**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 №4**

**Дисциплина:** Введение в программирование

**Тема: Функции**

**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**

**Тема**

**Функции**

**Цель:** **Получить практические навыки в создании функций и их последующем использовании.**

**Постановка задачи 1**

Напишите функцию с именем hms\_to\_secs(), имеющую три аргумента типа int: часы, минуты и секунды. Функция должна возвращать эквивалент переданного ей временного значения в секундах (типа long). Создайте программу, которая будет циклически запрашивать у пользователя ввод значения часов, минут и секунд и выводить результат работы функции на экран.

**Код:**

#include <iostream>

using namespace std;

long hms\_to\_secs(int hours, int minutes, int seconds) {

long allsecs;

allsecs = hours \* 3600 + minutes \* 60 + seconds;

return allsecs;

}

int main()

{

int hours; int minutes; int seconds;

char h;

setlocale(LC\_ALL, "Russian");

int i = 1;

if (i != 0)

{

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

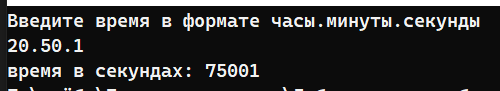
cin >> (hours) >> h >> (minutes) >> h >> (seconds);

cout << "время в секундах: " << hms\_to\_secs(hours, minutes, seconds);

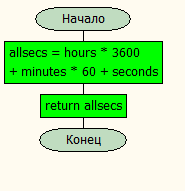
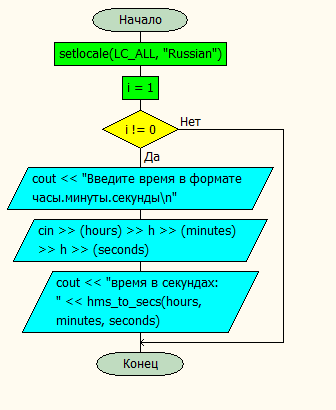
}

}

**Вывод:**

****

**Блок-схема**

****

**Постановка задачи 2**

Модифицируйте программу, описанную в упражнении 5 предыдущей лабораторной работы «Структуры и перечисления», складывающую два структурных значения типа time. Теперь программа должна включать в себя две функции. Первая, time\_to\_secs(), принимает в качестве аргумента значение типа time и возвращает эквивалентное значение в секундах типа long. Вторая, secs\_to\_time(), в качестве аргумента принимает число секунд, имеющее тип long, а возвращает эквивалентное значение типа time.

**Код:**

#include <iostream>

using namespace std;

struct Time {

public: int hours;

public: int minutes;

public: int seconds;

public: int totalsecs;

void info() {

cout << "сумма: " << hours << "." << minutes << "." << seconds;

}

}t1, t2, t3;

long time\_to\_secs(int hours, int minutes, int seconds) {

long totalsecs = hours \* 3600 + minutes \* 60 + seconds;

return totalsecs;

}

Time secs\_to\_time(int totalsecs) {

long hours = totalsecs / 3600;

long minutes = (totalsecs % 3600) / 60;

long seconds = (totalsecs % 3600) % 60;

Time final;

final.hours = hours;

final.minutes = minutes;

final.seconds = seconds;

return final;

}

int main()

{

int long totalsecs;

int l\_hours;

int l\_minutes;

int l\_seconds;

char h;

int a;

setlocale(LC\_ALL, "Russian");

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

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

t1.totalsecs = time\_to\_secs(t1.hours, t1.minutes, t1.seconds);

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

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

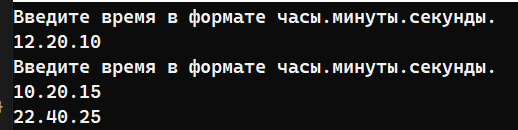
t2.totalsecs = time\_to\_secs(t2.hours, t2.minutes, t2.seconds);

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

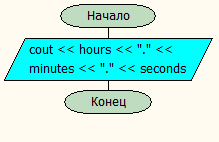
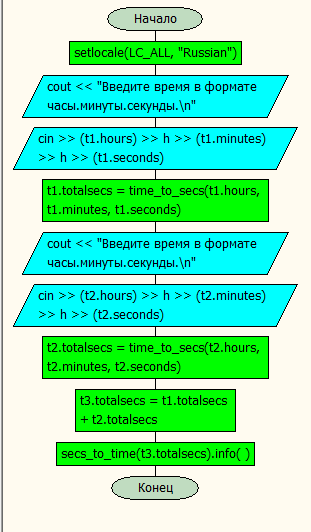
secs\_to\_time(t3.totalsecs).info();

