

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий
Кафедра вычислительной техники

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ № 3

Разветвляющаяся программа

Вариант № 19

Преподаватель

подпись, дата

Пушкарев К. В.

Студент КИ18-096, 031831293

подпись

01.10.2018

Овсянников В.А.

Красноярск 2018

1 Задание

1. Разработать графическую схему алгоритма и программу, которая для двух целых чисел, введенных с клавиатуры, вычисляет остаток от целочисленного деления, частное от целочисленного деления первого числа на второе, а также частное от вещественного деления.

2. Провести трассировку программы с помощью встроенного в среду программирования отладчика, анализируя значения переменных после каждого оператора присваивания.

3. Выполнить несколько запусков программы для заранее подготовленных тестовых наборов данных. Сделать вывод результатов с применением потокового ввода-вывода, используя следующие методы потоков – `width()`, `precision()` и `fill()` с различными параметрами (не менее 3 для каждого метода), а также флаги `left`, `right`, устанавливаемые с помощью метода `setf()`. Проанализировать полученные результаты.

4. Выполнить несколько запусков программы для заранее подготовленных тестовых примеров для проверки работоспособности всех ветвей программы. Для тестовых примеров рекомендуется взять как произвольные значения исходных данных, так и критические значения (например, ноль). Полученные результаты проанализировать.

2 Цели работы

1. Закрепить навыки работы в среде C++.
2. Познакомиться с потоковым вводом-выводом (`cin`, `cout`).
3. Изучить операцию условия.
4. Изучить условный оператор.
5. Изучить оператор множественного выбора.

3 Подход к решению задач

Согласно условию задачи, требуется разработать графическую схему алгоритма и программу, которая предлагает пользователю ввести два целых числа, получает числа от пользователя и затем, если введенные числа удовлетворяют условию программы, вычисляет и выводит остаток от целочисленного деления, частное от целочисленного деления первого числа на второе, а также частное от вещественного деления. Создадим переменные `num1`, `num2`, `mod`, `div_int`, типа `int` и одну переменную `div_float` типа `float`. Переменные `num1` и `num2` пользователь должен будет ввести с клавиатуры. Если введенные числа удовлетворяют условию программы, следует выполнение этих действий: вычисление остатка от целочисленного деления, частного от целочисленного деления первого числа на второе, а также частного от вещественного деления. После чего последует вывод вычисленных значений на экран. Иначе на экран будет выведена фраза "Произошла ошибка в вычислениях".

