





**IME-USP**

Linguagens interpretadas  
vs.  
Linguagens compiladas

Fabio Kon  
IME-USP



# Linguagens de Programação

- meados da década de 1940
  - Linguagem de Máquina
    - 0100 0011 1010 1111 0000 1010 0001 0011
- 1948
  - Linguagem de Montagem (Assembly)
    - ADD AX, BX
- Linguagens de Alto Nível
  - FORTRAN (1957)
  - LISP (1958)
  - Simula-67, Smaltalk-80
  - C, C++, Objective C
  - Java, Python, Ruby, Javascript, Clojure, Go, Hack, Haskell ...

# Linguagens Interpretadas



# Linguagens Compiladas



# Linguagens Híbridas (compiladas e interpretadas)

- Smalltalk, Java, Python
  - 1) Antes de iniciar a execução do programa, um compilador traduz o código-fonte para *bytecode* (código em bytes).
  - 2) Ao iniciar a execução do programa, o interpretador lê os *bytecodes* um-a-um executando os comandos correspondentes.

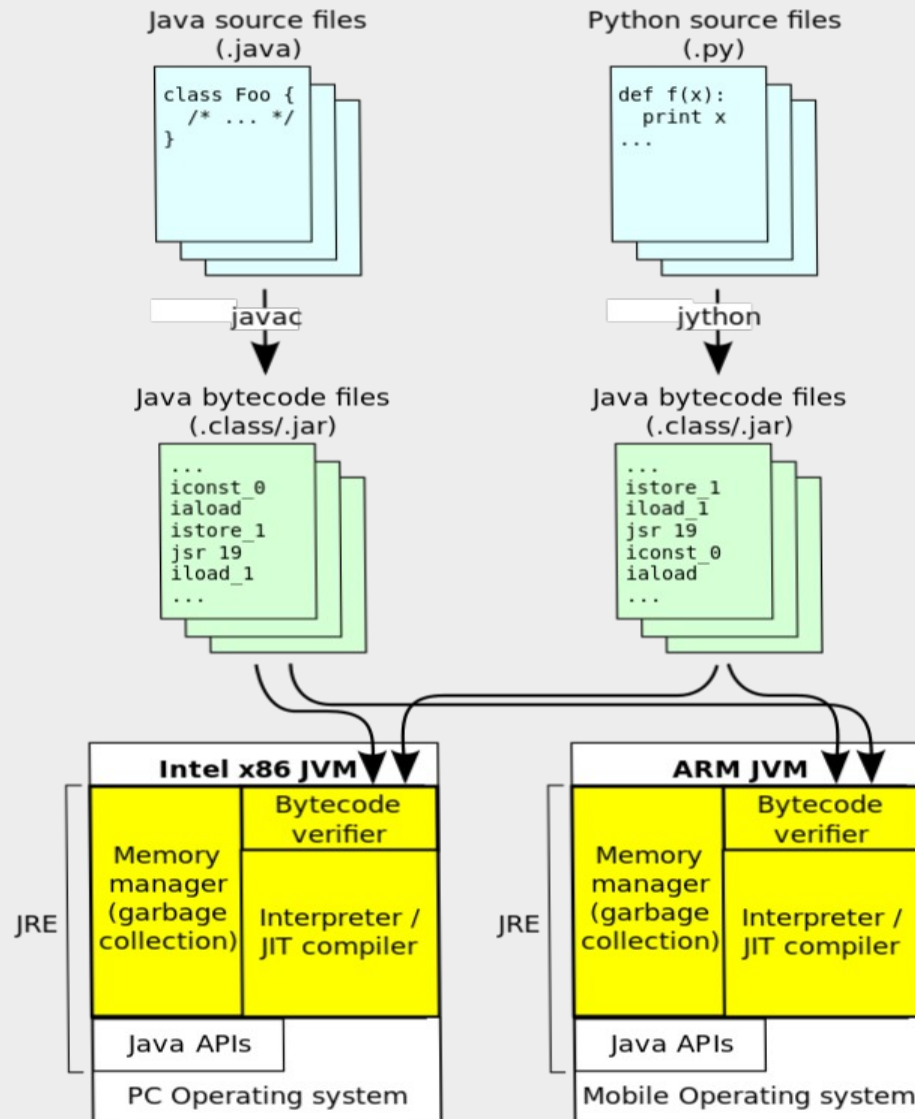
# A filosofia de Java

Java nasceu em 1995 com a seguinte ideia:

- *Write once, run everywhere*
  - diferentes SOs
  - diferentes tipos de computadores
- Uma linguagem para a Internet
- Como fazer isso?
  - usar uma ideia de Smalltalk: VM



# Máquina Virtual Java (JVM)

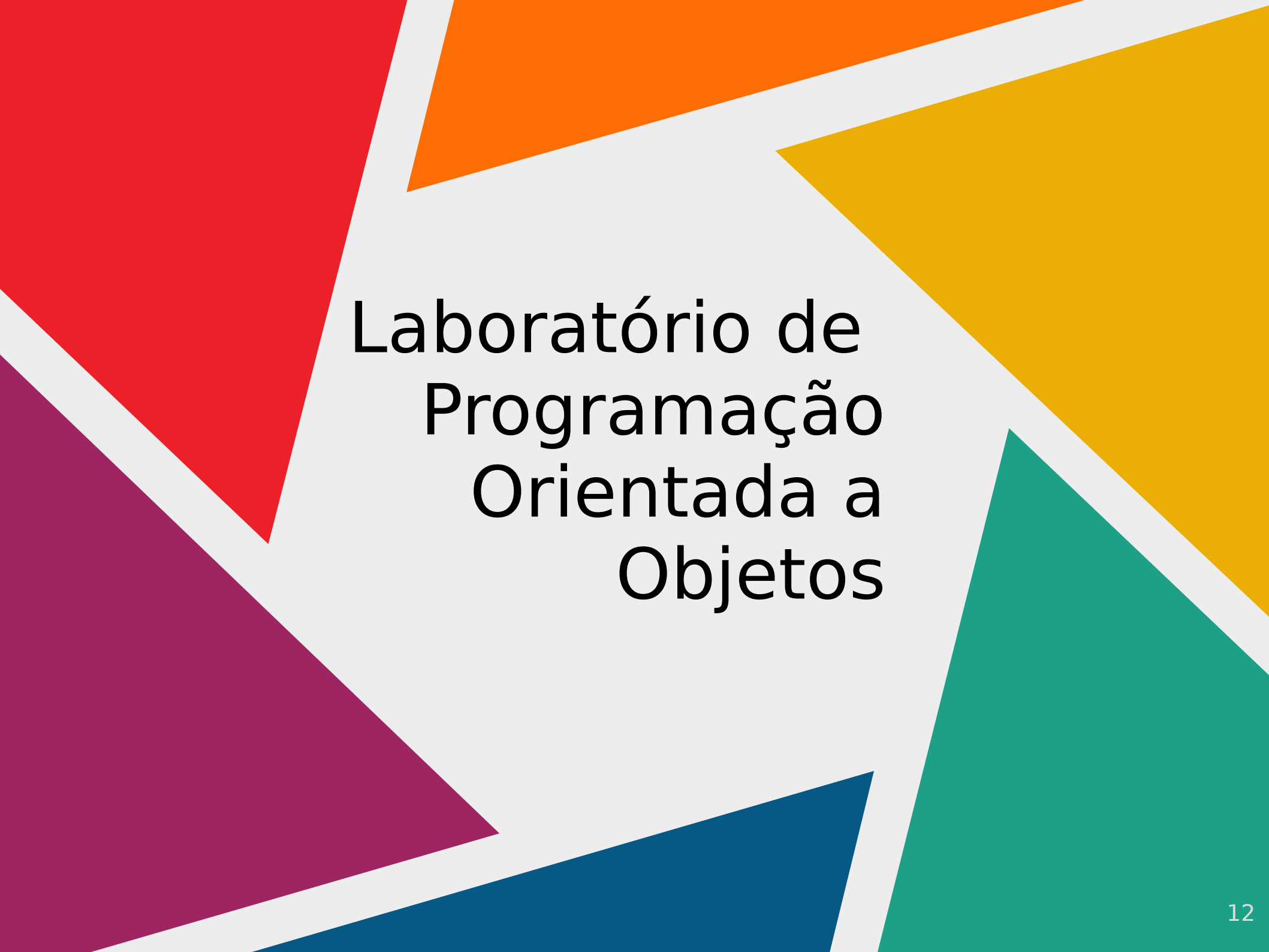


# Programação de bytecode

- Praticamente nunca vamos programar em bytecode na mão
- Para alguns usos avançados, existem *manipuladores de bytecode*

# Resumindo

- Linguagens podem ser:
  - compiladas
  - interpretadas
  - híbridas
- Boa parte das linguagens mais recentes são híbridas



# