



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

« МИРЭА Российский технологический университет»

РТУ МИРЭА

Институт Информационных технологий

Кафедра Вычислительной техники

УЧЕБНОЕ ЗАДАНИЕ

по дисциплине

« Объектно-ориентированное программирование»

Наименование задачи:

« Задание 3_1_1 »

С тудент группы

ИКБО-27-21

Родионов А.А.

Руководитель практики

Ассистент

Морозов В.А.

Работа представлена

«__»_____ 2022 г.

(подпись студента)

Оценка

(подпись руководителя)

Москва 2022

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
Постановка задачи.....	5
Метод решения.....	7
Описание алгоритма.....	9
Блок-схема алгоритма.....	14
Код программы.....	19
Тестирование.....	22
ЗАКЛЮЧЕНИЕ.....	23
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ (ИСТОЧНИКОВ).....	24

ВВЕДЕНИЕ

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.

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»

Метод решения

Для выполнения задачи нам потребуется :

- Объект ввода/вывода потока данных cin/cout (iostream)
- Условный оператор switch
- Цикл for
- Цикл while
- Объект m класса Math

Класс Math:

Поля:

Модификатор доступа public :

Наименование : Math

Функционал : Конструктор

Наименование : operation

Параметры char op, int s

Функционал : производит математическую операцию

Наименование : get_len

Функционал : получить количество операций

Наименование : set_val

Параметры : int val

Функционал : Устанавливает значение

Наименование : add_exp

Параметры : char op, int s

Функционал : установить выражение

Наименование : get_exp

Параметры : int i, char *op, int *s

Функционал : получить выражение

Описание алгоритма

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

Конструктор класса: Math

Модификатор доступа: public

Функционал: Конструктор класса

Параметры: нет

Алгоритм конструктора представлен в таблице 1.

Таблица 1. Алгоритм конструктора класса Math

№	Предикат	Действия	№ перехода	Комментарий
1		sec_values = new int[1000]	2	
2		operations = new char[1000]	3	
3		newt_id = 0;	Ø	

Класс объекта: Math

Модификатор доступа: public

Метод: operation

Функционал: Выполняет операцию

Параметры: char op, int s

Возвращаемое значение: int

Алгоритм метода представлен в таблице 2.

Таблица 2. Алгоритм метода operation класса Math

№	Предикат	Действия	№ перехода	Комментарий
1	op == '+'		2	
	op == '-'		3	
	op == '*'		4	
	op == '%'		5	
			Ø	
2		return val + s	Ø	
3		return val - s	Ø	
4		return val * s	Ø	
5		return val % s	Ø	

Класс объекта: Math

Модификатор доступа: public

Метод: get_len

Функционал: Возвращает количество выражений

Параметры: нет

Возвращаемое значение: int

Алгоритм метода представлен в таблице 3.

Таблица 3. Алгоритм метода get_len класса Math

№	Предикат	Действия	№ перехода	Комментарий
1		return next_id + 1	Ø	

Класс объекта: Math

Модификатор доступа: public

Метод: set_val

Функционал: Установить значение

Параметры: нет

Возвращаемое значение: void

Алгоритм метода представлен в таблице 4.

Таблица 4. Алгоритм метода set_val класса Math

№	Предикат	Действия	№ перехода	Комментарий
1		val = v	Ø	

Класс объекта: Math

Модификатор доступа: public

Метод: add_exp

Функционал: Добавить выражение

Параметры: char op, int s

Возвращаемое значение: void

Алгоритм метода представлен в таблице 5.

Таблица 5. Алгоритм метода add_exp класса Math

№	Предикат	Действия	№ перехода	Комментарий
1		sec_values[next_id] = s	2	
2		operation[next_id] = op	3	
3		next_id++	Ø	

Класс объекта: Math

Модификатор доступа: public

Метод: get_exp

Функционал: получить выражение

Параметры: int i, char *op, int *s

Возвращаемое значение: void

Алгоритм метода представлен в таблице 6.

