МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Нижегородский государственный архитектурно-строительный университет»

*Кафедра информационных систем и технологий*

**Курсовая работа по дисциплине**

**«Язык программирования C/С++» на тему:**

***«Создание консольного приложения, написание нестатического класса***

***выполняющего сложение и вычитание двух чисел в системе счисления 9. Из большего числа должно вычитаться меньшее.***

Выполнил студент гр. ИС-29: Оборожный Д.С.

Проверил доцент, канд. техн.наук: Кислицын Д. И.

Нижний Новгород

2021 г.

## Содержание

[Задание 3](#_Toc13637)

[Теоретическая часть 4](#_Toc13638)

[Блок-схема 5-7](#_Toc13639)

[Программный код 8-10](#_Toc13640)

[Пример работы приложения 11](#_Toc13641)

[[Список использованной литературы 12](#_Toc13641)](#_Toc13641)

# Задание

Напишите класс s9, выполняющий сложение и вычитание двух чисел в системе счисления 9. Из большего числа должно вычитаться меньшее. Заранее не известно, какое число больше. Вход (в конструктор): две строки типа string. Выход (методы): string plus() – результат сложения двух чисел/ Неверный формат исходных данных string minus() – результат вычитания из большего числа меньшего/ Неверный формат исходных данных  
*Пример работы:*

Вход: 121, 2

Выход:

plus() -> 123

minus() -> 118

Вход: 2, 121

Выход:

plus() -> 123

minus() -> 118

Вход: 121, 2z

Выход:

plus() -> Неверный формат исходных данных

minus() -> Неверный формат исходных данных

# Теоретическая часть

Для перевода числа из любой системы счисления в десятичную достаточно пронумеровать его разряды, начиная с нулевого (разряд слева от десятичной точки) аналогично примерам 1 или 2. Найдём сумму произведений цифр числа на основание системы счисления в степени позиции этой цифры:

**1.** Перевести число 1001101.11012 в десятичную систему счисления.  
**Решение:** 1001101.11012 = 1·26+0·25+0·24+1·23+1·22+0·21+1·20+1·2-1+1·2-2+0·2-3+1·2-4 = 64+8++4+1+0.5+0.25+0.0625 = 77.812510  
**Ответ:** 1001101.11012 = 77.812510

Для перевода из десятичной СС делим десятичное число на основание системы, в которую хотим перевести и записываем остатки от деления. Запишем полученные остатки в обратном порядке и получим искомое число.

Переведем число 37510 в восьмеричную систему:

375 / 8 = 46 (остаток 7)

46 / 8 = 5 (остаток 6)

5 / 8 = 0 (остаток 5)

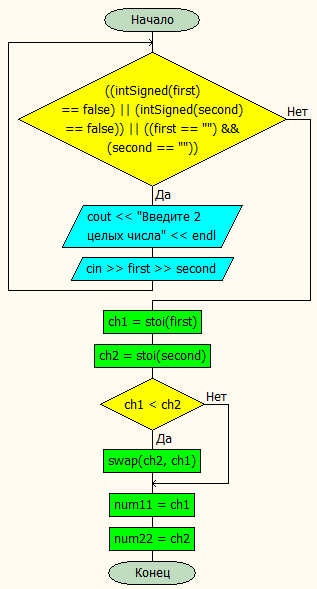
Записываем остатки и получаем 5678

# Блок-схемы

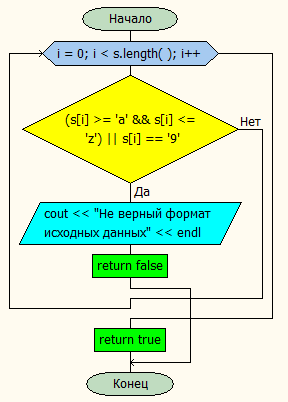
Блок схема функции main()

