) {

int temp = arr[j];

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

arr[j + 1] = temp;

}

}

}

//полученный массив

cout << "\nОтсортированный массив:" << "\t";

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

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

}

cout << endl;

}

Результат выполнения кода:

Изображение выглядит как текст, программное обеспечение, снимок экрана, компьютер

Автоматически созданное описание

Рисунок 1 – Результат выполнения кода задания 1

Блок-схема:

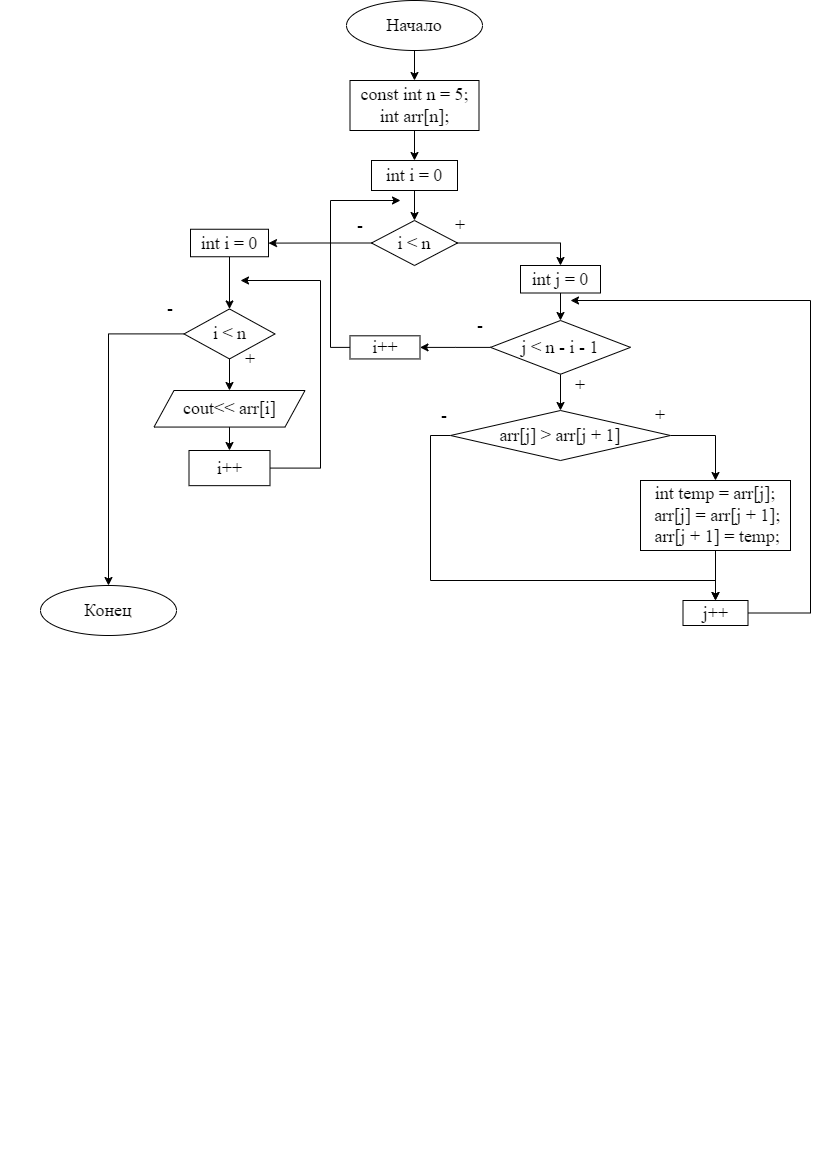


Рисунок 2 – Блок-схема задания 1

**Задание 2.**

Листинг программы:

void task2() {

cout << "\nЗадание 2. (Методом пузырька с тремя улучшениями: \n1)флаг перестановки;\n2)запоминание индекса последнего обмена;\n3)шейкер сортировка)." << endl;

cout << "\nФлаг перестановки" << endl;

//инициализация массива

const int n = 5;

int arrFlag[n];

//заполнение массива

cout << "Исходный массив:" << "\t";

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

arrFlag[i] = 1 + rand() % 100;

cout << arrFlag[i] << "\t";

}

cout << endl;

//сортировка пузырьком с флагом

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

bool flag = true;

for (int j = 1; j < n - 1; j++) {

if (arrFlag[j] > arrFlag[j + 1]) {

flag = false;

int temp = arrFlag[j];

arrFlag[j] = arrFlag[j + 1];

arrFlag[j + 1] = temp;

}

}

if (flag == false) {

break;

}

}

//отсортированный массив

cout << "Отсортированный массив:" << "\t";

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

cout << arrFlag[i] << "\t";

}

cout << endl;

cout << "\nЗапоминание индекса последнего обмена" << endl;

int arrIndex[n];

