1. **Операторы управления потоком. Условные операторы. Итерационные операторы. Операторы перехода (break, continue, goto).**

**Операторы управления потоком исполнения** позволяют изменять последовательность выполнения кода.

**Условные операторы**

*if* (условие) {

// Блок кода, выполняемый при истинном условии

} *else* {

// Блок кода, выполняемый при ложном условии

}

*switch* (выражение) {

case значение1:

// Блок кода, выполняемый при значении1

break;

case значение2:

// Блок кода, выполняемый при значении2

break;

default:

// Блок кода, выполняемый при отсутствии совпадений

break;

}

Особенности:

* Если не будет break, после выполнения первого блока кода (при первом true), компилятор продолжит выполнять команды, пока не дойдет до конца, даже не смотря на условия.

**Итерационные операторы**

*for* (инициализация; условие; инкремент) {

// Блок кода, выполняемый в каждой итерации

}

*while* (условие) {

// Блок кода, выполняемый в каждой итерации, пока условие истинно

}

*do* {

// Блок кода, выполняемый в каждой итерации, хотя бы один раз

} while (условие);

**Операторы перехода**

* **break** - выход из цикла.
* **continue** - начать следующую по счету итерацию
* **goto** - определитель метки, т.е. сначала ты пишешь "goto NameTag", а затем где-то в коде ставишь метку с этим имененим(NameTag) и когда компилятор дойдет до объявления метки он перескачет в место второй метки и продолжит исполнение оттуда.