

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
3.0 Алгоритм функции main.....	8
3.1 Алгоритм метода f1 класса cl.....	8
3.2 Алгоритм метода f2 класса cl.....	9
3.3 Алгоритм метода result класса cl.....	10
4 БЛОК-СХЕМЫ АЛГОРИТМОВ.....	11
5 КОД ПРОГРАММЫ.....	14
5.0 Файл main.cpp.....	14
6 ТЕСТИРОВАНИЕ.....	16
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	17

1 ПОСТАНОВКА ЗАДАЧИ

Создать объект, который вычисляет значение целочисленного арифметического выражения, состоящего из трех последовательных операций. 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 (значение можно получить после выполнения трех операций).

Написать программу, которая:

1. Создает объект.
2. Вводит значения аргументов для первой операции.
3. Выполняет первую операцию.
4. Вводит значение аргументов для второй операции.
5. Выполняет вторую операцию.
6. Вводит значение аргументов для третьей операции.
7. Выполняет третью операцию.
8. Выводит результат.

1.1 Описание входных данных

Первая строка:

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

десятичном формате»

Вторая строка:

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

Третья строка:

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

1.2 Описание выходных данных

Первая строка, с первой позиции:

«значение выражения»

2 МЕТОД РЕШЕНИЯ

Для решения задачи понадобится:

объект класса `cl`

условный оператор `if`

операторы `(+,-,*,%)`

Класс `cl`:

Поля:

доступные элементы с типом `int a, int b, char symbol`

скрытый элемент с типом `int answer`

Методы:

открытые:

`f1(int a, char symbol, int b)` - выполняет операцию (действие `+`, `-`, `*`, `%`).

`f2(char symbol, int b)` - выполняет операцию (действие `+`, `-`, `*`, `%`).

`result()` - возвращает итоговое значение

3 ОПИСАНИЕ АЛГОРИТМОВ

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

3.0 Алгоритм функции main

Функционал: главный метод программы.

Параметры: нет.

Возвращаемое значение: int.

Алгоритм функции представлен в таблице 1.

Таблица 1 – Алгоритм функции main

№	Предикат	Действия	№ перехода
1		ввод значения аргументов для первой операции	2
2		создание объекта	3
3		вызов метода f1, для выполнения операции	4
4		ввод значение аргументов для второй операции	5
5		вызов метода f2, для выполнения операции	6
6		ввод значение аргументов для третьей операции	7
7		вызов метода f2, для выполнения операции	8
8		вывод результата метода result	Ø

3.1 Алгоритм метода f1 класса cl

Функционал: вычисление первой операции над числами.

Параметры: int a, char symbol, int b.

Возвращаемое значение: нет.

Алгоритм метода представлен в таблице 2.

Таблица 2 – Алгоритм метода $f1$ класса cl

№	Предикат	Действия	№ перехода
1	знак операции равен "+"	сложение $a + b$ и присваиваем значение в <code>answer</code>	∅
			2
2	знак операции равен "-"	вычитание $a - b$ и присваиваем значение в <code>answer</code>	∅
			3
3	знак операции равен "*"	умножение $a * b$ и присваиваем значение в <code>answer</code>	∅
			4
4	знак операции равен "%"	деление на остаток $a \% b$ и присваиваем значение в <code>answer</code>	∅
			∅

3.2 Алгоритм метода $f2$ класса cl

Функционал: вычисление второй и далее операций над числами.

Параметры: `char symbol`, `int b`.

Возвращаемое значение: нет.

Алгоритм метода представлен в таблице 3.

Таблица 3 – Алгоритм метода $f2$ класса cl

№	Предикат	Действия	№ перехода
1	знак операции равен "+"	прибавление к <code>answer</code> значение <code>b</code>	∅
			2
2	знак операции равен "-"	вычитание к <code>answer</code> значение <code>b</code>	∅
			3
3	знак операции равен "*"	умножение к <code>answer</code> значение <code>b</code>	∅
			4
4	знак операции равен "%"	деление на остаток к <code>answer</code> значение <code>b</code>	∅

№	Предикат	Действия	№ перехода
			∅

3.3 Алгоритм метода result класса cl

Функционал: возврат значения.

Параметры: нет.

Возвращаемое значение: нет.

Алгоритм метода представлен в таблице 4.