Таблица 6. Алгоритм метода get_exp класса Math

№	Предикат	Действия	№ перехода	Комментарий
1		*op = operations[i]	2	
2		*s = sec_values[i]	Ø	

Функция: main

Функционал: главная функция программы

Параметры: нет

Возвращаемое значение: void

Алгоритм функции представлен в таблице 7.

Таблица 7. Алгоритм функции main

№	Предикат	Действия	№ перехода	Комментарий
1		Инициализируем объект типа Math Math m	2	
2		Инициализируем переменные типа int int f,s	3	
3		Инициализируем	4	

		переменную типа op		
4		Ввод значения переменных f, op, s	5	
5		m.set_val(f)	6	
6		m.add_exp(op,s)	7	
7		while(true)	8	
8		Ввод значения op	9	
9	op == 'c'		10	
			11	
10		break;	Ø	
11		Ввод значения s	12	
12		m.add_exp(op, s)	13	
13		for(int i = 0; i < m.get_len() - 1;i++)	14	
14		m.get_exp(i, &op, &s)	15	
15		Инициализируем целочисленную переменную типа integer int num = m.operatoin(op,s)	16	
16		m.set_val(num)	17	
17	(i+1) % 3 == 0		18	
			Ø	
18		Вывод значение num	19	
19	(i+1 < m.get_len() - 3)		20	
			Ø	
20		Перенос строки	Ø	

Блок-схема алгоритма

Представим описание алгоритмов в графическом виде на рисунках ниже.

