

Национальный исследовательский университет
«Высшая Школа Экономики»
Факультет компьютерных наук

C++ как первый язык программирования

Лекция №3

Лекторы: Кирилл Лунёв, Роман Халкечев

Цели лекции и семинара

- Продолжить знакомство с операторами в C++
- Изучить основные инструкции C++ (if, while, do, for, switch, goto, ...)



Поехали!

План лекции

- Про lvalue и rvalue
- Операторы: продолжение
- Инструкции (if, while, do, for, switch, goto, ...)
- Ключевые слова break и continue

Про lvalue и rvalue

Каждое выражение C++ является значением **lvalue** либо значением **rvalue**.

Значение **lvalue** — объект, существующий за пределами одного выражения (объект с именем).

Значение **rvalue** — это временное значение, которое не сохраняется за пределами выражения, в котором оно используется.

Вопрос

Какие из этих выражений корректны?

```
int x = 3 + 4;
```

```
cout << x << endl;
```

```
7 = x;
```

```
x * 7 = 49;
```

Вопрос

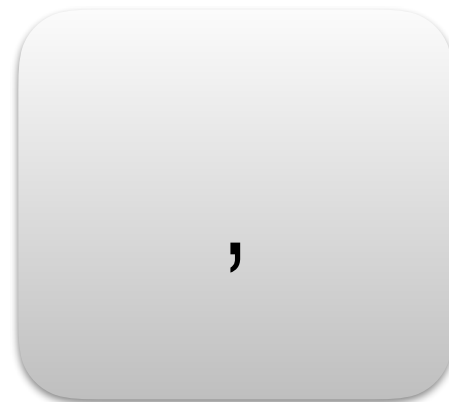
Какие из этих выражений корректны?

`int x;`

`-x = 49;`

`+x = 49;`

Оператор запятая



Вопрос

Что выведется на экран?

...

```
double a;
```

```
a = 1,2;
```

```
cout << a << endl;
```

...

Вопрос

Что выведется на экран?

...

```
double a;
```

```
a = (1, 2);
```

```
cout << a << endl;
```

...

Тернарный оператор

? :

<условие> ? оператор1 : оператор2;

```
int max = (a > b) ? a : b;
```

Операторы инкремента

Постфиксный ++

Постфиксный --

Префиксный ++

Префиксный --

Вопрос

Что выведется на экран?

...

```
int x = 10;
```

```
cout << x++ << endl;
```

```
cout << ++x << endl;
```

...

Вопрос

Что из этого корректно?

```
int i = 10;
```

```
i++ = 50;
```

```
++i = 50;
```

Вопрос

Что из этого корректно?

```
int i = 10;
```

```
++++i;
```

```
i++++;
```

```
++i++;
```

Приоритеты операторов

| | | | |
|---|--------------------------------------|----|---------------------------|
| 1 | (), постфиксные ++, -- | 8 | & |
| 2 | !, ~, унарные +, -, префиксные +, -- | 9 | ^ |
| 3 | *, /, % | 10 | |
| 4 | +, - | 11 | && |
| 5 | <<, >> | 12 | |
| 6 | <, >, <=, >= | 13 | ?:, =, +=, -=, *=, /=, %= |
| 7 | !=, == | 14 | , |

Приоритеты операторов

Некоторые комментарии:

- Арифметические бинарные операторы — приоритеты как в математике
- Постфиксные операторы имеют больший приоритет, чем префиксные
- Если есть сомнения — лучше ставить скобки

Инструкции (statements)

- if
- for
- while
- do while
- switch
- goto



Инструкция if

```
if (<условие>) {
```

```
    оператор;
```

```
    оператор;
```

```
    ...
```

```
} else {
```

```
    ...
```

```
}
```

Инструкция if

```
int applesCount = 5;  
  
if (applesCount > EnoughCountForPie()) {  
    std::cout << "We have enough apples\n";  
  
} else {  
    std::cout << "We need more apples\n";  
  
}
```

Инструкция while

```
while (<условие>) {  
    оператор;  
    оператор;  
    ...  
}
```

Инструкция while

```
std::vector<int> numbers = {1, -1, 23, -42, 42}
```

```
int i = 0;
```

```
while (i < numbers.size() && numbers[i] != 23) {
```

```
    ++i;
```

```
}
```

Инструкция do while

```
do {
```

```
    оператор;
```

```
    оператор;
```

```
    ...
```

```
} while (<условие>);
```

Инструкция do while

```
std::vector<int> numbers = {1, -1, 23, -42, 42}
```

```
int i = 0;
```

```
do {
```

```
    ++i;
```

```
} while (i < numbers.size() && numbers[i] != 23)
```

Инструкция for

```
for (<до начала>; <условие>; <после каждой итерации>) {  
    оператор;  
  
    оператор;  
  
    ...  
}
```


Инструкция for

```
std::vector<int> a = {1, -1, 23, -42, 42};
```

```
std::vector<int> b = {0, 10, 2};
```

```
for (int i = 0, j = b.size(); i < a.size() && j > -1; ++i, --j) {
```

```
    ...
```

```
}
```

Ключевые слова `break` и `continue`

- `break` — выход из цикла
- `continue` — переход на следующую итерацию цикла

Инструкция for

```
std::vector<int> a = {1, -1, 23, -42, 42};
```

```
std::vector<int> b = {0, 10, 2};
```

```
for (int i = 0, j = b.size(); i < a.size() && j > -1; ++i, --j) {
```

```
    ...
```

```
    if (a[i] == b[j]) {
```

```
        break;
```

```
    }
```

```
    ...
```

```
}
```

Инструкция goto

L:

goto L;

Инструкция switch

```
switch (<переменная простого типа>) {
```

```
    case value1:
```

```
        оператор;
```

```
        оператор;
```

```
    case value2:
```

```
        оператор;
```

```
        ...
```

```
    ...
```

```
    default:
```

```
        оператор;
```

```
        ...
```

```
}
```

Инструкция switch

```
switch (x) {  
    case 0:  
        x += 10;  
        break;  
    case 1:  
        x += 20;  
        break;  
    default:  
        x = -1;  
        std::cout << "Strange value\n";  
}
```

Пустая инструкция

;

Вопрос

Что выведется на экран?

```
int a = 0;
```

```
while (a < 10);
```

```
    std::cout << ++a << std::endl;
```


Вопросы?

Домашнее задание

- Прочесть главы 3-6 в книге Стивена Прата «Язык программирования C++» (кто ещё не успел прочесть)
- Дорешать задачи с семинара. Решения залить в свой репозиторий.