**федеральное государственное автономное образовательное учреждение высшего образования**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**(ВЫСШАЯ ШКОЛА ПЕЧАТИ И МЕДИАИНДУСТРИИ)**

**(Факультет информационных технологий)**

***(Институт Принтмедиа и информационных технологий)***

***Кафедра Информатики и информационных технологий***

**направление подготовки**

**09.03.02 «Информационные системы и технологии»**

**ЛАБОРАТОРНАЯ РАБОТА № 7**

**Дисциплина:** Введение в программирование

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Тема:** Указатели

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Выполнил(а): студент(ка) группы:** 221-373

Гайдарь Максим Денисович

(Фамилия И.О.)

**Замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

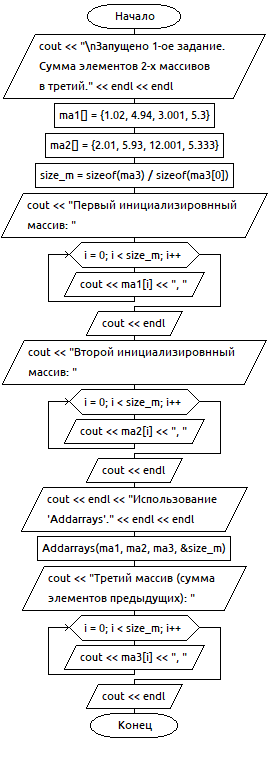
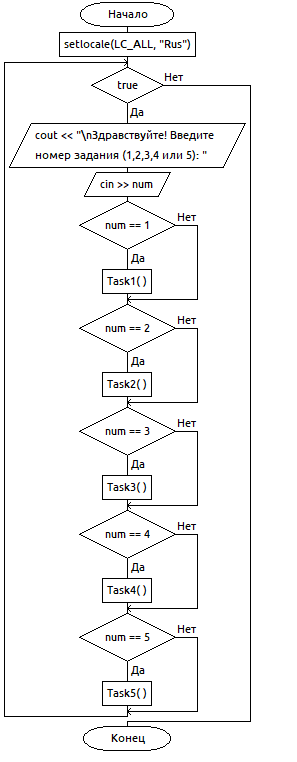
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Москва2022**