//заполнение массива

cout << "Исходный массив:" << "\t";

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

arrIndex[i] = 1 + rand() % 100;

cout << arrIndex[i] << "\t";

}

cout << endl;

//сортировка массива с индексом

int k = n - 1;

int index;

while (k > 0)

{

index = -1;

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

if (arrIndex[i] > arrIndex[i + 1]) {

int temp = arrIndex[i];

arrIndex[i] = arrIndex[i + 1];

arrIndex[i + 1] = temp;

index = i;

}

}

k = index;

}

//отсортированный массив

cout << "\nОтсортированный массив:" << "\t";

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

cout << arrIndex[i] << "\t";

}

cout << endl;

cout << "\nШейкер сортировка" << endl;

int arrSheiker[n];

//заполнение массива

cout << "Исходный массив:" << "\t";

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

arrSheiker[i] = 1 + rand() % 100;

cout << arrSheiker[i] << "\t";

}

cout << endl;

//сортировка массива шейкером

int last = 1, temp, left = 0, right = n - 1;

while (left < right) {

last = 0;

for (int i = left; i < right; i++) {

if (arrSheiker[i] > arrSheiker[i + 1]) {

temp = arrSheiker[i];

arrSheiker[i] = arrSheiker[i + 1];

arrSheiker[i + 1] = temp;

last = 1;

}

}

right--;

for (int i = right; i > left; i--) {

if (arrSheiker[I - 1] > arrSheiker[i]) {

temp = arrSheiker[i];

arrSheiker[i] = arrSheiker[i - 1];

arrSheiker[i - 1] = temp;

last = 1;

}

}

left++;

}

//отсортированный массив

cout << "\nОтсортированный массив:" << "\t";

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

cout << arrSheiker[i] << "\t";

}

cout << endl;

}

Результат выполнения кода:

Изображение выглядит как текст, снимок экрана, программное обеспечение, компьютер

Автоматически созданное описание

Рисунок 3 – Результат выполнения кода задания 2

Блок-схема:

Изображение выглядит как зарисовка, рисунок, текст, диаграмма

Автоматически созданное описание

Изображение выглядит как зарисовка, рисунок, текст, диаграмма

Автоматически созданное описание

Изображение выглядит как зарисовка, рисунок, текст, диаграмма

Автоматически созданное описание

Рисунок 4 – Блок-схема задания 2

**Задание 3.**

Листинг программы:

void task3() {

cout << "\nЗадание 3. (Простыми вставками)" << endl;

//инициализация

const int n = 5;

int arr[n];

//заполнение массива

cout << "Исходный массив:" << "\t";

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

arr[i] = 1 + rand() % 100;

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

}

cout << endl;

//сортировка вставкой

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

int temp = arr[i + 1];

int j = i;

while (j >= 0 && arr[j] > temp) {

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

j--;

}

arr[j + 1] = temp;

}

//отсортированный массив

cout << "\nОтсортированный массив:" << "\t";

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

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

}

cout << endl;

}

Результат выполнения кода:

Изображение выглядит как текст, снимок экрана, программное обеспечение, компьютер

Автоматически созданное описание

Рисунок 5 – Результат выполнения кода задания 3

Блок-схема:

Изображение выглядит как диаграмма, текст, План, зарисовка

Автоматически созданное описание

Рисунок 6 – Блок-схема задания 3

**Задание 4.**

Листинг программы:

void task4() {

cout << "\nЗадание 4. (Выбором)" << endl;

//инициализация

const int n = 5;

int arr[n];

//заполнение массива

cout << "Исходный массив:" << "\t";

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

arr[i] = 1 + rand() % 100;

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

}

cout << endl;

//сортировка выбором

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

int k = i;

int j = k + 1;

for (j; j < n; j++) {

if (arr[j] < arr[k]) {

k = j;

}

}

if (arr[i] != arr[k]) {

int temp = arr[i];

arr[i] = arr[k];

arr[k] = temp;

}

}

//вывод отсортированного массива

cout << "\nОтсортированный массив:" << "\t";

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

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

}

cout << endl;

}

Результат выполнения кода:

Изображение выглядит как текст, снимок экрана, программное обеспечение, компьютер

Автоматически созданное описание

Рисунок 7 – Результат выполнения кода задания 4

Блок-схема:

Изображение выглядит как диаграмма, План, Технический чертеж, зарисовка

Автоматически созданное описание

Рисунок 8 – Блок-схема задания 4

Листинг программы:

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "rus");

cout << "Лабораторная работа №7.\n";

task1();

task2();

task3();

task4();

system("pause");

return 0;

}

**Вывод:** были закреплены навыки составления блок-схем к программному коду, а также закреплены навыки способов сортировки данных в языке программирования C++.