

## Лекция 2

# Основные компоненты языка программирования

---

Преподаватель Палехова Ольга Александровна,  
кафедра О7 БГТУ «Военмех»

# Языки программирования

---



**Язык программирования** – формальная знаковая система, предназначенная для записи компьютерных программ.

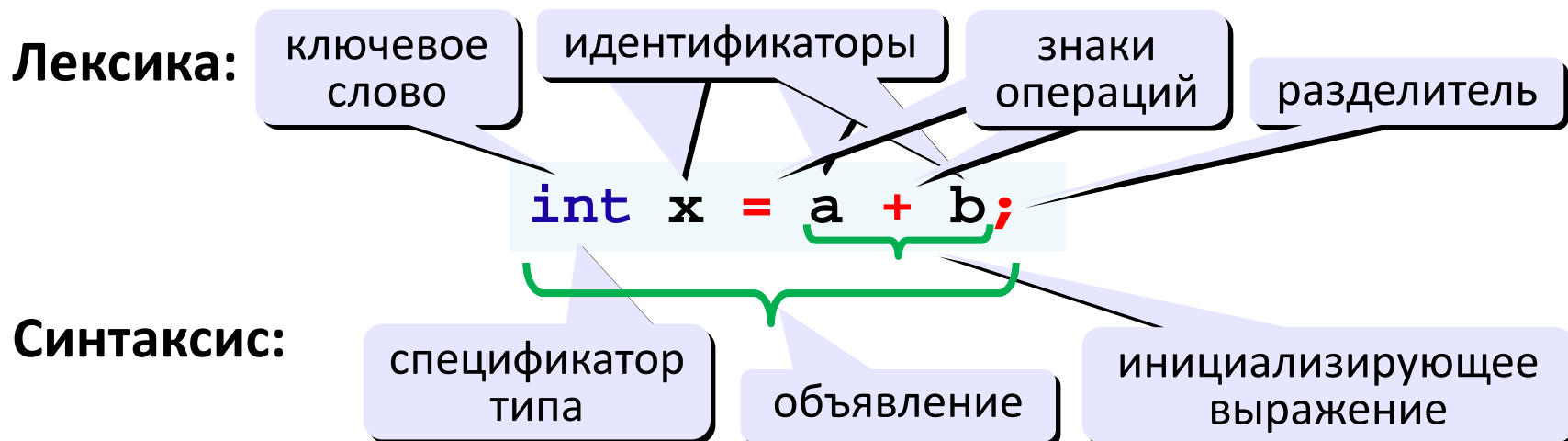
Язык программирования определяет набор **лексических, синтаксических и семантических** правил, задающих внешний вид программы и действия, которые выполнит исполнитель (компьютер) под её управлением.