Таблица 4 – Алгоритм метода result класса cl

№	Предикат	Действия	№ перехода
1		возврат значения answer	∅

4 БЛОК-СХЕМЫ АЛГОРИТМОВ

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

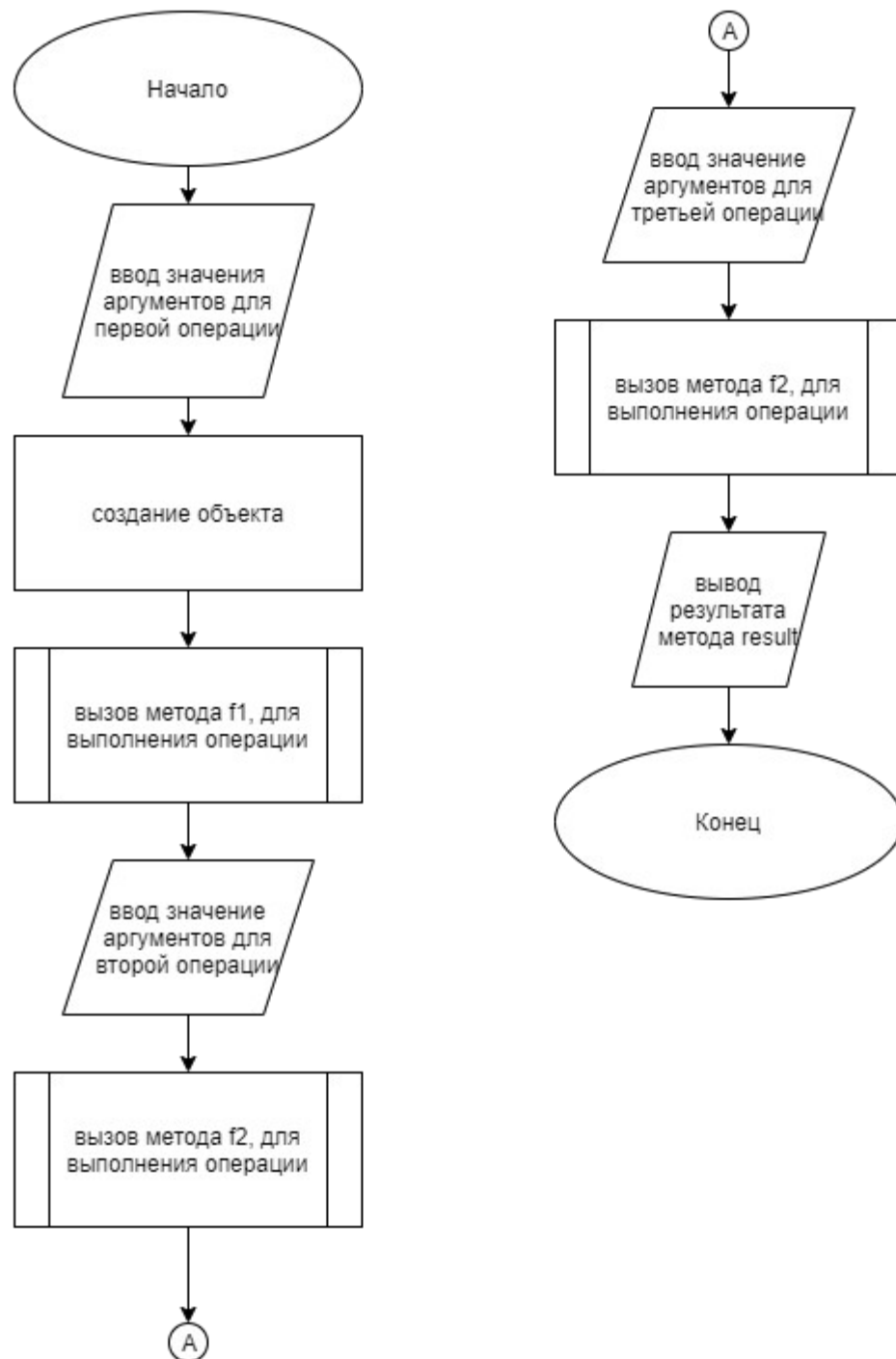


Рисунок 1 – Блок-схема алгоритма

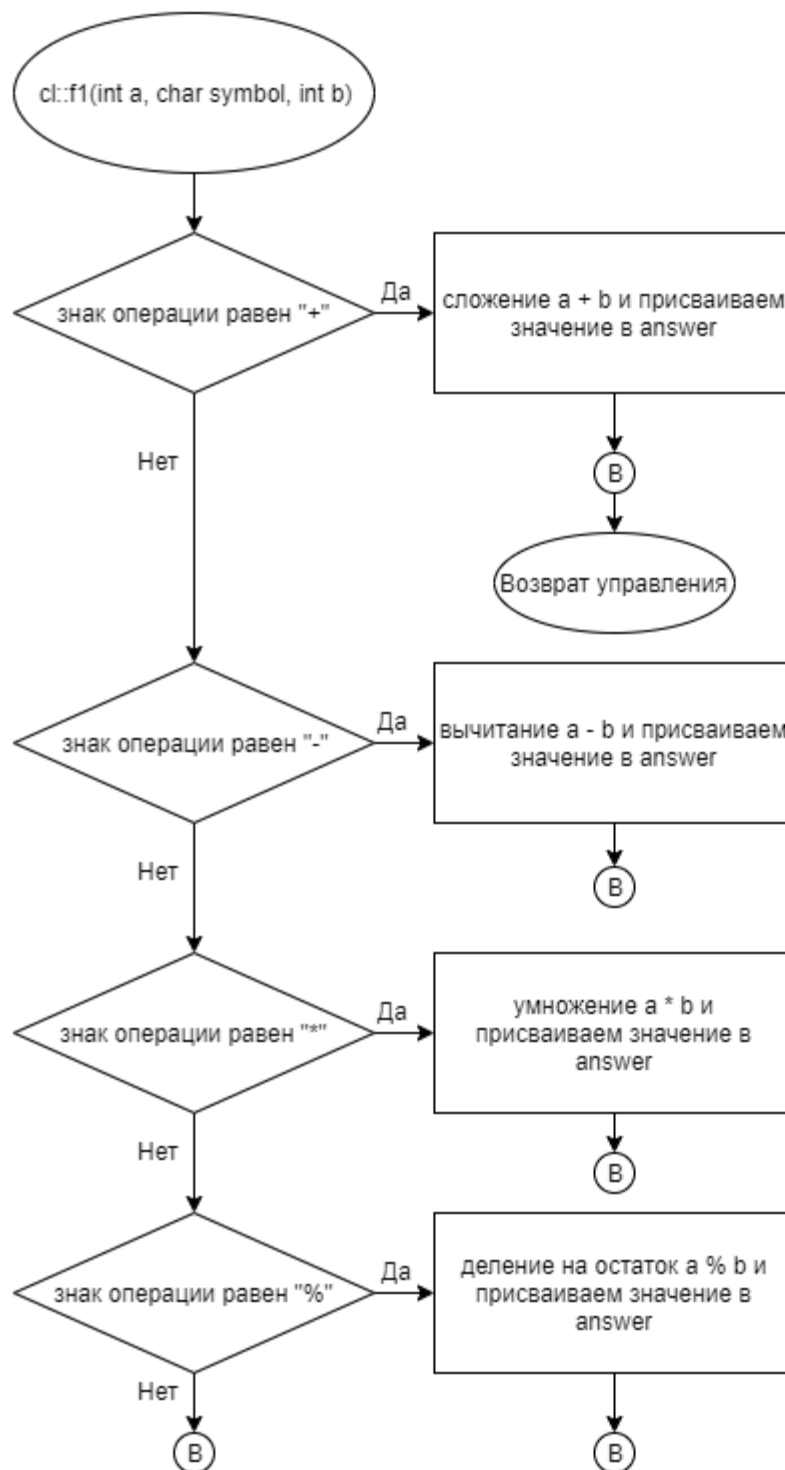


Рисунок 2 – Блок-схема алгоритма

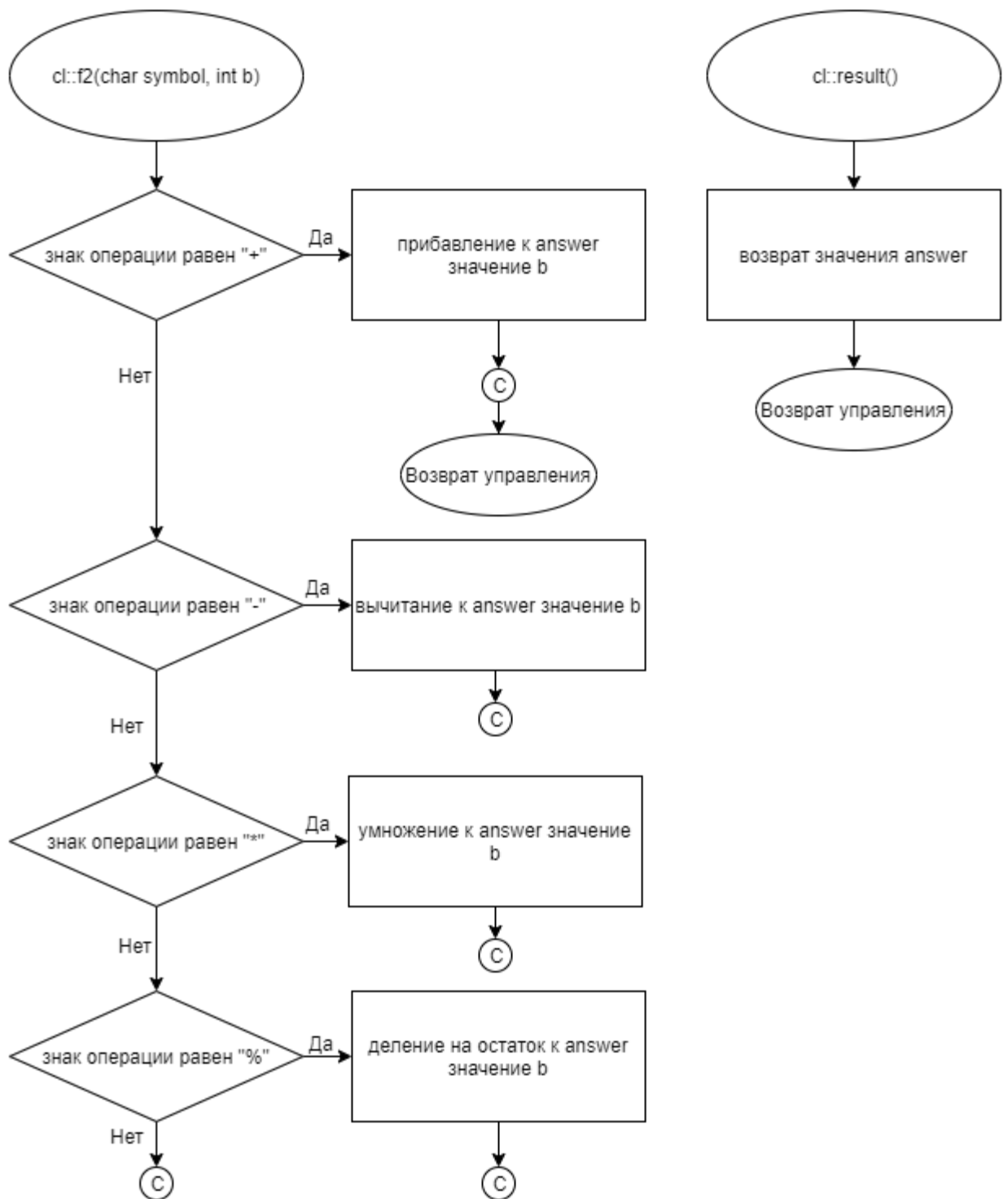


Рисунок 3 – Блок-схема алгоритма

5 КОД ПРОГРАММЫ

Программная реализация алгоритмов для решения задачи представлена ниже.

5.0 Файл main.cpp

Листинг 1 – main.cpp

```
#include <stdlib.h>
#include <stdio.h>
#include <iostream>

using namespace std;

class cl
{
private:
    int answer = 0;
public:
    void f1(int a, char symbol, int b);
    void f2(char symbol, int b);
    int result();
    int a, b;
    char symbol;
};

void cl::f1(int a, char symbol, int b)
{
    if (symbol == '+')
    {
