**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 № 5**

**Дисциплина:** Введение в программирование

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

**Тема:** Массивы и строки

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

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

Гайдарь Максим Денисович

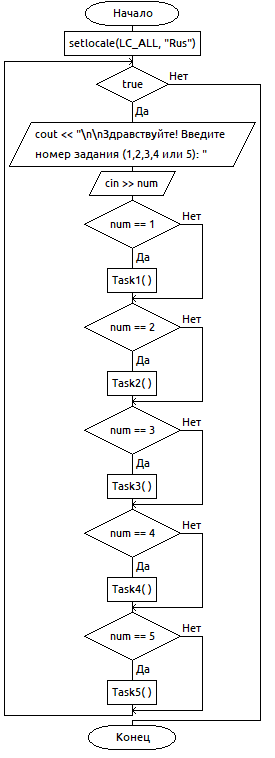
(Фамилия И.О.)

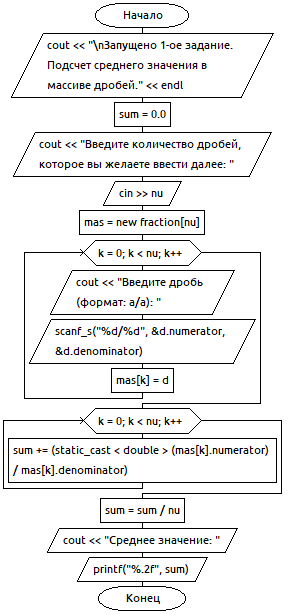
**Замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

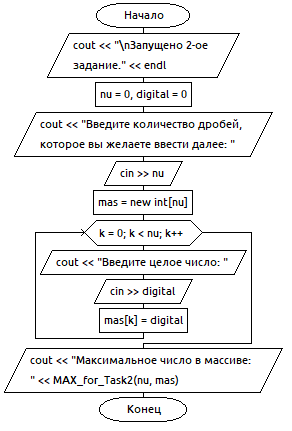
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

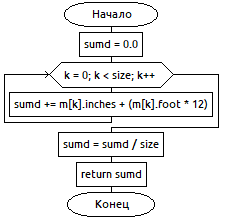
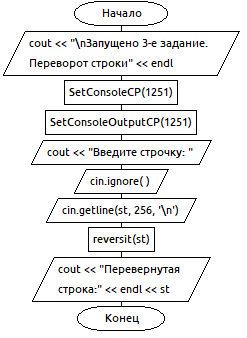
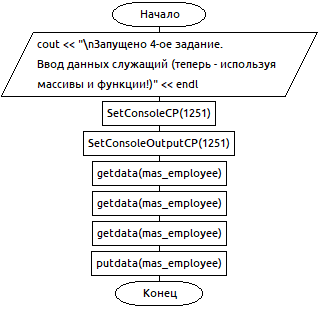
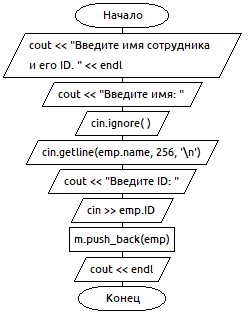
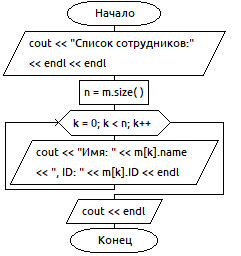
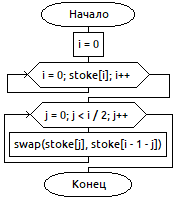
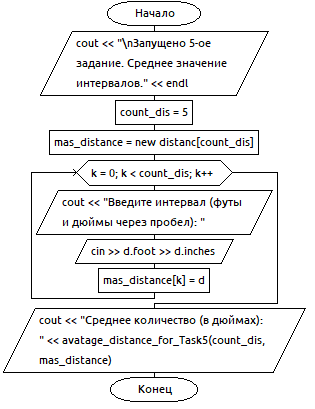
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Москва2022**

  **Отчет по работе**





**Код программы:**

#include <iostream>

#include <Windows.h>

#include <vector>

using namespace std;

void Task1()