4 Графическая схема алгоритма

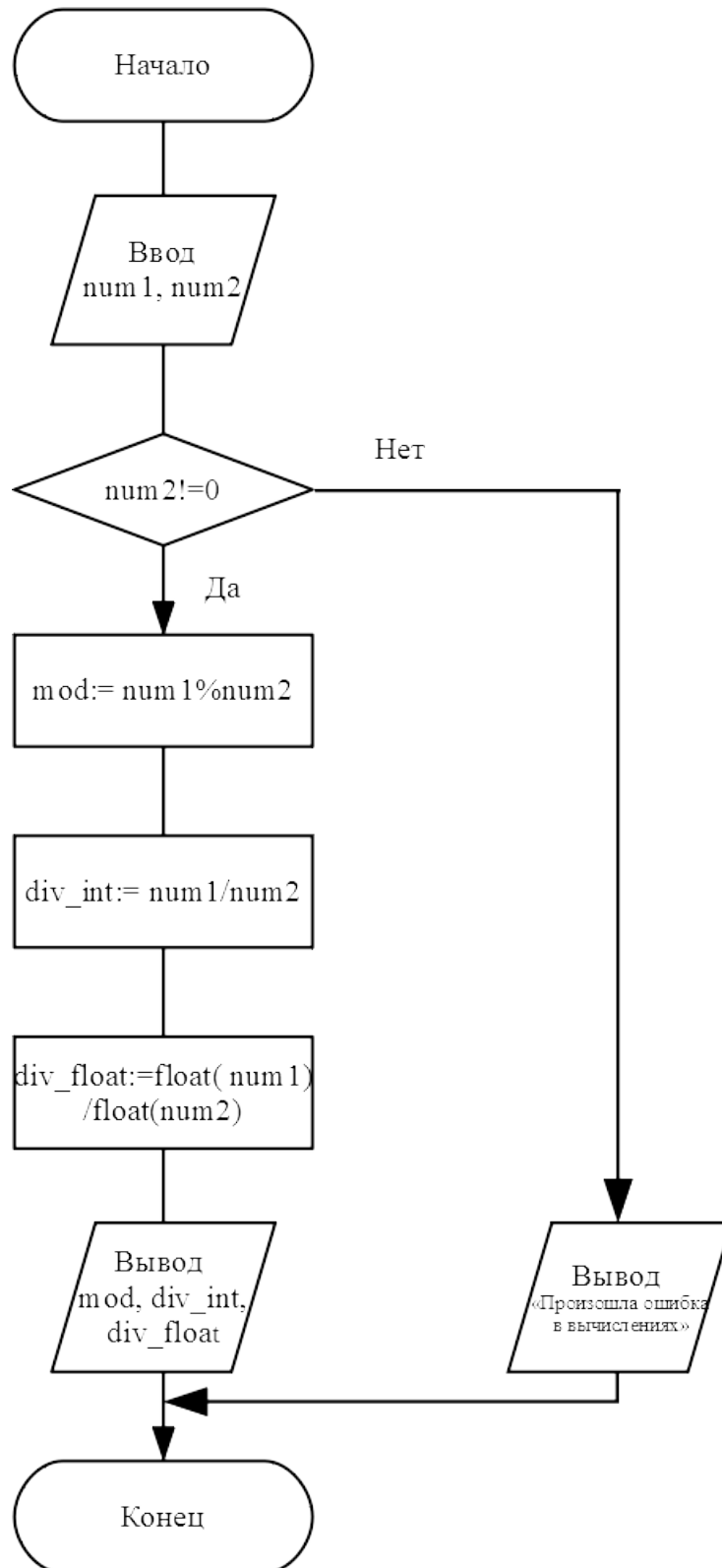


Рисунок 1 – Блок-схема основного упражнения.

5 Код программы

```
1 #include "pch.h"
2 #include <iostream>
```

```

3 #include <locale>
4
5 using namespace std;
6
7 int main(){
8     setlocale(LC_ALL, "rus");
9
10    int num1, num2, mod, div_int;
11    float div_float;
12
13    cout << "Введите первое число: ";
14    cin >> num1;
15
16    cout << "Введите второе число: ";
17    cin >> num2;
18
19    if (num2 != 0) {
20        mod = num1 % num2;
21        div_int = num1 / num2;
22        div_float = float(num1) / float(num2);
23
24        cout.setf(ios::fixed);
25        cout.setf(ios::left);
26        cout.width(25);
27        cout.precision(2);
28        cout.fill('.');
29        cout << "Остаток от деления: " << float(mod) << endl;
30
31        cout.setf(ios::fixed);
32        cout.setf(ios::left);
33        cout.width(30);
34        cout.precision(4);
35        cout.fill('^');
36        cout << "Остаток от деления: " << float(mod) << endl;
37
38        cout.setf(ios::fixed);
39        cout.setf(ios::left);
40        cout.width(35);
41        cout.precision(6);
42        cout.fill('%');
43        cout << "Остаток от деления: " << float(mod) << endl;
44
45        cout.setf(ios::fixed);
46        cout.setf(ios::left);
47        cout.width(50);
48        cout.precision(7);
49        cout.fill('-');
50        cout << "Частное от целочисленного деления : " << float(div_int) << endl;
51
52        cout.setf(ios::fixed);
53        cout.setf(ios::left);
54        cout.width(55);
55        cout.precision(9);
56        cout.fill('+');
57        cout << "Частное от целочисленного деления : " << float(div_int) << endl;
58
59        cout.setf(ios::fixed);
60        cout.setf(ios::left);
61        cout.width(60);
62        cout.precision(11);
63        cout.fill('=');
64        cout << "Частное от целочисленного деления : " << float(div_int) << endl;
65

```

```

66         cout.setf(ios::fixed);
67         cout.setf(ios::right);
68         cout.width(75);
69         cout.precision(5);
70         cout.fill('*');
71         cout << "Частное от вещественного деления : " << div_float << endl;
72
73         cout.setf(ios::fixed);
74         cout.setf(ios::right);
75         cout.width(80);
76         cout.precision(8);
77         cout.fill('#');
78         cout << "Частное от вещественного деления : " << div_float << endl;
79
80         cout.setf(ios::fixed);
81         cout.setf(ios::right);
82         cout.width(85);
83         cout.precision(12);
84         cout.fill('$');
85         cout << "Частное от вещественного деления : " << div_float << endl;
86     }
87     else {
88         cout << "Произошла ошибка в вычислениях" << endl;
89     }
90     return 0;
91 }

```

Протокол отладки программы

Трассировка программы при num1=5, num2=2, приведена в таблице 1.

Таблица 1 – Трассировка программы

Номер строки программы	Переменные				
	num1		mod	div_int	float
		2			
			1		
				2	
					2.50000000

7 Результаты выполнения экспериментальной части работы

Результаты запуска программы с различными входными значениями приведены в таблице 2.