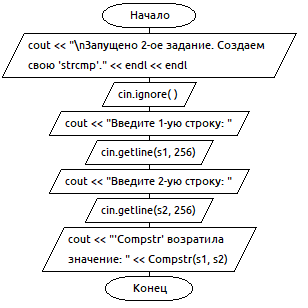
**Отчет по работе**

**Цикл выбора задания**

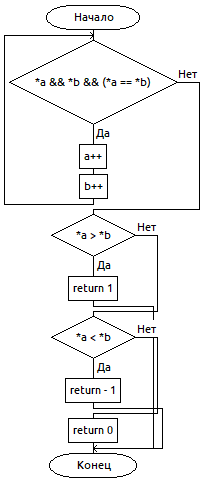
**1 задание**

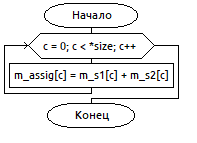


**2 задание**

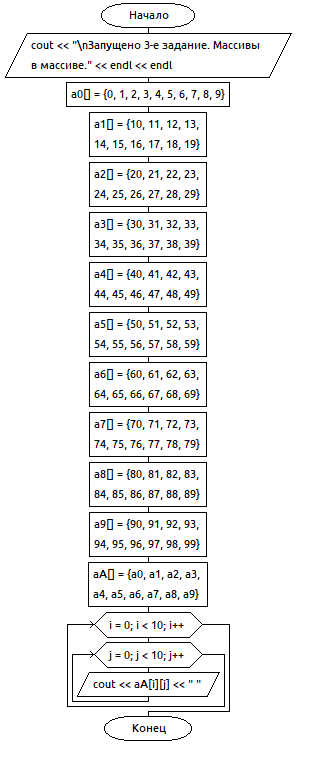
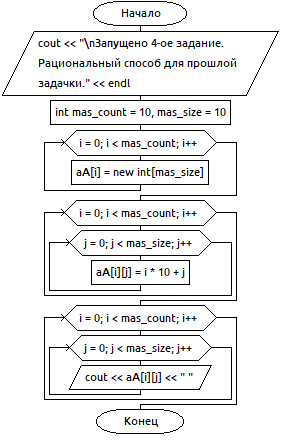
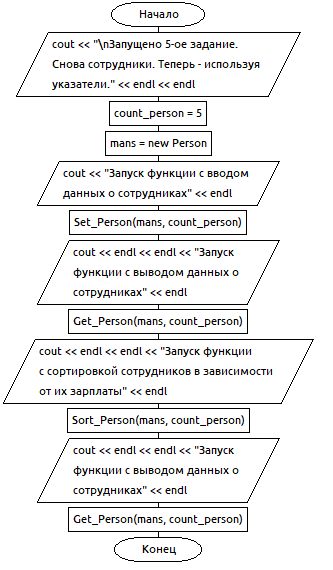


**Compstr()**





**Addarrays()**

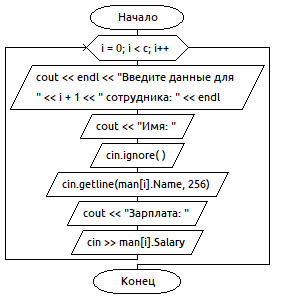
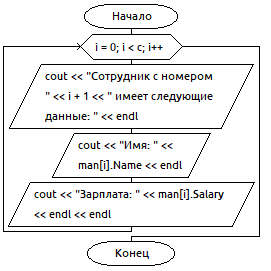
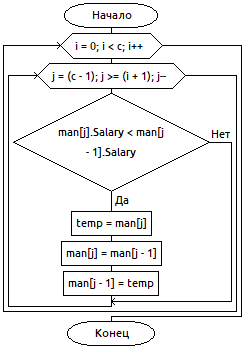
 

**4 задание**

**3 задание**

**Set\_Person()**

**5 задание**



**Soft\_Person()**

**Get\_Person()**

**Код программы:**

#include <iostream>

#include <iomanip>

#include <sstream>

using namespace std;

void Addarrays(float \*m\_s1, float \*m\_s2,float \*m\_assig, int \*size)

{

for (int c = 0; c < \*size; c++)

m\_assig[c] = m\_s1[c] + m\_s2[c];

}

void Task1()

{

cout << "\nЗапущено 1-ое задание. Сумма элементов 2-х массивов в третий." << endl << endl;

float ma1[] = { 1.02,4.94,3.001,5.3 };

float ma2[] = { 2.01,5.93,12.001,5.333 };

float ma3[4];

int size\_m = sizeof(ma3) / sizeof(ma3[0]);

cout << "Первый инициализировнный массив: ";

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

cout << ma1[i] << ", ";

cout << endl;

cout << "Второй инициализировнный массив: ";

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

cout << ma2[i] << ", ";

cout << endl;

cout << endl << "Использование 'Addarrays'." << endl << endl;

Addarrays(ma1, ma2, ma3, &size\_m);

cout << "Третий массив (сумма элементов предыдущих): ";

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

cout << ma3[i] << ", ";

cout << endl;

}

int Compstr(const char\* a, const char\* b)

{

while (\*a && \*b && (\*a == \*b))

{

a++; b++;

}

if (\*a > \*b) return 1;

if (\*a < \*b) return -1;

return 0;

}

void Task2()

{

cout << "\nЗапущено 2-ое задание. Создаем свою 'strcmp'." << endl << endl;

char s1[256], s2[256];

cin.ignore();

cout << "Введите 1-ую строку: "; cin.getline(s1, 256);

cout << "Введите 2-ую строку: "; cin.getline(s2, 256);

cout << "'Сompstr' возратила значение: " << Compstr(s1, s2);

}

void Task3()

{

cout << "\nЗапущено 3-е задание. Массивы в массиве." << endl << endl;

int a0[] = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 };

int a1[] = { 10, 11, 12, 13, 14, 15, 16, 17, 18, 19 };

int a2[] = { 20, 21, 22, 23, 24, 25, 26, 27, 28, 29 };

int a3[] = { 30, 31, 32, 33, 34, 35, 36, 37, 38, 39 };

int a4[] = { 40, 41, 42, 43, 44, 45, 46, 47, 48, 49 };

int a5[] = { 50, 51, 52, 53, 54, 55, 56, 57, 58, 59 };

int a6[] = { 60, 61, 62, 63, 64, 65, 66, 67, 68, 69 };

int a7[] = { 70, 71, 72, 73, 74, 75, 76, 77, 78, 79 };

int a8[] = { 80, 81, 82, 83, 84, 85, 86, 87, 88, 89 };

int a9[] = { 90, 91, 92, 93, 94, 95, 96, 97, 98, 99 };

int\* aA[] = { a0, a1, a2, a3, a4, a5, a6, a7, a8, a9 };

for (int i = 0; i < 10; i++)

{

for (int j = 0; j < 10; j++)

{

cout << aA[i][j] << " ";

}

}

}

void Task4()

{

cout << "\nЗапущено 4-ое задание. Рациональный способ для прошлой задачки." << endl;

const int mas\_count = 10, mas\_size = 10;

int\* aA[mas\_count];

for (int i = 0; i < mas\_count; i++)

{

aA[i] = new int[mas\_size];

}

for (int i = 0; i < mas\_count; i++)

{

for (int j = 0; j < mas\_size; j++)

{

aA[i][j] = i\*10 + j;

}

}

for (int i = 0; i < mas\_count; i++)

{

for (int j = 0; j < mas\_size; j++)

{

cout << aA[i][j] << " ";

}

}

}

struct Person {

char Name[256];

float Salary;

};

void Set\_Person(Person\* man, int c)

{

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

{

cout << endl << "Введите данные для " << i + 1 << " сотрудника: " << endl;

cout << "Имя: "; cin.ignore(); cin.getline(man[i].Name , 256);

cout << "Зарплата: "; cin >> man[i].Salary;

}

}

void Get\_Person(Person\* man, int c)

{

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

{

cout << "Сотрудник с номером " << i + 1 << " имеет следующие данные: " << endl;

cout << "Имя: " << man[i].Name << endl;

cout << "Зарплата: " << man[i].Salary << endl << endl;

}

}

void Sort\_Person(Person\* man, int c)

{

Person temp{};

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

for (int j = (c - 1); j >= (i + 1); j--) {

if (man[j].Salary < man[j - 1].Salary) {

temp = man[j];

man[j] = man[j - 1];

man[j - 1] = temp;

}

}

}

}

void Task5()

{

cout << "\nЗапущено 5-ое задание. Снова сотрудники. Теперь - используя указатели." << endl << endl;

int count\_person = 5;

Person\* mans = new Person;

cout << "Запуск функции с вводом данных о сотрудниках" << endl; Set\_Person(mans, count\_person);

cout << endl << endl << "Запуск функции с выводом данных о сотрудниках" << endl; Get\_Person(mans, count\_person);

cout << endl << endl << "Запуск функции с cортировкой сотрудников в зависимости от их зарплаты" << endl; Sort\_Person(mans, count\_person);

cout << endl << endl << "Запуск функции с выводом данных о сотрудниках" << endl; Get\_Person(mans, count\_person);

delete mans;

}

int main()

{

setlocale(LC\_ALL, "Rus");

int num;

while (true)

{

cout << "\nЗдравствуйте! Введите номер задания (1,2,3,4 или 5): "; cin >> num;

if (num == 1)

{

Task1();

}

if (num == 2)

{

Task2();

}

if (num == 3)

{

Task3();

}

if (num == 4)

{

Task4();

}

if (num == 5)

{

Task5();

}

}

}

**Результат работы:**

