

Здесь будет титульник, листай ниже

# СОДЕРЖАНИЕ

1 ПОСТАНОВКА ЗАДАЧИ.....	5
1.1 Описание входных данных.....	5
1.2 Описание выходных данных.....	6
2 МЕТОД РЕШЕНИЯ.....	7
3 ОПИСАНИЕ АЛГОРИТМОВ.....	8
3.1 Алгоритм функции main.....	8
3.2 Алгоритм метода first_op класса Calc.....	9
3.3 Алгоритм метода second_op класса Calc.....	9
3.4 Алгоритм метода ret класса Calc.....	10
4 БЛОК-СХЕМЫ АЛГОРИТМОВ.....	11
5 КОД ПРОГРАММЫ.....	14
5.1 Файл Calc.cpp.....	14
5.2 Файл Calc.hpp.....	15
5.3 Файл main.cpp.....	15
6 ТЕСТИРОВАНИЕ.....	17
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	18

# 1 ПОСТАНОВКА ЗАДАЧИ

Создать объект, который вычисляет значение целочисленного арифметического выражения.

Операция деления заменена на операцию вычисления целочисленного остатка.

Объект обладает следующей функциональностью:

- выполняет первую операцию выражения, в качестве параметров передается первый целочисленный параметр, символ операции (+,-,\*,%), второй целочисленный параметр;
- вычисляет вторую и далее операцию, в качестве параметров передается символ операции (+,-,\*,%), второй целочисленный параметр;
- возвращает значение вычисленного выражения.

Написать программу, которая обязательно вводит значения и выполняет первую операцию.

Далее, в цикле осуществляет ввод очередной операции и значения второго аргумента.

Если на месте операции введен символ «С», то программа завершает работу, иначе выполняет очередную операцию и выводит результат каждой третьей операции.

## 1.1 Описание входных данных

**Первая строка:**

«целое число в десятичном формате»\_«символ операции»\_«целое число в десятичном формате»

**Последующие строки:**

«символ операции»\_«целое число в десятичном формате»

**В последней строке:**

С

## **1.2 Описание выходных данных**

**Первая строка**, с первой позиции:

«значение выражения»

**Последующие строки**, с первой позиции:

«значение выражения»

## 2 МЕТОД РЕШЕНИЯ

Для решения задачи используется:

- объект класса Calc предназначен для ;
- функция main для основная функция;
- Стандартная библиотека объектов ввода и вывода iostream объекты cin и cout;
- оператор присвоения =;
- операторы сложения, вычитания, умножения и взятия остатка от деления;
- оператор сравнения ==;
- оператор выбора switch;
- цикл с предусловием while;
- заголовочный файл Calc.hpp с описанием класса Calc.

Класс Calc:

- функционал:
  - метод first\_op — выполнение первой операции с двумя значениями;
  - метод second\_op — выполнение второй операции с одним новым значением.

## 3 ОПИСАНИЕ АЛГОРИТМОВ

Согласно этапам разработки, после определения необходимого инструментария в разделе «Метод», составляются подробные описания алгоритмов для методов классов и функций.

### 3.1 Алгоритм функции main

Функционал: точка входа в программу, считывание операций, вызов методов, вывод значений.

Параметры: нет.

Возвращаемое значение: целое число.

Алгоритм функции представлен в таблице 1.

Таблица 1 – Алгоритм функции main

№	Предикат	Действия	№ перехода
1		объявление целых a, b, cbvdjkmysq op инициализация целой cnt = 1 Создание объекта calculator класса Calc	2
2		Ввод значений переменных a, op, b	3
3		вызов метода first_op с параметрами a, op, b объекта calculator	4
4		Ввод значения переменной op	5
5	op != C	cnt++	6
		return 0	∅
6		ввод значения a	7
7	cnt == 3	вызов метода second_op с параметрами op, a	10
			8
8	cnt кратно 3	вызов метода second_op с параметрами op, a	11
			9

№	Предикат	Действия	№ перехода
9		вызов метода second_or с параметрами op, a	12
10		вызов возвращаемого метода ret у calculator значения	12
11		Вывод с новой строки возвращаемого методом ret у calculator значения	12
12		ввод op	5

### 3.2 Алгоритм метода first\_or класса Calc

Функционал: выполнение первой операции с двумя значениями.

Параметры: нет.

Возвращаемое значение: целое число.

Алгоритм метода представлен в таблице 2.

Таблица 2 – Алгоритм метода first\_or класса Calc

№	Предикат	Действия	№ перехода
1	значение op равно "+"	присвоение sum суммы a и b	Ø
	значение op равно "-"	присвоение sum разности a и b	Ø
	значение op равно "*"	присвоение sum произведения a и b	Ø
	значение op равно "%"	присвоение sum остаток от деления a на b	Ø

### 3.3 Алгоритм метода second\_or класса Calc

Функционал: выполнение второй операции с одним новым значением.