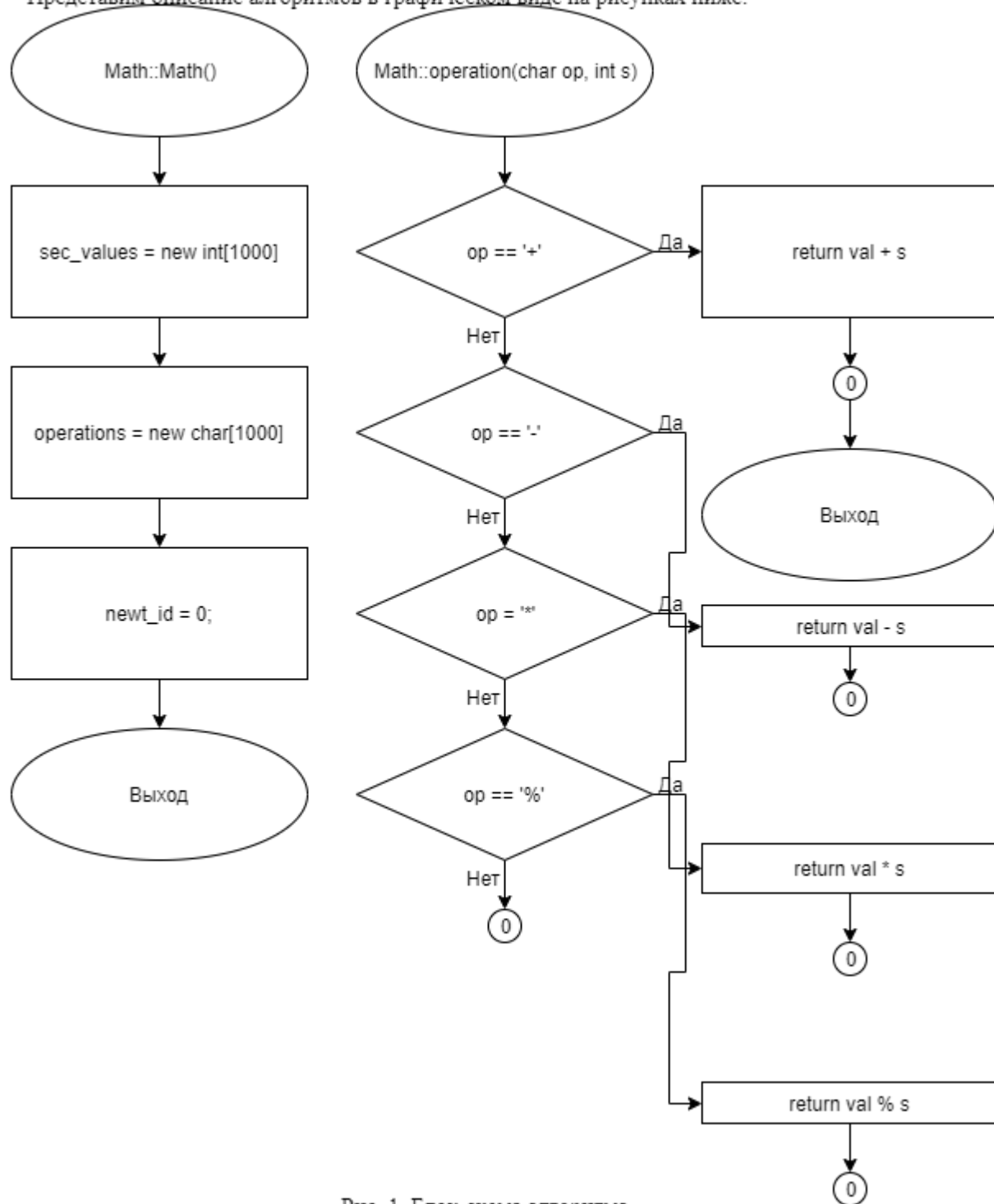


Рис. 1. Блок-схема алгоритма.

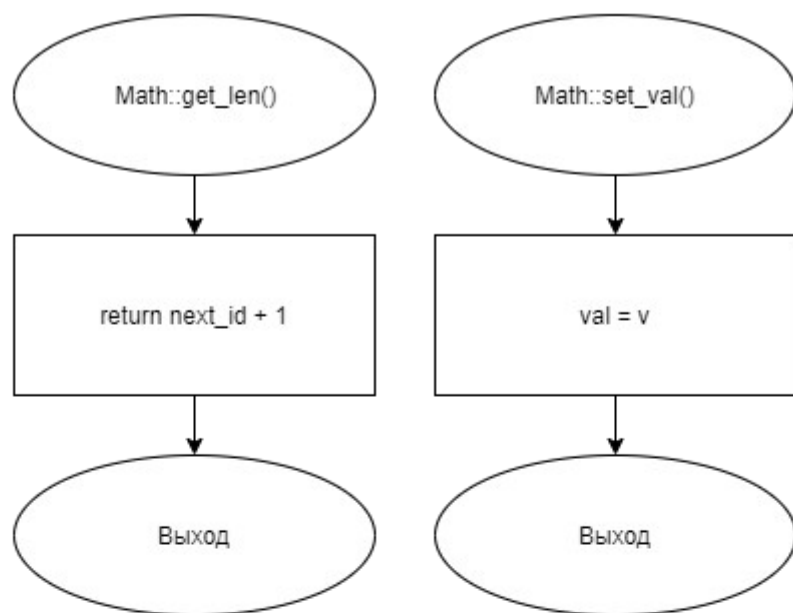


Рис. 2. Блок-схема алгоритма.