Таблица 2 – Работа программы на тестовых наборах данных

Входные данные	Результат
num1 = 5, num2 = 2	Остаток от целочисленного деления = ****1 Остаток от целочисленного деления = #####1

Входные данные	Результат
	<p>Остаток от целочисленного деления = 1</p> <p>****Частное от целочисленного деления = 2</p> <p>#####Частное от целочисленного деления = 2</p> <p>Частное от целочисленного деления = 2</p> <p>*****Частное от вещественного деления = 2.50</p> <p>#####Частное от вещественного деления =</p> <p>Частное от вещественного деления = 2.50000</p>
num1 = 48, num2 = 15	<p>Остаток от целочисленного деления = ****3</p> <p>Остаток от целочисленного деления = #####3</p> <p>Остаток от целочисленного деления = 3</p> <p>****Частное от целочисленного деления = 3</p> <p>#####Частное от целочисленного деления = 3</p> <p>Частное от целочисленного деления = 3</p> <p>*****Частное от вещественного деления = 3.20</p> <p>#####Частное от вещественного деления =</p> <p>Частное от вещественного деления = 3.20000</p>

8 Дополнительное упражнение № 2

Даны три точки заданы своими координатами. Определить принадлежат ли эти точки одной прямой. Значения x_1 , y_1 , x_2 , y_2 , x_3 , y_3 вводятся с клавиатуры, на монитор выводится слово YES (если точки принадлежат) или NO (если точки не принадлежат прямой).

9 Подход к решению задачи дополнительного упражнения № 2

Согласно условию задачи, требуется разработать программу, которая предлагает пользователю ввести координаты трёх точек, получает числа от пользователя и затем, если введённые координаты принадлежат одной прямой выводит слово YES, иначе программы должна вывести NO. Создадим переменные x_1 , y_1 , x_2 , y_2 , x_3 , y_3 типа float. Чтобы переменные x_1 , y_1 , x_2 , y_2 , x_3 , y_3 принадлежали одной прямой все точки x или все точки y должны быть

равны, если прямая вертикальная или горизонтальная, а в случае когда прямая проходит по диагонали точки должны соответствовать следующему условию: $(x3 - x1) / (x2 - x1) == (y3 - y1) / (y2 - y1)$, в результате чего последует вывод слова YES, иначе вывод слова NO.

10 Код программы дополнительного упражнения № 2

```
1 #include "pch.h"
2 #include <iostream>
3 #include <locale>
4
5 using namespace std;
6
7 int main(){
8     setlocale(LC_ALL, "rus");
9
10    float x1, x2, x3, y1, y2, y3;
11
12    cout << "Введите x1:";
13    cin >> x1;
14
15    cout << "Введите y1:";
16    cin >> y1;
17
18    cout << "Введите x2:";
19    cin >> x2;
20
21    cout << "Введите y2:";
22    cin >> y2;
23
24    cout << "Введите x3:";
25    cin >> x3;
26
27    cout << "Введите y3:";
28    cin >> y3;
29
30    if (x1 == x2 && x1 == x3) {
31        cout << "\nYES" << endl;
32    }
33    else if (y1 == y2 && y1 == y3) {
34        cout << "\nYES" << endl;
35    }
36    else if ((x3 - x1) / (x2 - x1) == (y3 - y1) / (y2 - y1)) {
37        cout << "\nYES" << endl;
38    }
39    else {
40        cout << "\nNO" << endl;
41    }
42    return 0;
43 }
```

11 Протокол отладки дополнительного упражнения № 2

Трассировка программы при $x_1 = 3$, $y_1 = 2$, $x_2 = 8$, $y_2 = 3$, $x_3 = 13$, $y_3 = 4$, приведена в таблице 3.

Таблица 3 – Трассировка программы

Номер строки программы	Переменные					
	x_1	1	x_2	y_2	x_3	3
		2				
			8			
				3		
6					13	
29						4

12 Результат выполнения экспериментальной части работы дополнительного упражнения №2

Результаты запуска программы с различными входными значениями приведены в таблице 4.

Таблица 4 – Работа программы на тестовых наборах данных

Входные данные	Результат
$x_1=3$, $y_1=2$, $x_2=8$, $y_2=3$, $x_3=13$, $y_3=4$	YES
$x_1=-1$, $y_1=-2$, $x_2=-4$, $y_2=-8$, $x_3=-16$, $y_3=-32$	YES
$x_1=3$, $y_1=7$, $x_2=5$, $y_2=0$, $x_3=15$, $y_3=4$	NO