Параметры: нет.

Возвращаемое значение: целое число.

Алгоритм метода представлен в таблице 3.

Таблица 3 – Алгоритм метода *second\_or* класса *Calc*

№	Предикат	Действия	№ перехода
1		вызов метода <i>first_or</i> с параметрами <i>sum</i> , <i>ор</i> , <i>с</i>	Ø

### 3.4 Алгоритм метода *ret* класса *Calc*

Функционал: возврат значения поля *sum*.

Параметры: нет.

Возвращаемое значение: целое.

Алгоритм метода представлен в таблице 4.

Таблица 4 – Алгоритм метода *ret* класса *Calc*

№	Предикат	Действия	№ перехода
1		возврат значения поля <i>sum</i>	Ø



## 4 БЛОК-СХЕМЫ АЛГОРИТМОВ

Представим описание алгоритмов в графическом виде на рисунках 1-3.

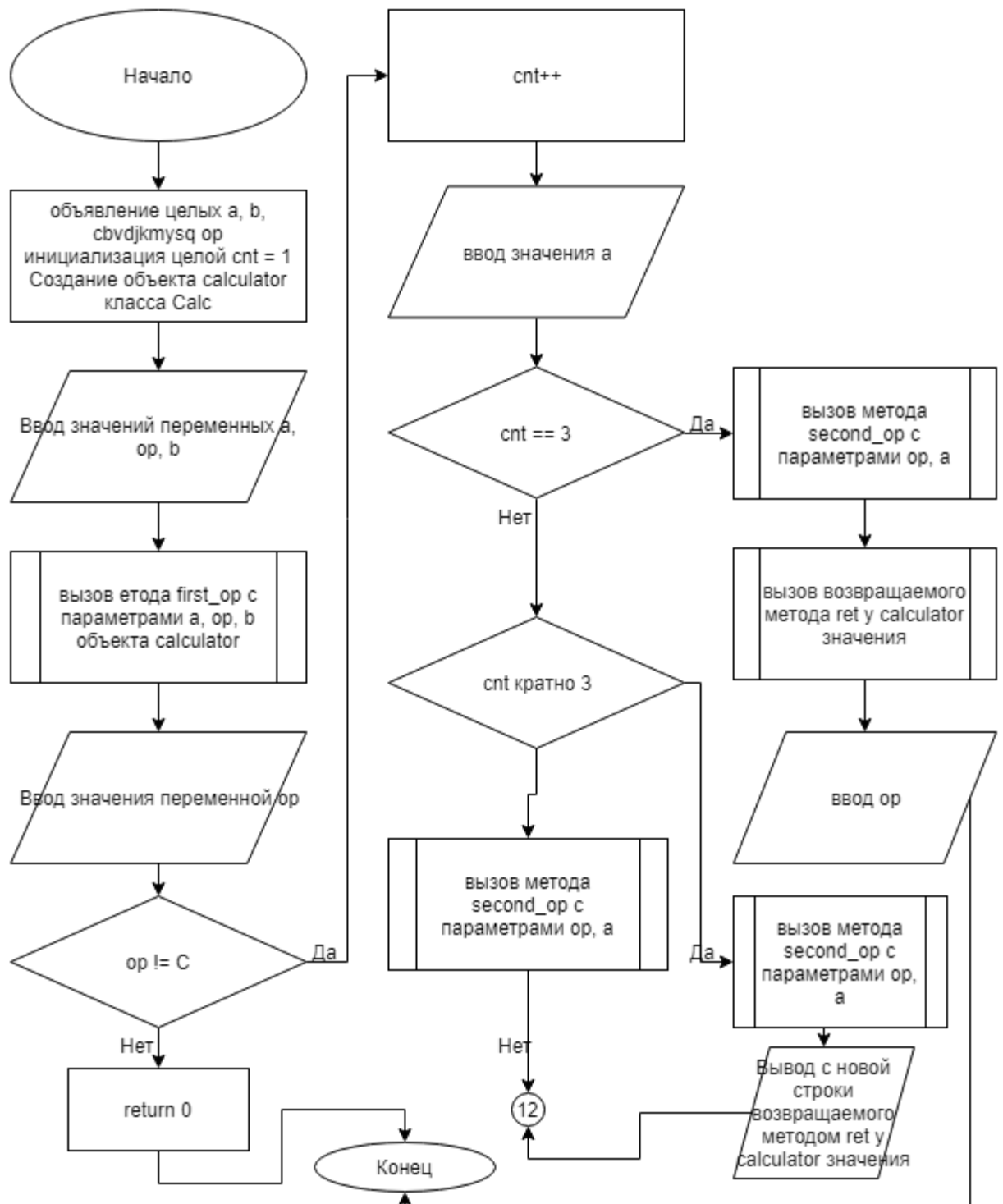
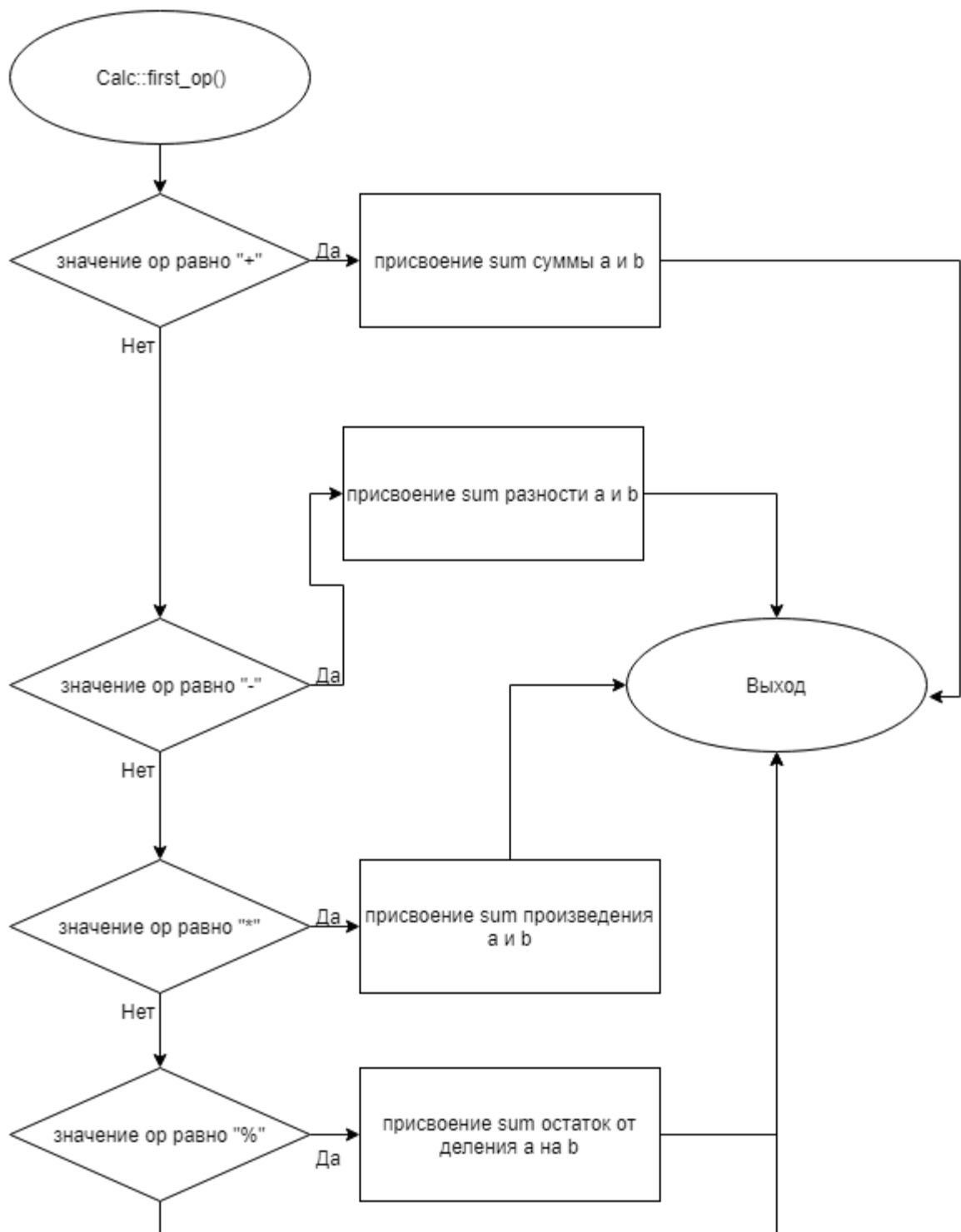
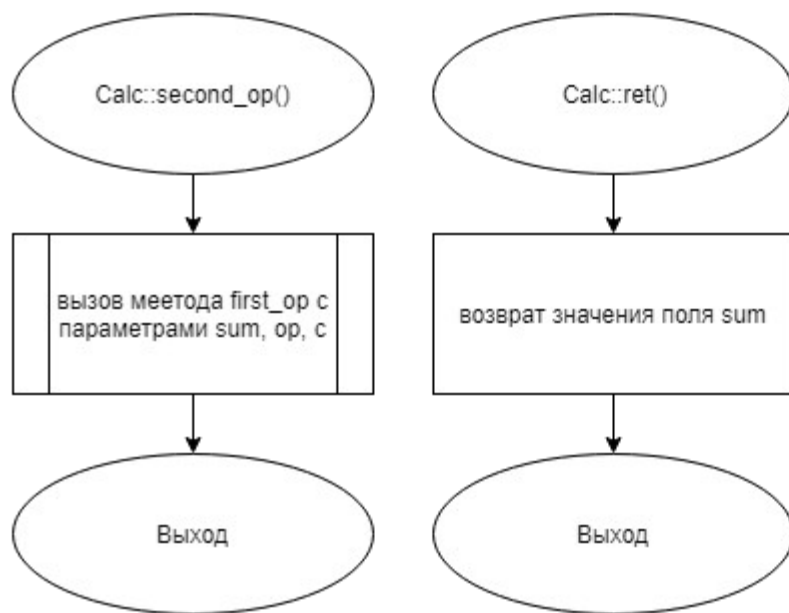