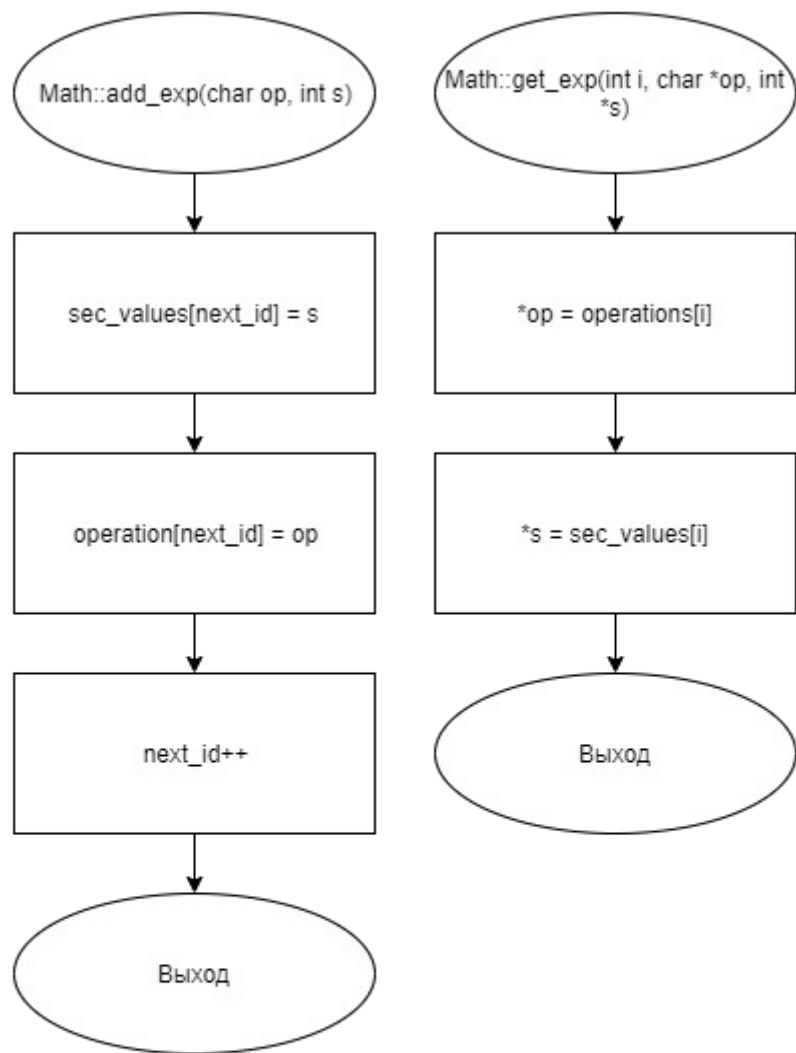


Рис. 3. Блок-схема алгоритма.

