

Развитие парадигм и языков программирования

1. Поколения языков программирования
2. Языки низкого уровня
3. Языки высокого уровня
4. Структурная парадигма
5. Парадигмы программирования
6. История развития объектно-ориентированных языков
7. Сферы разработки ПО: какие языки изучать

Преподаватель:

Ботов Дмитрий Сергеевич

Поколения языков

Generation Language — GL

1GL

- Система команд процессора (Программы в машинном коде)

2GL

- Машинно-ориентированные языки (Программы на ассемблере)

3GL

- Процедурные языки (FORTRAN, COBOL, C, Pascal, Basic)
- Структурное программирование

4GL

- Объектно-ориентированные языки (Java, C++, C#, Delphi, Smalltalk)

5GL

- Декларативные языки, ориентированные на данные (SQL)
- Языки логического программирования (ПРОЛОГ)
- Предметно-ориентированные языки (DSL)
- Событийные языки (Java, Perl, C#)
- Сценарные языки (JavaScript, Perl, PHP, Python)

1GL: Машинный код

- Пример для Intel x86:

```
BB 11 01 B9 0D 00 B4 0E 8A 07 43 CD 10 E2 F9  
CD 20 48 65 6C 6C 6F 2C 20 57 6F 72 6C 64 21
```

- Система команд определяется **архитектурой**
проц

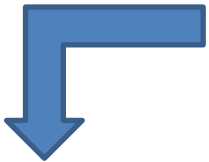


2GL: Ассемблер – язык низкого уровня

- Пример программы на **ассемблере** (assembler):

```
    org     100h
    mov     bx, 111h
    mov     cx, 0Dh
    mov     ah, 0Eh
label:
    mov     al, [bx]
    inc     bx
    int     10h
    loop   label
    int     20h
HW db 'Hello, World!'
```

Трансляц
ия



```
BB 11 01 B9 0D 00 B4 0E 8A 07 43 CD 10 E2 F9
CD 20 48 65 6C 6C 6F 2C 20 57 6F 72 6C 64 21
```

Предпосылки к появлению языков высокого уровня

- Рост производительности **аппаратного обеспечения**
- Рост спроса на разработку **сложных программных систем**
- Высокая сложность разработки, отладки и поддержки программ на **языках низкого уровня**

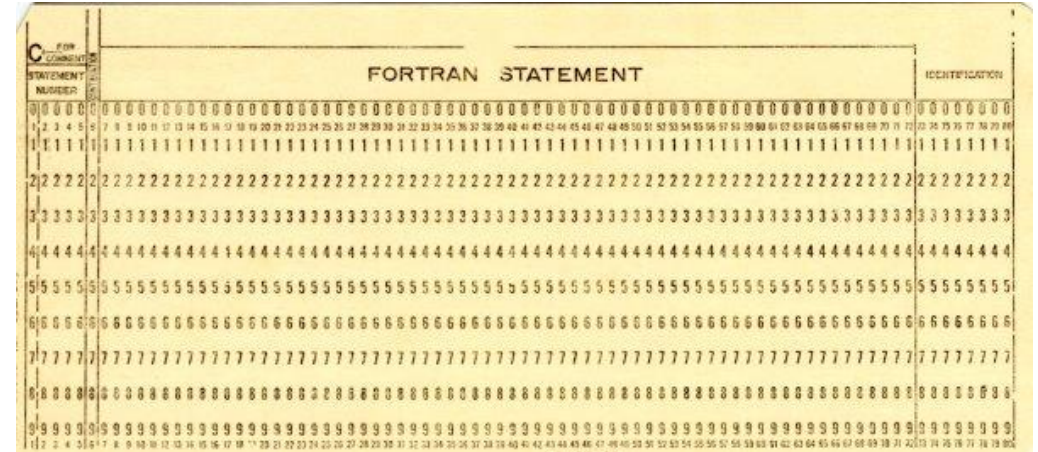


*Первый транзистор
(1947)*

Первые языки высокого уровня

- Fortran (1956)

```
_____ PRINT* , 'Hello, World!'  
_____ END
```



- Переносимость на **уровне исходных кодов**
- Использование **абстракций высокого уровня**:
 - Переменные, массивы
 - Операторы ветвлений, циклов, переходов
 - Функции, процедуры, подпрограммы
 - Работа с памятью
 - Библиотеки: работа с файлами, вводом/выводом и т.д.
- Использование команд на естественном языке