# 

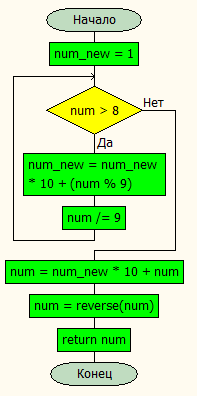
Схема функции set\_numbers() класса s9



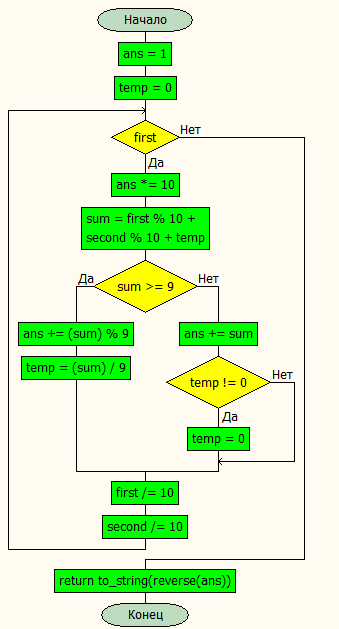
Функция intSigned() класса s9



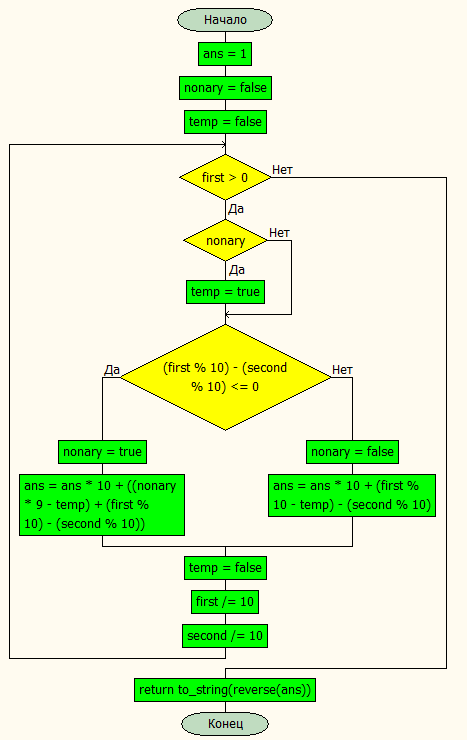
Функция to\_ninefold() класса s9



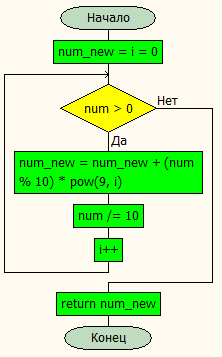
Функция Plus() класса s9



Функция Minus() класса s9



Функция to\_tenfold() класса s9



Функция reverse() класса s9

# 

# Программный код

#include <iostream>

#include <cstdlib>

#include <string>

#include <locale>

#include <cmath>

using namespace std;

class s9

{

public:

void set\_numbers(int &num11, int &num22)

{

string first, second;

while (((intSigned(first) == false) || (intSigned(second) == false)) || ((first == "") && (second == "")))//проверка корректности вводимых данных, работает до тех пор пока пользователь не введёт числа

{

cout << "Введите 2 целых числа" << endl;

cin >> first >> second;

}

ch1 = stoi(first);

ch2 = stoi(second);

if (ch1 < ch2)//изначально считается, что ch1 > ch2, иначе обмен

{

swap(ch2, ch1);

}

num11 = ch1;

num22 = ch2;

}

string plus(int first, int second)

{

int temp, sum;

ans = 1;

temp = 0;

while (first || temp)

{

ans \*= 10;

sum = first % 10 + second % 10 + temp;

if (sum >= 9)

{

ans += (sum) % 9;

temp = (sum) / 9;

}

else

{

ans += sum;

if (temp != 0)

temp = 0;

}

first /= 10;

second /= 10;

}

return to\_string(reverse(ans));

}

string minus(int first, int second)

{

ans = 1;

bool nonary = false; //занят ли "десяток" в следующей цифре

bool temp = false; //занято ли "десяток" в текущей цифре

while (first)

{

if (nonary)

temp = true;

if ((first % 10) - (second % 10) < 0)

{

nonary = true;

ans = ans \* 10 + ((nonary \* 9 - temp) + (first % 10) - (second % 10));

