**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 № 3**

**Дисциплина:** Введение в программирование

**Тема:** Структуры и перечисления

**Выполнил(а): студент(ка) группы 221-377**

Костоваров А.С.

****** (Фамилия И.О.)

**Дата, подпись** 30.09.2022 ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Проверил: \_\_*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_***

(Фамилия И.О., степень, звание) **(Оценка)**

**Дата, подпись** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

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

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

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

**Москва2022**

**Структуры и перечисления**

**Цель:** Получить практические навыки в создании программ, использующих структуры и перечисления

**Постановка задачи 1**

Создайте структуру с именем employee, содержащую два поля: номер сотрудника типа int и величину его зарплаты в рублях типа float. Запросите с клавиатуры данные о трех сотрудниках, сохраните их в трех структурных переменных типа employee и выведите информацию о каждом из сотрудников на экран.

**Код:**

#include <iostream>

using namespace std;

struct employee

{

int number;

float money;

} m1, m2, m3;

int main()

{

setlocale(LC\_ALL, "Russian");

cout << "Введите номер сотрудника: ";

cin >> m1.number;

cout << "Введите величину его зарплаты в рублях: ";

cin >> m1.money;

cout << "Введите номер сотрудника: ";

cin >> m2.number;

cout << "Введите величину его зарплаты в рублях: ";

cin >> m2.money;

cout << "Введите номер сотрудника: ";

cin >> m3.number;

cout << "Введите величину его зарплаты в рублях: ";

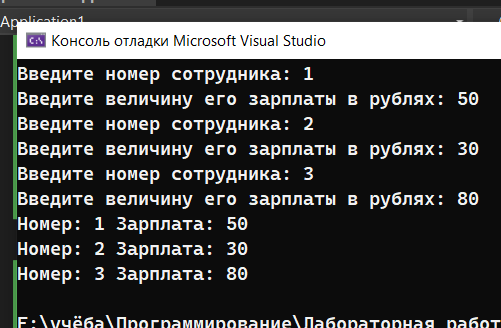
cin >> m3.money;

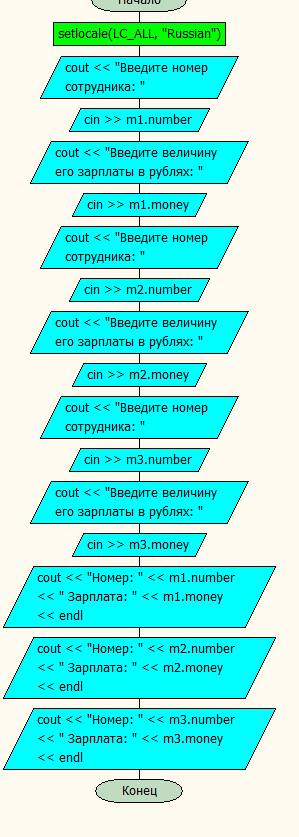
cout << "Номер: " << m1.number << " Зарплата: " << m1.money << endl;

cout << "Номер: " << m2.number << " Зарплата: " << m2.money << endl;

cout << "Номер: " << m3.number << " Зарплата: " << m3.money << endl;

}

**Вывод:**

** Блок-схема**

**Постановка задачи 2**

Стандартные средства ввода/вывода C++ вместо значений перечисляемых типов данных выводят их внутреннее представление в виде целых чисел. Для того чтобы преодолеть это ограничение, вы можете использовать конструкцию switch, с помощью которой устанавливается соответствие между значением переменной перечисляемого типа и се внутренним представлением. Пусть, например, в программе определен перечисляемый тип данных etype, отражающий должность сотрудника: enum etype { laborer, secretary, manager, accountant, executive, researcher };

Напишите программу, которая сначала по первой букве должности, введенной пользователем, определяет соответствующее значение переменной, помещает это значение в переменную типа etype, а затем выводит полностью название должности, первую букву которой ввел пользователь. Взаимодействие программы с пользователем может выглядеть следующим образом:

Введите первую букву должности

(laborer, secretary, manager, accountant, executive, researcher): a

Полное название должности; accountant

Возможно, вам понадобится два ветвления switch: одно — для ввода значения, другое — для вывода.

**Код:**

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Russian");

enum etype { laborer, secretary, manager, accountant, executive, researcher };

etype a;

char w;

cout << " Введите первую букву должности \n"

"(laborer, secretary, manager, accountant, executive, researcher) : \n";

cin >> w;

switch (w)

{

case 'l':

a = laborer;

cout << "Полное название должности: laborer";

break;

case 's':

a = laborer;

cout << "Полное название должности: secretary";

break;

case 'm':

a = laborer;

cout << "Полное название должности: manager";

break;

case 'a':

a = laborer;

cout << "Полное название должности: accountant";

break;

case 'e':

a = laborer;

cout << "Полное название должности: executive";

break;

case 'r':