        answer = a + b;
    }
    else if (symbol == '-')
    {
        answer = a - b;
    }
    else if (symbol == '*')
    {
        answer = a * b;
    }
    else if (symbol == '%')
    {
        answer = a % b;
    }
}

void cl::f2(char symbol, int b)
{
    if (symbol == '+')
```

```

        {
            answer += b;
        }
        else if (symbol == '-')
        {
            answer -= b;
        }
        else if (symbol == '*')
        {
            answer *= b;
        }
        else if (symbol == '%')
        {
            answer %= b;
        }
    }

int cl::result()
{
    return answer;
}

int main()
{
    int a, b;
    char symbol;
    cin >> a >> symbol >> b;

    cl object;
    object.f1(a, symbol, b);

    cin >> symbol >> b;
    object.f2(symbol, b);

    cin >> symbol >> b;
    object.f2(symbol, b);

    cout << object.result();

    return(0);
}

```

6 ТЕСТИРОВАНИЕ

Результат тестирования программы представлен в таблице 5.

Таблица 5 – Результат тестирования программы

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
2 + 2 + 2 + 2	8	8

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Васильев А.Н. Объектно-ориентированное программирование на С++. Издательство: Наука и Техника. Санкт-Петербург, 2016г. 543 стр.
2. Шилдт Г. С++: базовый курс. 3-е изд. Пер. с англ.. — М.: Вильямс, 2017. — 624 с.
3. Методическое пособие для проведения практических заданий, контрольных и курсовых работ по дисциплине «Объектно-ориентированное программирование» [Электронный ресурс] — URL: https://mirea.aco-avvora.ru/student/files/methodicheskoe_posobie_dlya_laboratornyh_rabot_3.pdf (дата обращения 05.05.2021).
4. 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 (дата обращения 05.05.2021).
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).