3GL: Процедурные языки

- Программа на языке **Си**:

```
#include <stdio.h>
int main()
{
    printf("Hello, world!\n");
    return 0;
}
```

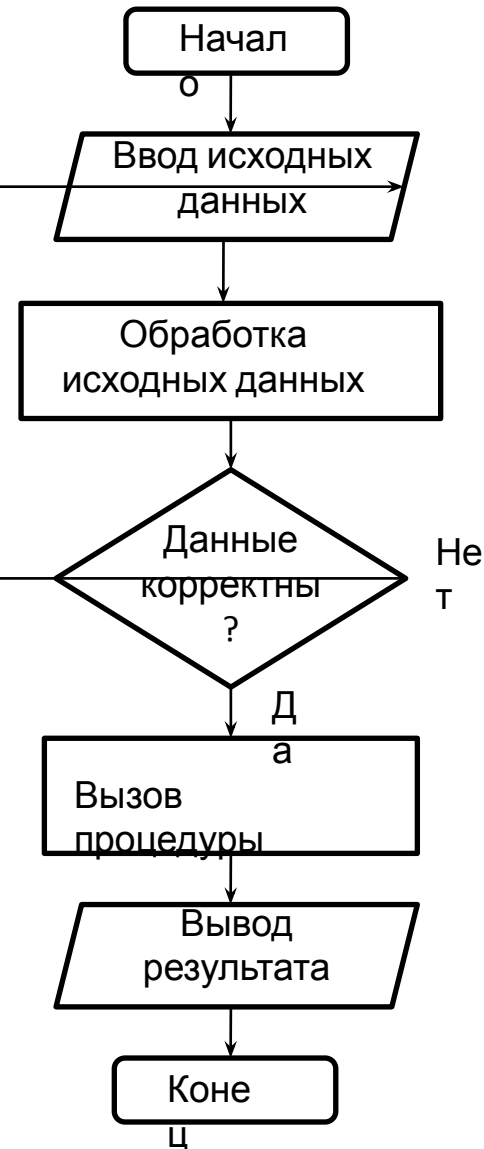
- Программа на языке **Turbo Pascal**:

```
Begin
    WriteLn('Hello, world!');
End.
```

Структурная парадигма

Эдсгер Дейкстра (1968)

- Нет оператора **GOTO**
- Программа состоит из иерархической структуры **блоков**
- Используются **управляющие конструкции**:
 - Последовательность
 - Ветвление
 - Цикл
- Повторяющиеся фрагменты выносятся в **подпрограммы** (процедуры, функции)
- Разработка ведётся пошагово **методом «сверху-вниз»**



Парадигмы программирования

Императивн
ая
Декларативн
ая

Императивная парадигма	Структурное программирование	<ul style="list-style-type: none">• Абстракции: последовательность, ветвление, цикл• Программа состоит из подпрограмм (алгоритмов)
	Объектно-ориентированное программирование	<ul style="list-style-type: none">• Абстракции: классы, объекты, поля, методы• Программа состоит из классов
	Обобщенное программирование	<ul style="list-style-type: none">• Абстракции: обобщенные типы и функции (шаблоны классов/методов)• Программа представлена в виде обобщенных типов и алгоритмов
	Аспектно-ориентированное программирование	<ul style="list-style-type: none">• Абстракции: аспект, совет, точка соединения
Декларативная парадигма	Функциональное программирование	<ul style="list-style-type: none">• Абстракции: функции (лямбда-выражения), рекурсия• Программа состоит из функций
	Логическое программирование	<ul style="list-style-type: none">• Абстракции: предикаты, факты, ограничения• Программа состоит из правил логического вывода
	Языково-ориентированное программирование	<ul style="list-style-type: none">• Перед написанием программ разрабатывается предметно-ориентированный язык программирования