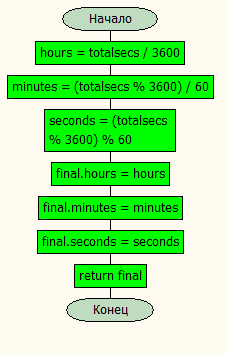
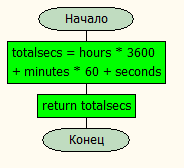
}

**Вывод:**

****

**Блок-схема**

****

****

**Постановка задачи 3**

Напишите функцию с именем swap(), обменивающую значениями два своих аргумента типа int (обратите внимание, что изменяться должны значения переменных из вызывающей функции, а не локальных переменных функции). Выберите способ передачи аргументов. Напишите вызывающую программу main(), использующую данную функцию.

**Код:**

#include <iostream>

using namespace std;

void Swap(int &first, int &second) {

int buf = first;

first = second;

second = buf;

}

int main() {

setlocale(LC\_ALL, "Russian");

int firstValue, secondValue;

cout << "Введите два числа для операции с ними(через enter)" << endl;

cin >> firstValue >> secondValue;

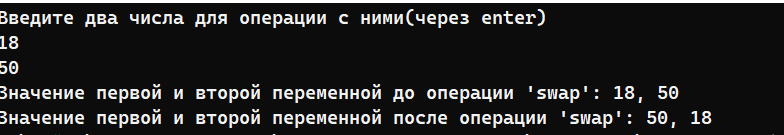
cout << "Значение первой и второй переменной до операции 'swap': " << firstValue << ", " << secondValue << endl;

Swap(firstValue, secondValue);

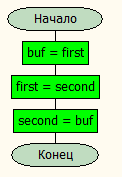
cout << "Значение первой и второй переменной после операции 'swap': " << firstValue << ", " << secondValue;

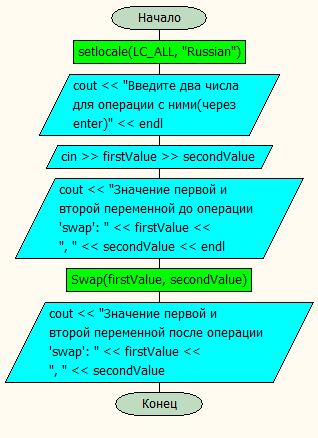
}

**Вывод:**

****

**Блок-схема**

****

**с**

**Постановка задачи 4**

Переработайте программу из предыдущего упражнения 3 так, чтобы функция swap() принимала в качестве аргументов значения типа time (см. упражнение 2).

**Код:**

#include <iostream>

using namespace std;

void Swap(int& first, int& second) {

int buf = first;

first = second;

second = buf;

}

struct Time {

long hours;

long minutes;

long seconds;

static void SwapTime(Time& time1, Time& time2) {

Time buf;

buf = time1;

time1 = time2;

time2 = buf;

}

void GetInfo() {

cout << hours << ":" << minutes << ":" << seconds;

}

};

int main() {

setlocale(LC\_ALL, "Russian");

Time firstTime, SecondTime;

cout << "Введите первое значение времени(через enter)" << endl;

cin >> firstTime.hours >> firstTime.minutes >> firstTime.seconds;

cout << "Введите второе значение времени(через enter)" << endl;

cin >> SecondTime.hours >> SecondTime.minutes >> SecondTime.seconds;

cout << "Значение первой и второй переменной до операции 'swap': ";

firstTime.GetInfo();

cout << "||";

SecondTime.GetInfo();

cout << endl;

Time::SwapTime(firstTime, SecondTime);

cout << "Значение первой и второй переменной после операции 'swap': ";

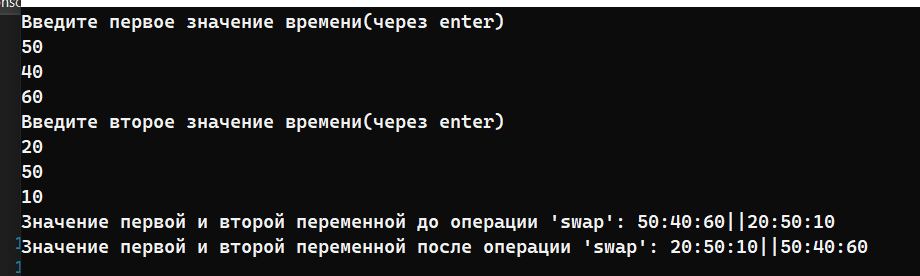
firstTime.GetInfo();

cout << "||";

