



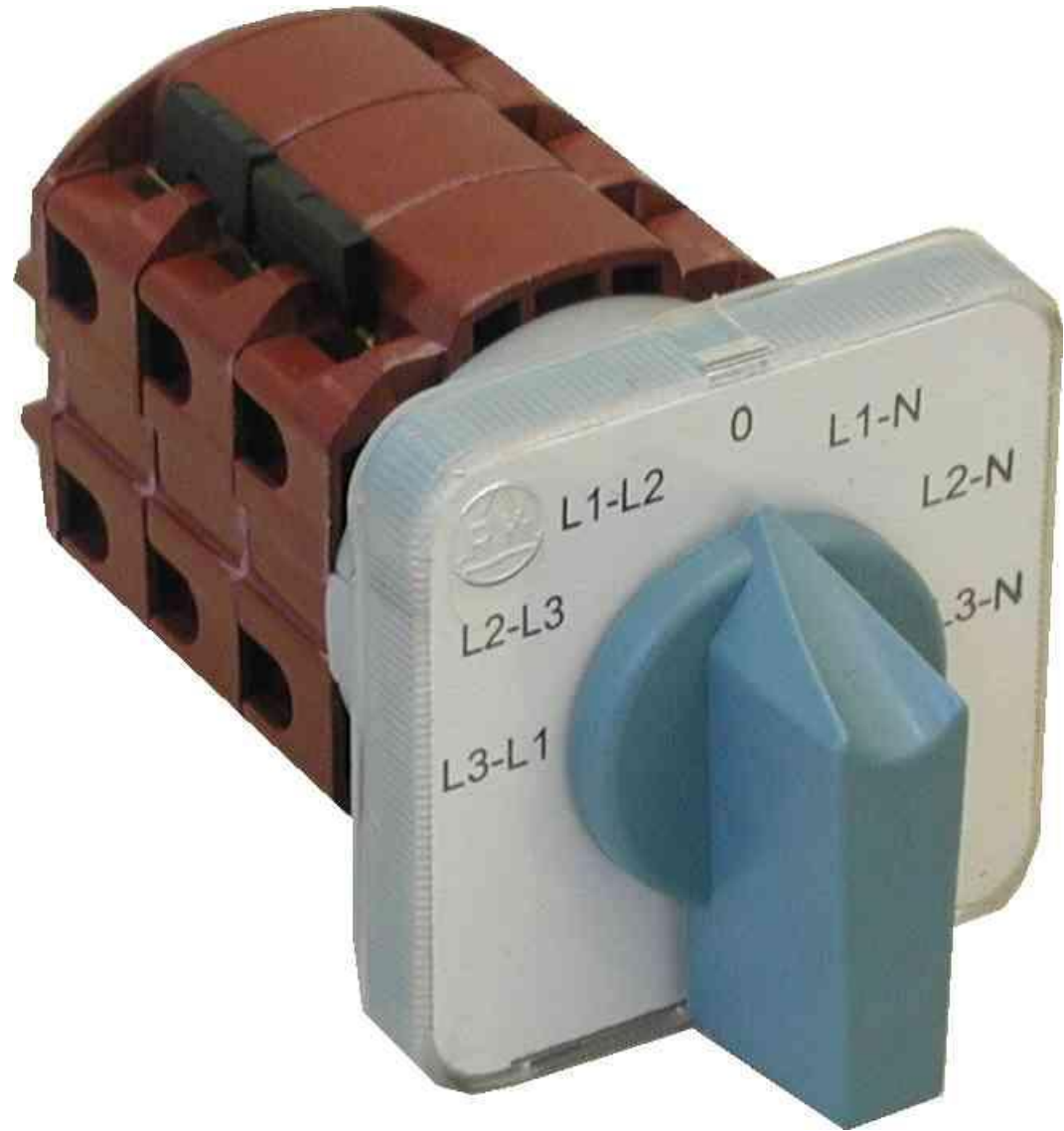




# Оператор множественного выбора

switch – переключатель.