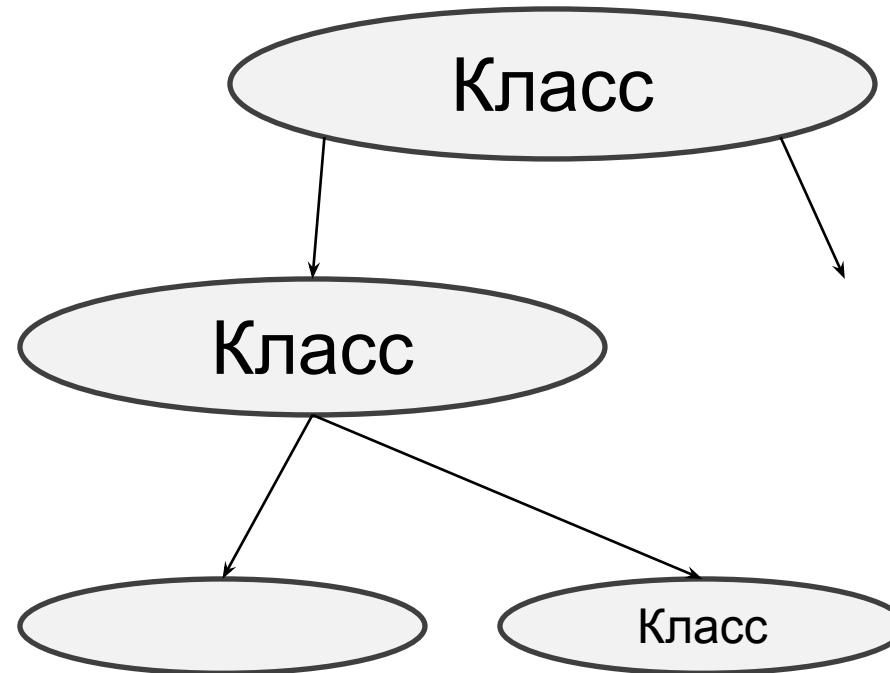
Объектно-ориентированная парадигма

Основные понятия:

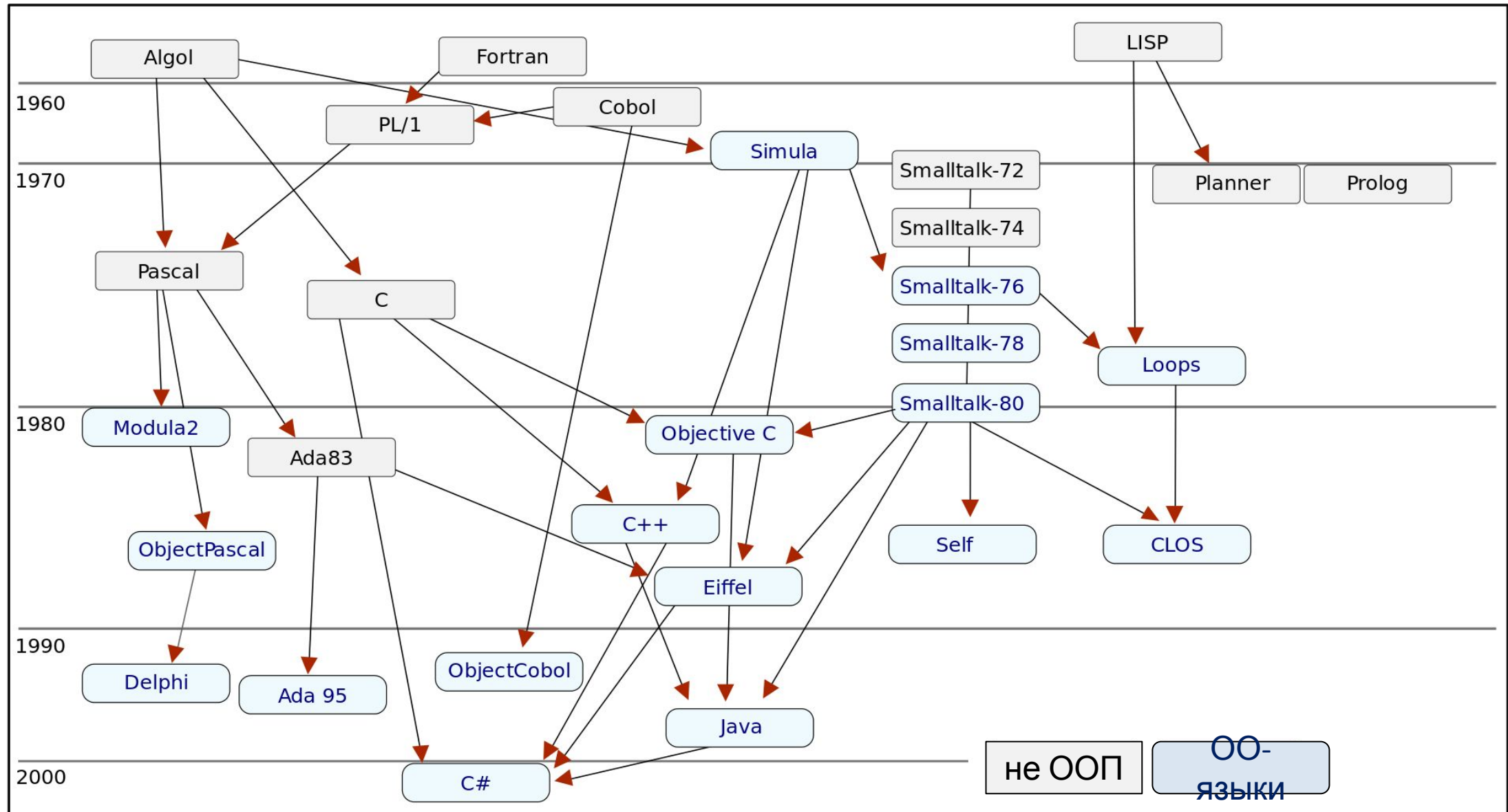
- Класс
- Объект

Основные механизмы:

- Абстрагирование
- Модульность
- Инкапсуляция
- Иерархия (наследование, агрегация)
- Полиморфизм



История развития объектно-ориентированных языков



Популярные языки программирования

ЯЗЫКИ ПРОГРАММИРОВАНИЯ



PYTHON



JAVA



C



PHP



C++



JAVASCRIPT



C#



RUBY



OBJECTIVE-C

Разработка корпоративных приложений (Enterprise Development)

- Приложения для электронного документооборота, автоматизации, контроля, финансовых и иных видов отчетностей
- Языки и базы данных:
 - ❖ Java + Oracle
 - ❖ C# (.Net) + MS SQL
 - ❖ реже C++
 - ❖ иногда 1C

The screenshot displays a web-based interface for managing a problem or service request. The top section contains various input fields for contact information (Last, Email, Phone, Ext, Address) and asset details (Number, Account, Address, City, State). A 'Type' dropdown is set to 'Maintenance Requi', 'Status' is 'Open', and 'Severity' is 'Low'. Below this, there are tabs for 'Workbench', 'Interactions', 'Related Objects', 'Service History', and 'Work Orders'. The 'Service History' tab is active, showing a 'Problem Summary' with a yellow highlight, 'Urgency' set to 'Respond - 24', and 'Error Code'. The 'Resolution Summary' section includes 'Resolution Code', 'Respond By' (03-AUG-2011 18:0), and 'Resolve By' (17-AUG-2011 14:0). A 'Notes' section on the right allows for viewing and printing notes, with a date range from 01-FEB-2011 00:00 to 31-JUL-2011 23:59. At the bottom, there is a table with columns for 'Outcome', 'Title', 'Type', 'Number', and 'Visibility', with a single row containing the number '60'.