# Лексика, синтаксис, семантика

**Лексика** – совокупность слов языка.

**Синтаксис** – правила написания языковых конструкций.

**Семантика** – правила истолкования языковых конструкций.

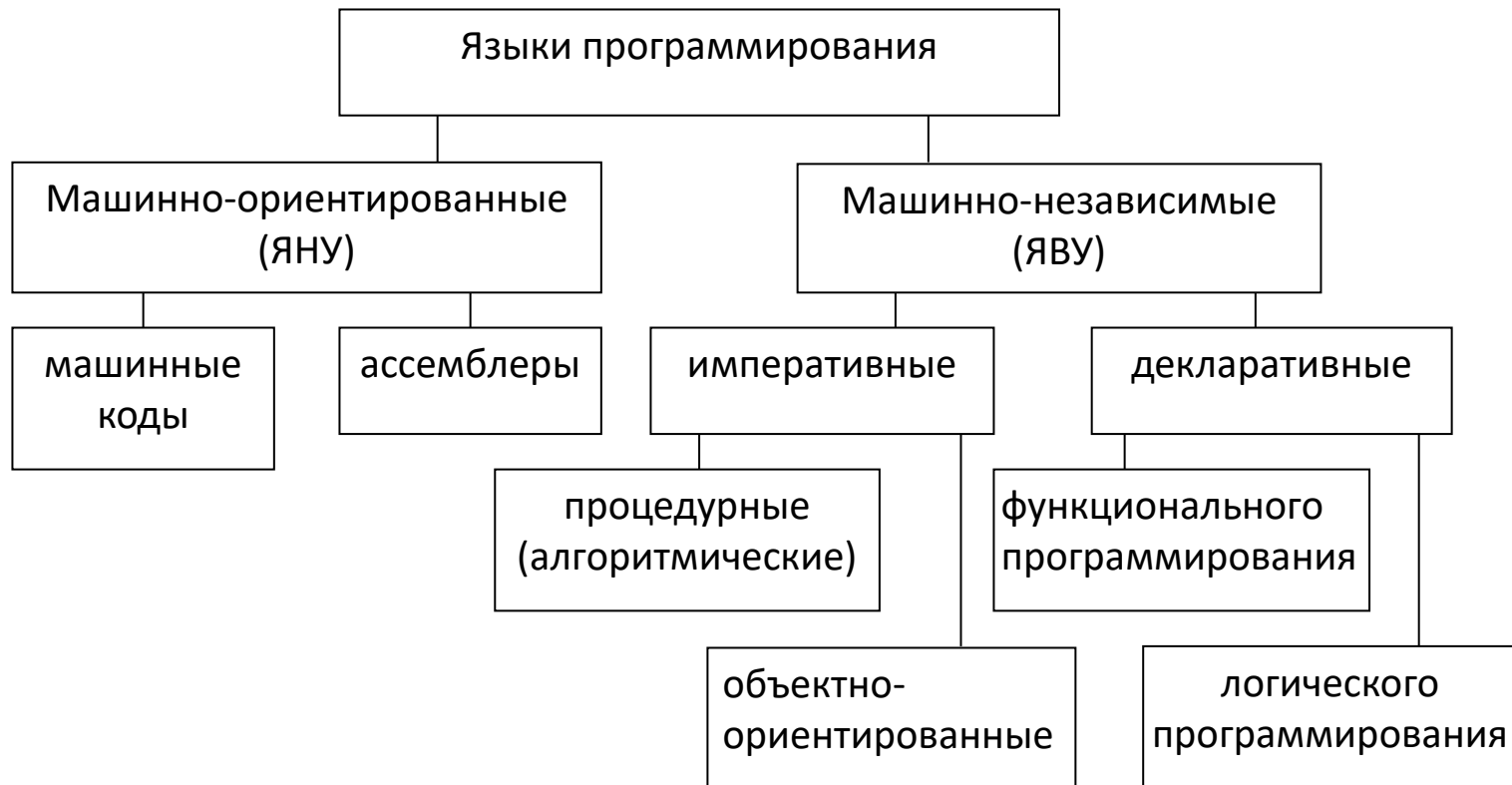


**Семантика:**

- выделить в стеке память для хранения новой переменной;
- вычислить значение выражения-инициализатора;
- определить тип выражения-инициализатора, выполнить преобразование типа результата к целому типу;
- сохранить полученное значение в переменной x;
- сделать переменную x доступной в текущем контексте.

# Классификация языков программирования

- По зависимости от архитектуры процессора: языки низкого уровня и языки высокого уровня.
- По семантике: императивные и декларативные языки.
- По назначению: универсальные и специальные.
- По типизации: языки со статической (сильной) и динамической (слабой) типизацией.
- По способу обработки программ: компилируемые и интерпретируемые.



# Интегрированная среда разработки

---

## Интегрированная среда разработки, ИСР

(англ. *Integrated development environment* — **IDE**), также **единая среда разработки, ЕСР** — комплекс программных средств, используемый программистами для разработки программного обеспечения.

Среда разработки включает в себя:

- текстовый редактор,
- компилятор и/или интерпретатор,
- компоновщик (редактор связей, linker),
- отладчик.

# Алгоритм

---



**Алгоритм** – строго определенная последовательность действий, позволяющая получить удовлетворяющий поставленным условиям результат по исходным данным.

## Свойства алгоритма:

- дискретность,
- детерминированность (определенность),
- понятность,
- конечность (результативность),
- массовость.

# Запись алгоритма

Способы описания алгоритма:

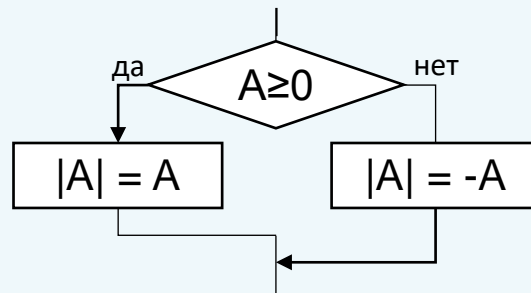
- словесный (на естественном языке)

Чтобы вычислить модуль числа, надо сравнить его с нулем. Если число положительное или равно 0, то модуль равен самому числу. А если число отрицательное, то модуль будет равен противоположному числу.