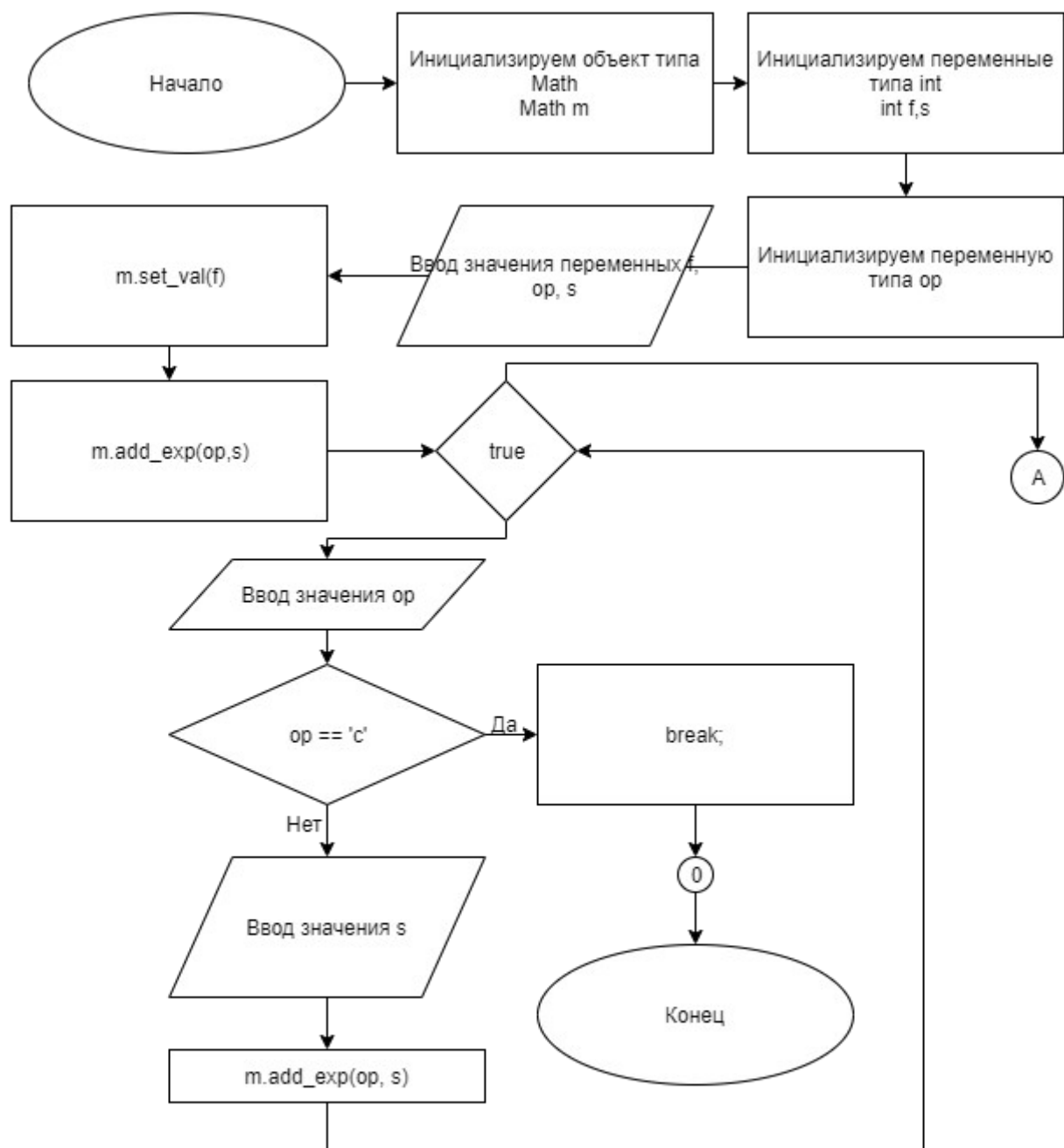


Рис. 4. Блок-схема алгоритма.