Веб-разработка

- Создание веб-сайтов, от простых до сложных, интернет-магазины, функциональные веб-приложения.
- Языки и базы данных:
 - HTML, CSS, JS
 - **PHP + MySQL**
 - Ruby
 - MS ASP .Net



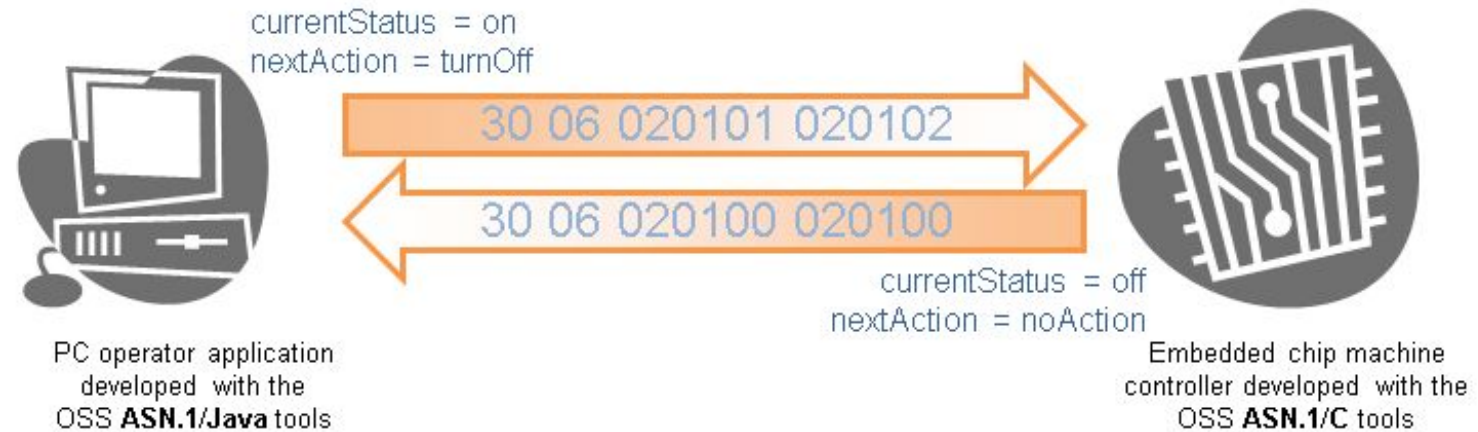
Мобильная разработка

- Придется кодить для смартфонов и планшетов.
- Программы самые разнообразные, от простых игр до функциональных приложений для корпораций.
- Платформы и языки:
 - ❖ **Android:** Java
 - ❖ **Apple:** Objective C и Swift
 - ❖ **Windows Phone:** .Net



Системное программирование

- все что связано с системным ПО, системами управления.
- Разрабатывать драйверы, утилиты, firmware, встроенное ПО.
- Знать устройство железа, интерфейсов, операционных систем, сетей.
- Языки: C/C++, Assembler



Резюме: рассмотренные вопросы

- На какие поколения делятся языки программирования?
- Как шло развитие языков программирования?
- В чем отличие языков высокого уровня от языков низкого уровня?
- Какие существуют парадигмы программирования?
- В чем заключается концепция структурной парадигмы?
- Какими понятиями и механизмами оперирует объектно-ориентированная парадигма?
- В каких ключевых сферах разработки какие языки программирования применяются сегодня?