Рисунок 1 – Блок-схема алгоритма



**Рисунок 2 – Блок-схема алгоритма**



**Рисунок 3 – Блок-схема алгоритма**

## 5 КОД ПРОГРАММЫ

Программная реализация алгоритмов для решения задачи представлена ниже.

### 5.1 Файл Calc.cpp

*Листинг 1 – Calc.cpp*

```
#include "Calc.hpp"

void Calc::first_op(int a, char op, int b)
{
    switch (op)
    {
        case '+':
            sum = a + b;
            break;
        case '-':
            sum = a - b;
            break;
        case '*':
            sum = a * b;
            break;
        case '%':
            sum = a % b;
            break;
        default:
            break;
    }
}

void Calc::second_op(char op, int c){
    first_op(sum, op, c);
}

int Calc::ret(){
    return sum;
}
```

## 5.2 Файл Calc.hpp

Листинг 2 – Calc.hpp

```
#ifndef __CALC__H
#define __CALC__H

class Calc{
    int sum = 0;
public:
    void first_op(int a, char op, int b);
    void second_op(char op, int c);
    int ret();
};

#endif
```

## 5.3 Файл main.cpp

Листинг 3 – main.cpp

```
#include <stdlib.h>
#include <stdio.h>
#include <iostream>
#include "Calc.hpp"

using namespace std;

int main()
{
    int a, b;
    char op;
    int cnt = 1;
    Calc calculator;
    cin >> a >> op >> b;
    calculator.first_op(a, op, b);
    cin >> op;
    while (op != 'C')
    {
        cnt++;
        cin >> a;
        if (cnt == 3)
        {
            calculator.second_op(op, a);
            cout << calculator.ret();
        }
        else if (cnt % 3 == 0) {
            calculator.second_op(op, a);
        }
    }
}
```

```
        cout << "\n" << calculator.ret();
    }
    else
        calculator.second_op(op, a);
    cin >> op;
}
return(0);
}
```

## 6 ТЕСТИРОВАНИЕ

Результат тестирования программы представлен в таблице 5.

*Таблица 5 – Результат тестирования программы*

<b>Входные данные</b>	<b>Ожидаемые выходные данные</b>	<b>Фактические выходные данные</b>
4 + 2 + 5 + 5 С	16	16

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 19 Единая система программной документации.
2. Методическое пособие студента для выполнения практических заданий, контрольных и курсовых работ по дисциплине «Объектно-ориентированное программирование» [Электронный ресурс] – URL: [https://mirea.aco-avvora.ru/student/files/methodichescoe\\_posobie\\_dlya\\_laboratornyh\\_rabot\\_3.pdf](https://mirea.aco-avvora.ru/student/files/methodichescoe_posobie_dlya_laboratornyh_rabot_3.pdf) (дата обращения 05.05.2021).
3. Приложение к методическому пособию студента по выполнению заданий в рамках курса «Объектно-ориентированное программирование» [Электронный ресурс]. URL: [https://mirea.aco-avvora.ru/student/files/Prilozheniye\\_k\\_methodichke.pdf](https://mirea.aco-avvora.ru/student/files/Prilozheniye_k_methodichke.pdf) (дата обращения 05.05.2021).
4. Шилдт Г. С++: базовый курс. 3-е изд. Пер. с англ.. — М.: Вильямс, 2019. — 624 с.
5. Видео лекции по курсу «Объектно-ориентированное программирование» [Электронный ресурс]. АСО «Аврора».
6. Антик М.И. Дискретная математика [Электронный ресурс]: Учебное пособие /Антик М.И., Казанцева Л.В. — М.: МИРЭА — Российский технологический университет, 2018 — 1 электрон. опт. диск (CD-ROM).