# Структура множественного выбора **switch**

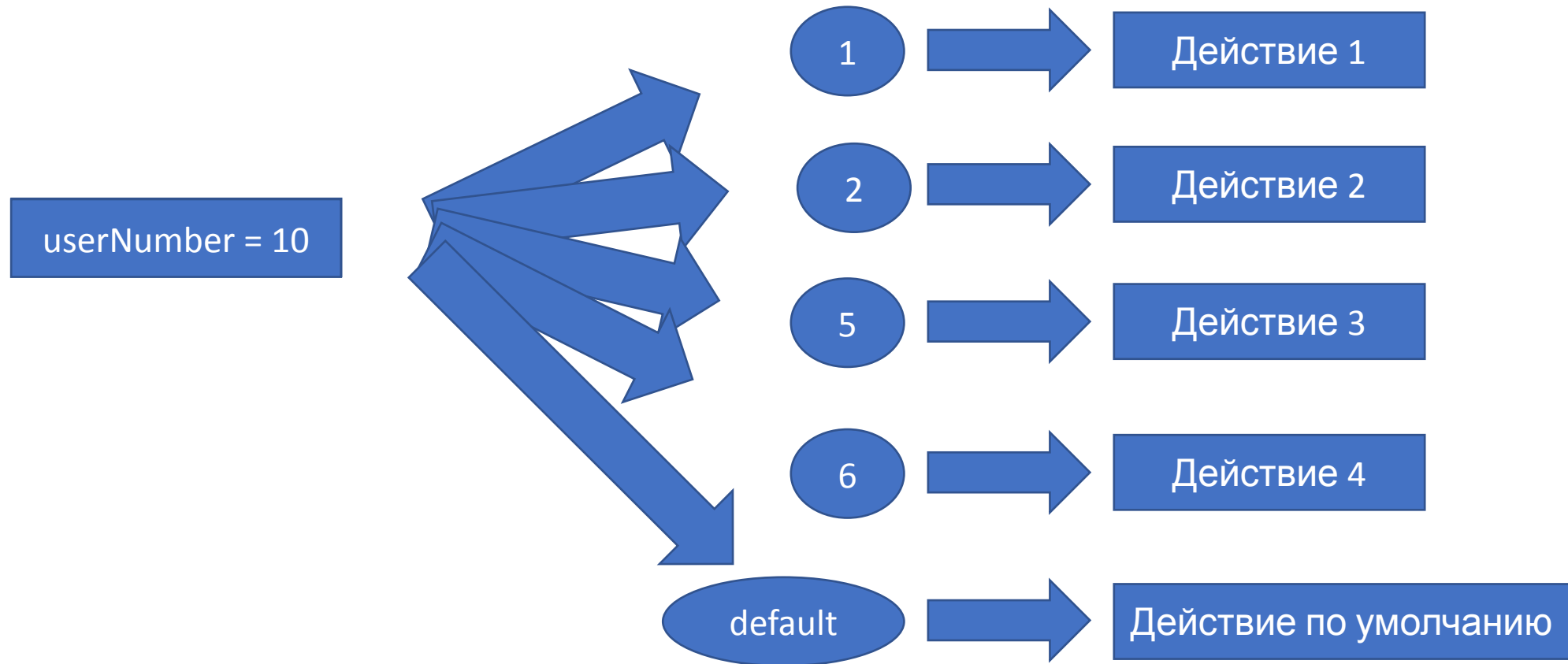
Оператор выбора **switch** является очень удобной заменой множественного использования операторов **if**.

Оператор **switch** сравнивает значение одной переменной с несколькими константами.

Значение переменной указанной в условии **switch** сравнивается со значениями, которые следуют за ключевым словом **case**.

Когда значение в переменной, соответствует значению в строке с оператором **case**, компьютер продолжит выполнение программы с этого места.

