1. **Выражения языка С++. Порядок выполнения команд. Приоритет и ассоциативность операторов. Арифметические операторы. Логические операторы и операторы отношения. Операнды.**

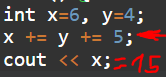
**Выражения —** это комбинации операторов и операндов. Операнды – это любые значения, над которыми выполняются операции.

**Порядок выполнения команд –** компилятор выполняет команды последовательно сверху вниз. Порядок выполнения команд может быть изменен с условными операторами, циклами, переходами и функциями.

**Приоритет и ассоциативность операторов**

Ассоциативность - порядок выполнения операций, когда есть несколько операторов с одинаковым приоритетом.

Может быть «слева направо» или «справа налево»

Приоритеты (от ↑ к ↓)

* Область видимости (::)
* Скобки () - управление порядком выполнения операций

Приведение типов static\_cast

Постфиксные ++, --

* Унарные операторы (выполняются над одним оператором: ++x, !, разыменование \* и взятие ссылки &, -x +x):

Ассоциативность: «справа налево»

* Арифметические операторы
* Операторы отношения (<, >, <=, >=):
* Операторы равенства (==, !=):
* Побитовые операторы (&, | ):
* Логический операторы ( &&, | | ):
* Тернарный условный оператор (?:):
* Операторы присваивания (=, +=, -=, \*=, /=, %=)

Ассоциативность «справа налево»:

* Оператор запятая (,)

**Арифметические операторы** нужны для выполнения математических операций над числами:

Сложение (+), вычитание (-), умножение (\*), деление (/), остаток от деления (%)

Операции с присваиванием (+=, -=, \*=, /=, %=)

Декремент (--) и инкремент (++) уменьшают переменную на единицу.

Есть два вида:

* постфиксный (i++) – сначала возвращает значение, потом меняет переменную
* префиксный (++i) – сначала меняет значение, потом возвращает

**Логические операторы и операторы отношения**

* Логические:

И (&&): true, если оба операнда истинны.

ИЛИ (||): true, если хотя бы один из операндов истина.

НЕ (!): инвертирует значение операнда.

* Отношения:

Равно (==): Возвращает true, если значения операндов равны.

Не равно (!=): Возвращает true, если значения операндов не равны.

Больше (>): Возвращает true, если левый операнд больше правого.

Меньше (<): Возвращает true, если левый операнд меньше правого.

Больше или равно (>=): Возвращает true, если левый операнд больше или равен правому.

Меньше или равно (<=): Возвращает true, если левый операнд меньше или равен правому.

**Операнды:**

* переменные (int a, float b)
* константы (const int a)
* литералы (фиксированные значения конкретных типов данных: 5, 8.9, true)
* вызовы функций, возвращающих значение