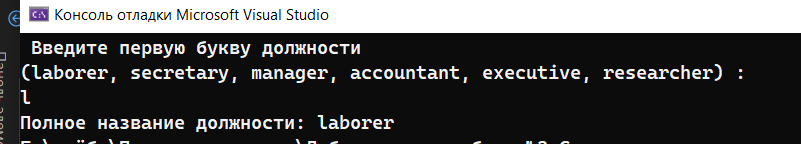
a = laborer;

cout << "Полное название должности: researcher";

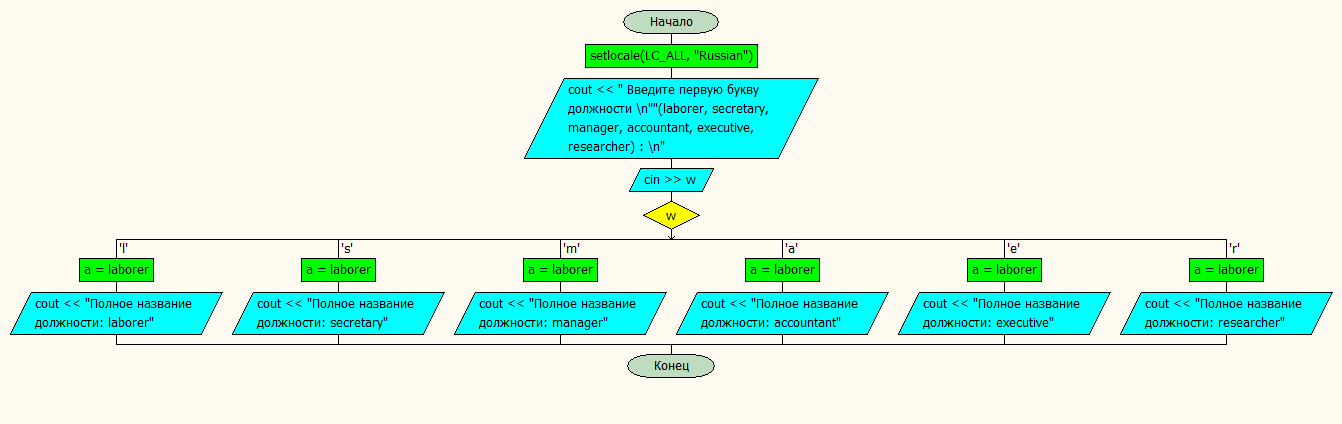
break;

}

}

**Вывод:** ****

**Блок-схема**



**Постановка задачи 3**

Вернитесь к упражнению 3 ЛР1. В этом упражнении требуется написать программу, которая хранит значения двух дробей в виде числителя и знаменателя, а затем складывает эти дроби согласно арифметическому правилу. Измените эту программу так, чтобы значения дробей хранились в структуре fraction, состоящей из двух полей типа int. предназначенных для хранения числителя и знаменателя. Все значения дробей должны храниться в переменных типа fraction.

**Код:**

#include <iostream>

using namespace std;

struct fraction {

int chislitel;

int znamenatel;

int czeloe;

};

int main()

{

fraction d1, d2, d3;

setlocale(LC\_ALL, "Russian");

char s = '/';

cout << "Введите первую дробь: ";

cin >> d1.chislitel >> s >> d1.znamenatel;

cout << "Введите вторую дробь: ";

cin >> d2.chislitel >> s >> d2.znamenatel;

if (d1.znamenatel != d2.znamenatel) {

d3.znamenatel = d1.znamenatel \* d2.znamenatel;

d3.chislitel = d1.chislitel \* d2.znamenatel + d2.chislitel \* d1.znamenatel;

cout << "Сумма дробей:";

cout << d3.chislitel << s << d3.znamenatel;

}

else {

d3.chislitel = d1.chislitel + d2.chislitel;

d3.znamenatel = d1.znamenatel;

cout << "Сумма дробей:";

cout << d3.chislitel << s << d3.znamenatel;

}

if (d3.chislitel >= d3.znamenatel) {

d3.czeloe = d3.chislitel / d3.znamenatel;

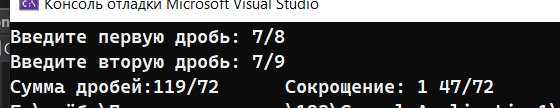
d3.chislitel = d3.chislitel % d3.znamenatel;

cout << " Сокрощение: ";

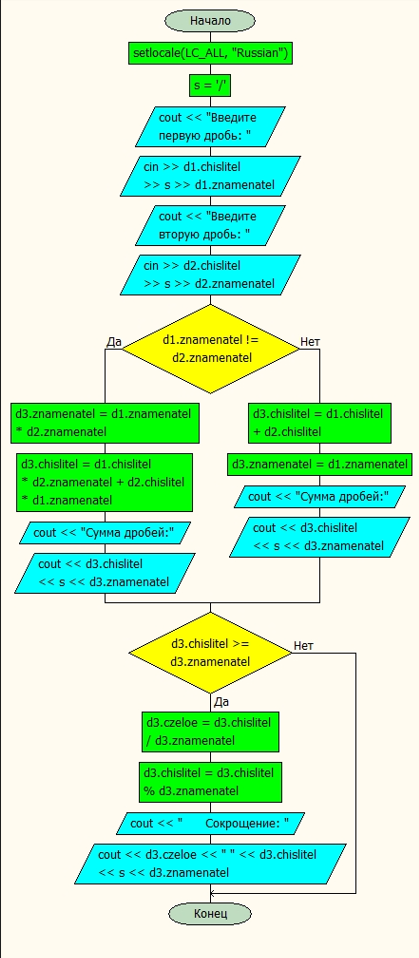
cout << d3.czeloe << " " << d3.chislitel << s << d3.znamenatel;