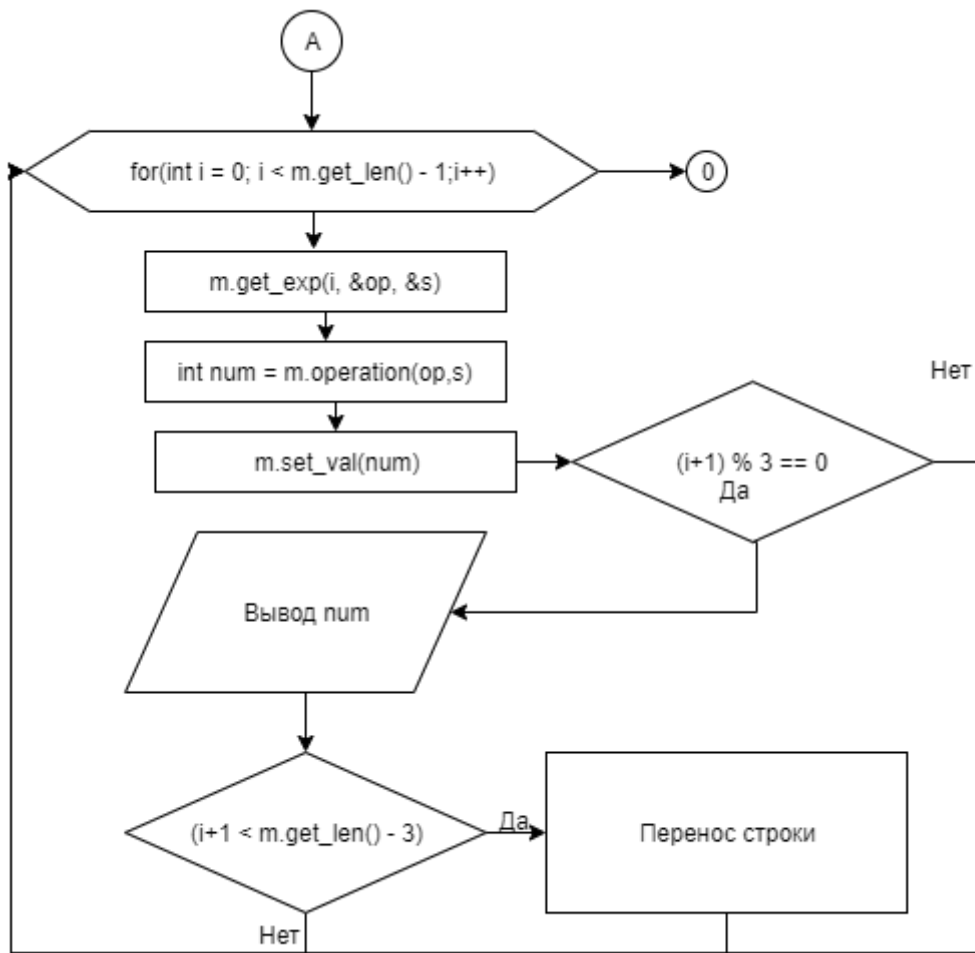


Рис. 5. Блок-схема алгоритма.

Код программы

Программная реализация алгоритмов для решения задачи представлена ниже.

Файл main.cpp

```
#include <iostream>
#include "Math.h"

using namespace std;
int main()
{
    Math m;
    int f,s;
    char op;
    cin >> f >> op >> s;
    m.set_val(f);
    m.add_exp(op,s);
    while(true){
        cin >> op;
        if(op == 'C')break;
        cin >> s;
        m.add_exp(op,s);
    }
    for(int i = 0; i < m.get_len() - 1;i++){
        m.get_exp(i, &op, &s);
        int num = m.operation(op,s);
        m.set_val(num);
        if((i+1) % 3 == 0){
            cout << num;
            if((i+1 < m.get_len() - 3))cout << endl;
        }
    }
}
```

Файл Math.cpp

```
#include "Math.h"
#include <iostream>

using namespace std;

Math::Math(){
    sec_values = new int[1000];
    operations = new char[1000];
    next_id = 0;
}

int Math::operation(char op, int s){
```

```

        switch(op){
            case '+':
                return val + s;
                break;
            case '-':
                return val - s;
                break;
            case '*':
                return val * s;
                break;
            case '%':
                return val % s;
                break;
        }
    }

    int Math::get_len(){
        return next_id + 1;
    }

    void Math::set_val(int v){
        val = v;
    }

    void Math::add_exp(char op, int s){
        sec_values[next_id] = s;
        operations[next_id] = op;
        next_id++;
    }

    void Math::get_exp(int i, char *op, int *s){
        *op = operations[i];
        *s = sec_values[i];
    }
}

```

Файл Math.h

```

#ifndef Math_h
#define Math_h

class Math{
private:
    int val;
    int next_id;
    int *sec_values;
    char *operations;
public:
    Math();
    int operation(char op, int s);
    int get_len();
    void set_val(int v);
}

```

```
void add_exp(char op, int s);  
void get_exp(int i, char *op, int *s);  
};  
#endif
```

Тестирование

Результат тестирования программы представлен в следующей таблице.

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
$5 \% 2 + 2 - 2 * 2 \% 2 \text{ C}$	1	1
$8 + 1 + 2 - 2 * 2 \% 2 \text{ C}$	9	9
$0 + 1 + 0 - 0 * 0 \% 1 \text{ C}$	1	1

ЗАКЛЮЧЕНИЕ

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ (ИСТОЧНИКОВ)

1. Васильев А.Н. Объектно-ориентированное программирование на C++. Издательство: Наука и Техника. Санкт-Петербург, 2016г. 543 стр.
2. Шилдт Г. C++: 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-avrrora.ru/student/files/methodichescoe_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-avrrora.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).