**Назначение:** продолжаемучиться реализовывать более сложные разветвляющийся алгоритмы в С++.

**Теория**

Условный оператор if может также содержать условие (а часто два и три и более). Но практика программирования показывает, что если возникает необходимость во множестве вложенных условий, то целесообразно изменить сам алгоритм.

Рассмотрим решение следующей задачи: выполнить контроль введенного дня недели. Контроль выполнить для следующих ситуаций:

1) значение должно быть целым числом и

2) число в диапазоне от 1 до 7

Алгоритм проверки такой:

1. Введено число? Если нет, то выдать сообщение «Введено не число»
2. Иначе проверяем далее: число входит в диапазон от 1 до 7? Если да, то контроль завершен.
3. Иначе выдать сообщение «Вне диапазона»

Код программы может быть таким:

**#include <iostream>**

**using namespace std;**

**int main()**

**{**

**int k;**

**cout << "Введите день недели – ";**

**cin >> k;**

**if (cin)**

**{**

**if (k>=1 && k<=7) cout << "День недели введен правильно";**

**else cout << "День недели введен НЕ правильно";**

**}**

**else cout << "Вы ввели не число";**

**}**

Красным покрашено первое условие, зеленым – второе, вложенное.

Констукция ***if (cin)*** позволяет проверить на число, т.к. ***cin*** можно рассматривать как булево значение, принимающего true для числа и false для не числа.

Для проверки требуется выполнять запуск программы три раза минимально:

1. День недели введен верно. Например, 5
2. Введен символ. Например, a
3. Число вне диапазона. Например, 22

Рассмотрим еще один пример:

1. выполняется ввод одного символа.
2. Если введен символ **a**, то вводим второй символ
3. Если не **a**, то сообщение об ошибке и конец работы.
4. Если второй символ **b**, то выдаем два символа на экран
5. Иначе сообщение об ошибке

Код программы:

**#include <iostream>**

**using namespace std;**

**int main()**

**{**

**char sym1, sym2;**

**cout << "Введите первый символ – ";**

**cin >> sym1;**

**if (sym1=='a')**

**{**

**cout << "Введите второй символ - ";**

**cin >> sym2;**

**if (sym2=='b') cout << sym1 << sym2;**

**else cout << "Второй символ НЕ b";**

**}**

**else cout << "Первый символ НЕ a";**

**}**

Для проверки требуется выполнять запуск программы три раза минимально:

1. Символы введены верно. Например, a и b
2. Первый символ не a. Например, c
3. Второй символ не b. Например, d

**Практика**

***Общее условие:***

Ввести два значения в разные переменные. Имена придумать самостоятельно. Выполнит контроль введенных значений. Если контроль прошел, то выдать значения на экран. Если контроль не прошел, то выдать сообщение об ошибке.

Составить программу с условиями для следующего своего варианта:

1. Два целых числа. Проверить первое на число, если контроль прошел, то ввести второе менее -2 и более 5. Выдать сообщения об ошибках и если контроль прошел, то выдать введенные значения на экран.
2. Два целых числа. Проверить первое на число, если контроль прошел, то ввести второе. Второе должно быть число 7. Выдать сообщения об ошибках и если контроль прошел, то выдать введенные значения на экран.
3. Два целых числа. Первое должно быть число 10. Если контроль прошел, то ввести второе. Выполнить контроль второго значения на число. Выдать сообщения об ошибках и если контроль прошел, то выдать введенные значения на экран.
4. Два целых числа. Первое должно быть в диапазоне 1-20. Если контроль прошел, то ввести второе. Выполнить контроль второго значения на число. Выдать сообщения об ошибках и если контроль прошел, то выдать введенные значения на экран.
5. Два символа. Первый должен быть q. Если контроль прошел, то ввести второй символ. Второй символ должен быть w. Выдать сообщения об ошибках и если контроль прошел, то выдать введенные значения на экран.
6. Одно целое число и один символ. Контроль первого на число. Если контроль прошел, то ввести символ. Символ должен быть %. Выдать сообщения об ошибках и если контроль прошел, то выдать введенные значения на экран.
7. Один символ и целое число. Символ должен быть &. Если контроль прошел, то ввести число. Выполнить контроль на число. Выдать сообщения об ошибках и если контроль прошел, то выдать введенные значения на экран.
8. Два символа. Первый должен быть @. Если контроль прошел, то ввести второй символ. Второй символ должен быть 1. Выдать сообщения об ошибках и если контроль прошел, то выдать введенные значения на экран.
9. Два символа. Первый должен быть a или b. Если контроль прошел, то ввести второй символ. Второй символ должен быть ). Выдать сообщения об ошибках и если контроль прошел, то выдать введенные значения на экран.
10. Два целых числа. Проверить первое на число, если контроль прошел, то ввести второе. Второе должно быть число в диапазоне 10-15. Выдать сообщения об ошибках и если контроль прошел, то выдать введенные значения на экран.
11. Один символ и целое число. Символ должен быть x. Если контроль прошел, то ввести число. Выполнить контроль на число. Выдать сообщения об ошибках и если контроль прошел, то выдать введенные значения на экран.
12. Два символа. Первый должен быть 1 или 2. Если контроль прошел, то ввести второй символ. Второй символ должен быть ). Выдать сообщения об ошибках и если контроль прошел, то выдать введенные значения на экран.