}

}

**Вывод:** 

**Блок-схема**

****

**Постановка задачи 4**

Создайте структуру с именем time. Три ее поля, имеющие тип int. будут называться hours, minutes и seconds. Напишите программу, которая просит пользователя ввести время в формате часы, минуты, секунды. Можно запрашивать на ввод как три значения сразу, так и выводить для каждой величины отдельное приглашение.

Программа должна хранить время в структурной переменной типа time и выводить количество секунд в введенном времени, определяемое следующим образом:

long totalsecs = t1.hours\*3600 + t1 minutes\*60 + t1.seconds

**Код:**

struct time {

public: int hours;

public: int minutes;

public: int seconds;

public: int totalsecs;

}t1;

#include <iostream>

using namespace std;

int main()

{

int long totalsecs;

char h;

setlocale(LC\_ALL, "Russian");

cout << "Введите время в формате часы.минуты.секунды.\n";

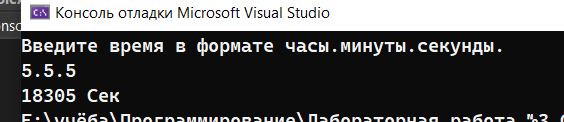
cin >> (t1.hours) >> h >> (t1.minutes) >> h >> (t1.seconds);

t1.totalsecs = t1.hours \* 3600 + t1.minutes \* 60 + t1.seconds;

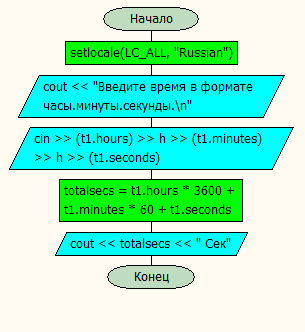
cout << t1.totalsecs << " Сек";

}

**Вывод:**

****

**Блок-схема**

****

**Постановка задачи 5**

Используя структуру time из предыдущего упражнения, напишите программу, которая получает от пользователя два значения времени в формате 12:59:59, сохраняет их в переменных типа struct time, затем переводит оба значения в секунды, складывает их, переводит сумму в исходный формат, сохраняет его в переменной типа time и выводит полученный результат на экран в формате 12:59:59.

**Код:**

struct time {

public: int hours;

public: int minutes;

public: int seconds;

public: int totalsecs;

}t1, t2 ,t3;

#include <iostream>

using namespace std;

int main()

{

int long totalsecs;

char h;

setlocale(LC\_ALL, "Russian");

cout << "Введите время в формате часы.минуты.секунды.\n";

cin >> (t1.hours) >> h >> (t1.minutes) >> h >> (t1.seconds);

t1.totalsecs = t1.hours \* 3600 + t1.minutes \* 60 + t1.seconds;

cout << "Введите время в формате часы.минуты.секунды.\n";

cin >> (t2.hours) >> h >> (t2.minutes) >> h >> (t2.seconds);

t2.totalsecs = t2.hours \* 3600 + t2.minutes \* 60 + t2.seconds;

t3.totalsecs = t1.totalsecs + t2.totalsecs;

t3.hours = t3.totalsecs / 3600;

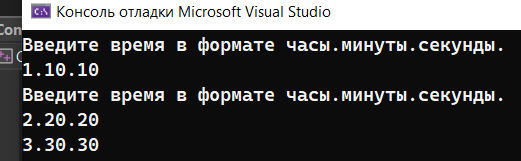
t3.minutes = (t3.totalsecs % 3600) / 60;

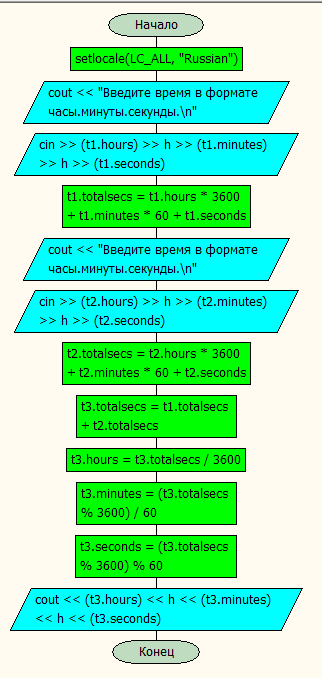
t3.seconds = (t3.totalsecs % 3600) % 60;

cout << (t3.hours) << h << (t3.minutes) << h << (t3.seconds);

}

**Вывод:**

****

** Блок-схема**