- словесно-формульный (псевдокод)

Если  $A \geq 0$ , то  $|A| = A$ , иначе  $|A| = -A$

- графический (схема алгоритма по ГОСТ 19.701-90)



- на языке программирования

```
abs_a = a >= 0 ? a : -a;
```

# ГОСТ 19.701-90

Название	Обозначение	Пояснение
Процесс		Действие или последовательность действий по обработке данных, приводящих к их изменению
Решение		Проверка условия - операция с одним входом и несколькими альтернативными выходами
Границы цикла		Начало и конец цикла. Условия для инициализации, приращения и завершения помещаются внутри символа начала или конца в зависимости от расположения операции, проверяющей условие
Предопределенный процесс		Процесс, состоящий из операций, описанных в другом месте (на другой схеме) – обращение к вспомогательному алгоритму.
Ввод-вывод		Ввод-вывод в общем виде
Пуск-останов		Начало, конец алгоритма, вход в подпрограмму и выход



# Основные компоненты языка программирования

---

**Алфавит** – совокупность символов, используемых для записи языковых конструкций.

**Лексема** – неделимая последовательность символов, рассматриваемая компилятором как единое целое.

**Выражение** – синтаксически допустимая последовательность операндов и операций, позволяющая вычислить значение.

**Инструкция** (оператор, команда) – наименьшая автономная часть программы, выполняющая какое-то действие. Программа обычно представляет собой последовательность инструкций.

# Структура программы на языке Си

```
/* Перевод каратов в граммы */
```

Директивы  
препроцессора

```
#include <stdio.h>
```

```
double carat_gramm (double);
```

Глобальные  
объявления

```
int main()
```

Главная функция

Тело  
функции

```
{  
    double carat, gramm;  
    printf ("Введите массу в каратах\n");  
    scanf ("%lf", &carat);  
    gramm = carat_gramm (carat);  
    printf ("\n%f карат = %f грамм\n", carat, gramm);  
    return 0;  
}
```

```
double carat_gramm (double x)
```

Определения  
функций

```
{  
    return .2 * x;  
}
```

# Алфавит языка Си

---

Алфавит языка Си включает:

- прописные и строчные буквы латинского алфавита
- арабские цифры от 0 до 9
- специальные символы:  
+ − ∗ / \ = > < [ ] { } ( ) ; \_ . , : # % ~ ? ! ' "  
&(амперсанд)
- пробельные (разделительные) символы: пробел, символы табуляции, перевода строки, возврата каретки



# Классы лексем языка Си

---

Шесть классов лексем:

- идентификаторы,
- ключевые слова,
- константы,
- строковые литералы,
- знаки операций,
- разделители (символы-разделители и комментарии).

```
/* комментарий */
```

```
/* многострочный  
комментарий */
```

```
// однострочный комментарий (начиная со стандарта C99)
```

# Цветовая подсветка лексем

---

Текстовые редакторы IDE выделяют различные типы лексем цветом.

```
int main()
{
    char c = '1';
    int a; /* комментарий */
    double x, res;
    scanf ("%d%lf", &a, &x);
    res = 2.8 * a / x + c;
    printf ("res = %f\n", res);
    return 0;
}
```

■ ключевые слова

■ идентификаторы

■ знаки операций

■ числовые константы

■ символьные константы

■ строковые литералы

# Ключевые слова языка Си

---

**Ключевые** (зарезервированные, служебные) **слова**

– это лексемы, имеющие определенный смысл, изменять который нельзя.

- спецификаторы и модификаторы типов:

`char, double, int, long, enum, float, signed, short, struct, union, unsigned, void,`

- операторы:

`break, case, continue, default, do, else, for, goto, if, return, switch, while, typedef,`

- классы памяти:

`auto, register, static, extern,`

- квалификаторы:

`const, volatile,`

- операции: `sizeof`

# Идентификаторы в языке Си

---

**Идентификаторы** (имена) – это лексемы, которые служат для обозначения объектов программы (переменных, констант, функций и т. д.).

## Правила написания идентификаторов:

- идентификатор может состоять только из букв латинского алфавита, цифр и знака подчеркивания;
- идентификатор не может начинаться с цифры;
- в идентификаторах прописные и строчные буквы различаются, например MAX, Max, max – это три разных идентификатора;
- идентификатор не может совпадать ни с одним из ключевых слов (int, while).

# Константы в языке Си

---

**Константы** – это лексемы, которые описывают неизменяемые данные. Константы задаются своими значениями.

Тип константы зависит от формы записи. Различают

- *целочисленные константы;*
- *вещественные константы;*
- *символьные константы.*

В отдельную группу выносят *строковые литералы.*