{

cout << "\nЗапущено 1-ое задание. Подсчет среднего значения в массиве дробей." << endl;

struct fraction

{

int numerator; //Числитель

int denominator; //Знаменатель

};

int nu;

double sum = 0.0;

fraction d;

cout << "Введите количество дробей, которое вы желаете ввести далее: "; cin >> nu;

fraction\* mas = new fraction[nu];

for (int k = 0; k < nu; k++)

{

cout << "Введите дробь (формат: a/a): "; scanf\_s("%d/%d", &d.numerator, &d.denominator);

mas[k] = d;

}

for (int k = 0; k < nu; k++)

{

sum += (static\_cast<double>(mas[k].numerator) / mas[k].denominator);

}

sum = sum / nu;

cout << "Среднее значение: "; printf("%.2f", sum);

}

int MAX\_for\_Task2(int size, int m[])

{

int index = 0;

int max = -2147483648;

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

{

if (max < m[k])

{

max = m[k];

index = k;

}

}

return index;

}

void Task2()

{

cout << "\nЗапущено 2-ое задание. Поиск наибольшего числа и его индекса в массиве" << endl;

int nu = 0,digital = 0;

cout << "Введите количество целых чисел, которое вы желаете ввести далее: "; cin >> nu;

int\* mas = new int[nu];

for (int k = 0; k < nu; k++)

{

cout << "Введите целое число: "; cin >> digital;

mas[k] = digital;

}

cout << "Индекс максимального числа в массиве: " << MAX\_for\_Task2(nu, mas) << ", максимальное число: " << mas[MAX\_for\_Task2(nu, mas)];

}

void reversit(char stoke[])

{

int i = 0;

for (i = 0; stoke[i]; i++);

for (int j = 0; j < i / 2; j++)

swap(stoke[j], stoke[i - 1 - j]);

}

void Task3()

{

cout << "\nЗапущено 3-е задание. Переворот строки" << endl;

char st[256];

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

cout << "Введите строчку: ";

cin.ignore();

cin.getline(st, 256, '\n');

reversit(st);

cout << "Перевернутая строка:" << endl << st;

}

struct employee

{

char name[256];

long ID;

};

void getdata(vector<employee> &m)

{

employee emp;

cout << "Введите имя сотрудника и его ID. " << endl;

cout << "Введите имя: "; cin.ignore(); cin.getline(emp.name, 256, '\n');

cout << "Введите ID: "; cin >> emp.ID;

m.push\_back(emp);

cout << endl;

}

void putdata(vector<employee> &m)

{

cout << "Список сотрудников:" << endl << endl;

int n = m.size();

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

cout << "Имя: " << m[k].name << ", ID: " << m[k].ID << endl;

cout << endl;

}

void Task4()

{

cout << "\nЗапущено 4-ое задание. Ввод данных служащий (теперь - используя массивы и функции!)" << endl;

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

std::vector<employee> mas\_employee;

getdata(mas\_employee);

getdata(mas\_employee);

getdata(mas\_employee);

putdata(mas\_employee);

}

struct distanc

{

int foot;

int inches;

};

float avatage\_distance\_for\_Task5(int size, struct distanc m[])

{

float sumd = 0.0;

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

{

sumd += m[k].inches + (m[k].foot \* 12);

}

sumd = sumd / size;

return sumd;

}

void Task5()

{

cout << "\nЗапущено 5-ое задание. Среднее значение интервалов." << endl;

int count\_dis = 5; //Количество вводимых интервалов (по заданию - 5)

distanc\* mas\_distance = new distanc[count\_dis];

distanc d;

for (int k = 0; k < count\_dis; k++)

{

cout << "Введите интервал (футы и дюймы через пробел): "; cin >> d.foot >> d.inches;

mas\_distance[k] = d;

}

cout << "Среднее количество (в дюймах): " << avatage\_distance\_for\_Task5(count\_dis, mas\_distance);

}

int main()

{

setlocale(LC\_ALL, "Rus");

int num;

while (true)

{

cout << "\n\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:**