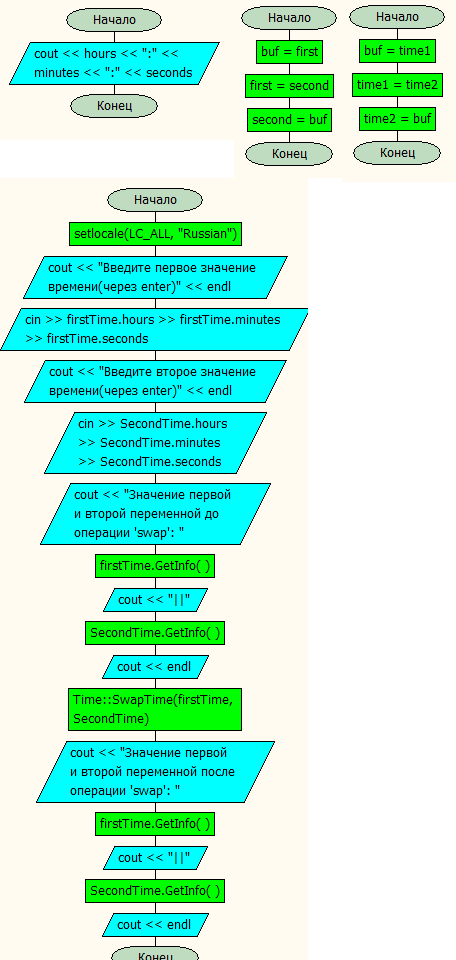
SecondTime.GetInfo();

cout << endl;

}

**Вывод:**

**Блок-схема**



**Постановка задачи 5**

Напишите функцию, которая при каждом вызове будет выводить на экран количество раз, которое она вызывалась ранее. Напишите программу, которая будет вызывать данную функцию не менее 10 раз. Попробуйте реализовать данную функцию двумя различными способами: с использованием глобальной переменной и статической локальной переменной для хранения числа вызовов функции. Какой из способов предпочтительней? Почему для решения задачи нельзя использовать обычную локальную переменную?

**Код:**

#include <iostream>

using namespace std;

int counter = 0;

void CallCounter() {

static int staticCounter = 0;

counter++;

staticCounter++;

cout << "Количество вызовов(глобальная переменная) : " << counter << endl;

cout << "Количество вызовов(статическая переменная) : " << staticCounter << endl;

return;

}

int main() {

setlocale(LC\_ALL, "Russian");

int n;

cout << "Введите количество вызовов функции: ";

cin >> n;

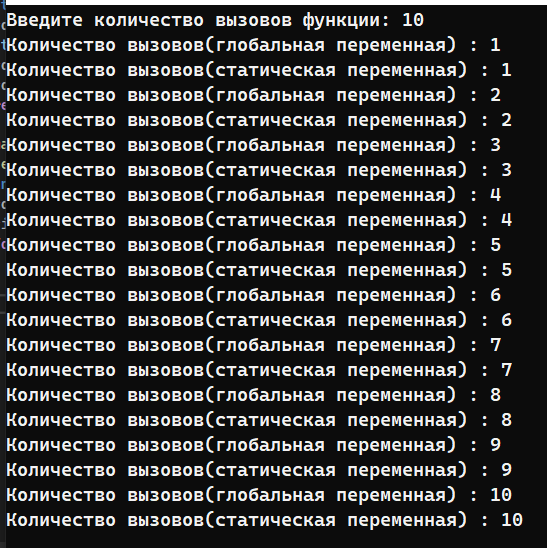
for (int i = 0; i < n; i++) {

CallCounter();

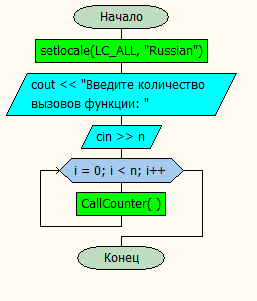
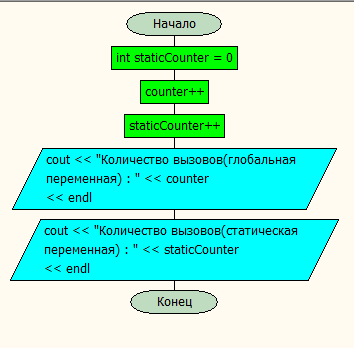
}

}

**Вывод:**

****

**Блок-схема**

****