}

else

{

nonary = false;

ans = ans \* 10 + (first % 10 - temp) - (second % 10);

}

temp = false;

first /= 10;

second /= 10;

}

return to\_string(reverse(ans));

}

private:

int ch1, ch2, ans;

int reverse(int num)

{

int num\_new = 0;

while (num)

{

num\_new = num\_new \* 10 + num % 10;

num /= 10;

}

return num\_new/10;

}

int to\_ninefold(int num)

{

int num\_new = 1;

while (num > 8)

{

num\_new = num\_new \* 10 + (num % 9);

num /= 9;

}

num = num\_new \* 10 + num;

num = reverse(num);

return num;

}//конвертирование в 9ю СС

int to\_tenfold(int num)

{

int num\_new, i;

num\_new = i = 0;

while (num > 0)

{

num\_new = num\_new + (num % 10) \* pow(9, i);

num /= 10;

i++;

}

return num\_new;

}//конвертирование в 10ю СС

bool intSigned(const string& s)

{

for (int i = 0; i < s.length(); i++) {

if ((s[i] >= 'a' && s[i] <= 'z' )|| s[i] == '9')

{

cout << "Не верный формат исходных данных" << endl;

return false;

}

}

return true;

}//проверка строки на наличие отличных от цифр символов

};

int main()

{

s9 check;

int num1, num2;

setlocale(LC\_ALL, "rus");

cout << "Оборожный Данил Сергеевич\nИС-29\nНапишите класс s9, выполняющий сложение и вычитание двух чисел в системе счисления 9. Из большего числа должно вычитаться меньшее. Заранее не известно, какое число больше." << endl;

check.set\_numbers(num1, num2);

cout << "Plus: ";

cout << check.plus(num1, num2) << endl;

cout << "Minus: ";

cout << check.minus(num1, num2) << endl;

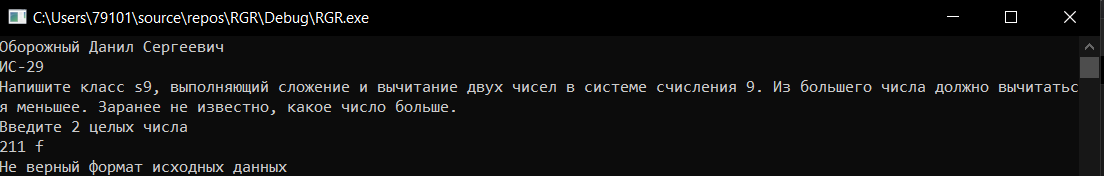
system("pause");

return 0;

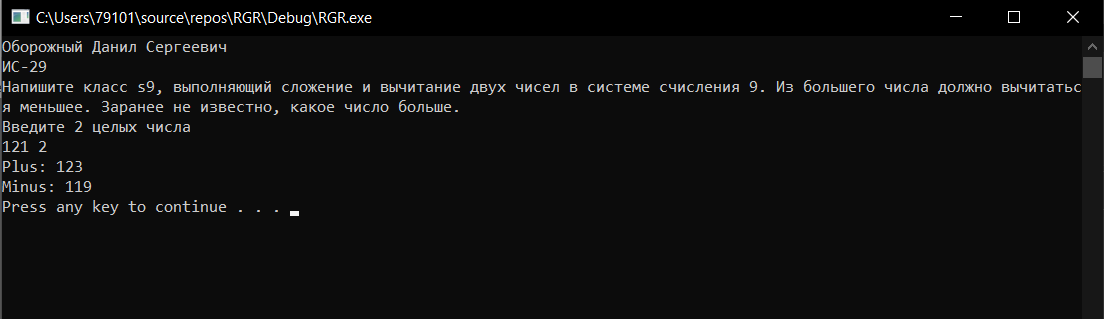
}

# Пример работы приложения

Работа приложения при некорректном вводе данных



Работа приложения при корректном вводе данных



# Список использованной литературы

1. [Электронный ресурс] https://programforyou.ru/calculators/number-systems
2. Бьярне Страуструп. Программирование Принципы и практика с использованием С++, ООО “И.Д.Вильямс”, 2011. 1248с