# Структура множественного выбора **switch**



```
switch(выражение) {
```

```
  case значение1:
```

```
    действие1;
```

```
    и ещё одно действие1;
```

```
  break;
```

```
  case значение2:
```

```
    действие2;
```

```
  break;
```

```
  case значение3:
```

```
    действие3;
```

```
  break;
```

```
.....
```

```
  default: // default не обязательный раздел, может быть, может не быть.
```

```
    действие_по_умолчанию;
```

```
  break;
```

```
}
```

```
short userNumber;  
cin>> userNumber;  
switch(userNumber) {  
    case 1:  
        cout << "действия 1";  
        cout << «ещё действия 1»;  
    break;  
    case 2:  
        cout << "действия 2";  
    break;  
    case 5:  
        cout << "действие 3";  
    break;  
    case 6:  
        cout << "действие 4";  
    break;  
    default:  
        cout << "действия по умолчанию";  
    break;  
}
```



## Попробуйте убрать все **break**

```
short userNumber;
```

```
cin>> userNumber;
```

```
switch(userNumber) {
```

```
  case 1:
```

```
    cout << "действия 1";
```

```
    cout << «ещё действия 1";
```

```
  case 2:
```

```
    cout << "действия 2";
```

```
  case 5:
```

```
    cout << "действие 3";
```

```
  case 6:
```

```
    cout << "действие 4";
```

```
  default:
```

```
    cout << "действия по умолчанию";
```

```
  break;
```

```
} // и потестируйте работу программы, поймите как оно работает без break
```

# Структура множественного выбора **switch**

- Оператор **default** можно вообще не указывать, в случае, если не найдется совпадений, просто ничего не произойдет.
- В случае, если необходимо **выполнять один и тот же набор** действий для разных значений проверяемого выражения, можно записывать несколько меток подряд.

```
switch (cRate) {  
    case 6  
        cout<<«Оценка 6»;  
    break;  
    case 7  
    case 8:  
        cout<<«Оценка выше 6»;  
    break;  
    default:  
        cout<<«Оценка не предусмотренная»;  
    break;  
}
```

# Структура множественного выбора **switch**

- Если в case используются символьные значения, они должны указываться в одинарных кавычках, если целочисленные, то без кавычек.
- Оператор default может располагаться в любом месте системы switch, выполняться он все равно будет в том случае, если нет ни одного совпадения.
- Однако правилом «хорошего тона» является указывать default в конце всей конструкции.



# Структура множественного выбора **switch**

- Если случайно пропустить `break` в любом блоке `case`, кроме последнего, и этот блок в последствии отработает, то выполнение `switch` не остановится.
- Тот блок оператора `case`, который будет идти вслед за уже выполненным, так же выполнится без проверки.
- После **самого последнего** оператора в списке (будь то `case` или `default`) оператор `break` можно не указывать.

# Структура множественного выбора **switch**

- Оператор **default** можно вообще не указывать, в случае, если не найдется совпадений, просто ничего не произойдет.

```
float numberA, numberB, result;
cout << «Введите число A:\n";
cin >> numberA;
cout << «Введите число B:\n";
cin >> numberB;
// реализация программного меню
char arithmeticOperation;
cout << "\nВыберите арифметическую операцию:\n";
cout << "\n +  если Вы хотите увидеть сумму.\n";
cout << "\n -  если Вы хотите увидеть разность.\n";
cout << "\n *  если Вы хотите увидеть произведение.\n";
cout << "\n /  если Вы хотите увидеть частное.\n";
```



```
// ожидание выбора пользователя
cin >> arithmeticOperation;
// проверяется значение переменной key
switch (arithmeticOperation) {
case '+': // если пользователь выбрал сложение
    result = numberA + numberB;
    cout << "\nОтвет: " << result << "\n";
    break; // parada do switch
case '-': // если пользователь выбрал вычитание
    result = numberA - numberB;
    cout << "\nОтвет : " << result << "\n";
    break; // остановка switch
case '*': // если пользователь выбрал умножение
    result = numberA * numberB;
    cout << "\nОтвет : " << result << "\n";
    break;
```

**case** '/':// остановка switch case '/':

    // если пользователь выбрал деление

if (numberB) { // если делитель не равен нулю

    result = numberA / numberB;

    cout << "\nОтвет : " << result << "\n";

}

else { // если делитель равен нулю

    cout << "\nОтвет!!! Divide by null!!!!\n";

}

**break;** // остановка switch

**default:** // если введенный символ некорректен

    cout << "\nError!!! This operator isn't correct\n";

**break;** // остановка switch

}