

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 № 12

Преподаватель

подпись, дата

Пушкарев К. В.

Студент КИ18-096, 031830645

_____ 01.10.2018

подпись

Котов С.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 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.

Создадим 2 переменные `num1` и `num2` соответственно, значения которых будет введено с клавиатуры.

Так же следует создать переменную `mod`, в которой будет записан остаток от целочисленного деления, вычисленного по формуле $mod = num1 \% num2$;
Переменную `div_i` в которой будет записано частное от целочисленного деления, вычисленного по формуле $div_i = num1 / num2$;
Переменную `div_f` в которой будет записано частное от вещественного деления, вычисленного по формуле $div_f = float(num1) / num2$;
После всех вычислений выведем переменные `mod`, `div_i`, `div_f` на экран.

4 Графическая схема алгоритма

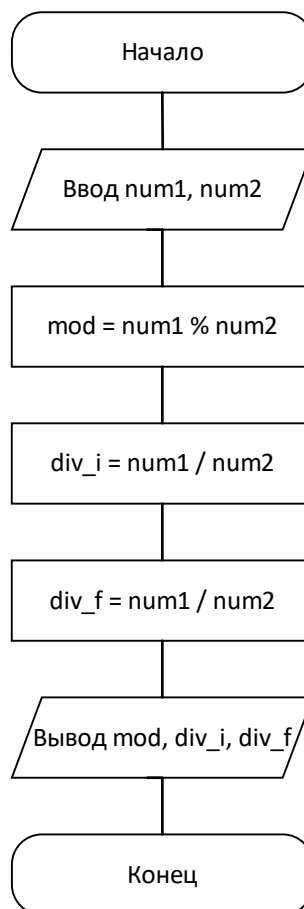


Рисунок 1 – Блок-схема основного упражнения.

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_i;
11         float div_f;
12
13         cout << "Введите первое число: " << endl;
14         cin >> num1;
15         cout << "Введите второе число: " << endl;
16         cin >> num2;
17
18         if (num2 != 0) {
19             mod = num1 % num2;
20             div_i = num1 / num2;
21             div_f = float(num1) / num2;
22
23             cout.setf(ios::fixed);
24             cout.setf(ios::left);
```

```

25     cout.width(40);
26     cout.precision(2);
27     cout.fill('*');
28     cout << "Остаток от целочисленного деления = " << mod << endl;
29     cout.setf(ios::fixed);
30     cout.setf(ios::left);
31     cout.width(50);
32     cout.precision(7);
33     cout.fill('#');
34     cout << "Остаток от целочисленного деления = " << mod << endl;
35     cout.setf(ios::fixed);
36     cout.setf(ios::right);
37     cout.width(75);
38     cout.precision(5);
39     cout.fill('$');
40     cout << "Остаток от целочисленного деления = " << mod << endl << endl;
41
42     cout.setf(ios::fixed);
43     cout.setf(ios::left);
44     cout.width(40);
45     cout.precision(2);
46     cout.fill('*');
47     cout << "Частное от целочисленного деления = " << div_i << endl;
48     cout.setf(ios::fixed);
49     cout.setf(ios::left);
50     cout.width(50);
51     cout.precision(7);
52     cout.fill('#');
53     cout << "Частное от целочисленного деления = " << div_i << endl;
54     cout.setf(ios::fixed);
55     cout.setf(ios::right);
56     cout.width(75);
57     cout.precision(5);
58     cout.fill('$');
59     cout << "Частное от целочисленного деления = " << div_i << endl << endl;
60
61     cout.setf(ios::fixed);
62     cout.setf(ios::left);
63     cout.width(40);
64     cout.precision(2);
65     cout.fill('*');
66     cout << "Частное от вещественного деления = " << div_f << endl;
67     cout.setf(ios::fixed);
68     cout.setf(ios::left);
69     cout.width(50);
70     cout.precision(7);
71     cout.fill('#');
72     cout << "Частное от вещественного деления = " << div_f << endl;
73     cout.setf(ios::fixed);
74     cout.setf(ios::right);
75     cout.width(75);
76     cout.precision(5);
77     cout.fill('$');
78     cout << "Частное от вещественного деления = " << div_f << endl;
79 }
80 else {
81     cout << "Произошла ошибка в вычислениях." << endl;
82 }
83 return 0;
84 }

```

6 Результат выполнения экспериментальной части работы.

Результаты запуска программы с различными входными значениями приведены в таблице 1.

Таблица 1 – Работа программы на тестовых наборах данных

Входные данные	Результат
num1 = 5, num2 = 2	<p>Остаток от целочисленного деления = ****1</p> <p>Остаток от целочисленного деления = #####1</p> <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 = 2.500000</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 = 3.200000</p> <p>Частное от вещественного деления = 3.20000</p>

7 Дополнительное упражнение № 1

2. Грузовой автомобиль выехал из одного города в другой со скоростью v_1 км/ч. Через t ч в этом же направлении выехал легковой автомобиль со скоростью v_2 км/ч. Составить программу, определяющую, догонит ли

легковой автомобиль грузовой через t_1 ч после своего выезда. Значения v_1 , v_2 , t , t_1 вводятся с клавиатуры.

8 Подход к решению задачи дополнительного упражнения № 1

Согласно условию задачи, требуется разработать программу, которая будет определять, догонит ли легковой автомобиль грузовой через t_1 часов после своего выезда. Создаем переменные типа float: v_1 , v_2 , t и t_1 . Запрашиваем ввод значений с клавиатуры в эти переменные. Создаем дополнительные вспомогательные переменные типа float: s_1 и s_2 . Они нам нужны, для вычисления пути, который прошли автомобили. s_1 будет вычисляться по формуле $s_1 = v_1 * (t + t_1)$, а s_2 соответственно по формуле $s_2 = v_2 * t_1$. Сравним полученные значения s_1 и s_2 и определим, сможет ли легковой автомобиль догнать грузовой.

9 Код программы дополнительного упражнения № 1

```
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 v1, v2, t, t1;
11
12         cout << "Введите скорость грузового автомобиля: ";
13         cin >> v1;
14         cout << "Введите скорость легкового автомобиля: ";
15         cin >> v2;
16         cout << "Введите время выезда легкового автомобиля: ";
17         cin >> t;
18         cout << "Введите время легкового автомобиля в пути: ";
19         cin >> t1;
20
21         float s1, s2;
22
23         s1 = v1 * (t + t1);
24         s2 = v2 * t1;
25
26         if (s1 <= s2) {
27             cout << "\nЛегковой автомобиль догонит грузовой за это время." << endl;
28         }
29         else {
30             cout << "\nЛегковой автомобиль не сможет догнать грузовой за это время." <<
31             endl;
32         }
33
34         return 0;
35     }
```

10 Результат выполнения экспериментальной части работы дополнительного упражнения № 1

Результаты запуска программы с различными входными значениями приведены в таблице 1.

Таблица 1 – Работа программы на тестовых наборах данных

Входные данные	Результат
$v1 = 50$ $v2 = 90$ $t = 1$ $t1 = 3$	Легковой автомобиль догонит грузовой за это время.
$v1 = 75$ $v2 = 80$ $t = 2$ $t1 = 6$	Легковой автомобиль не сможет догнать грузовой за это время.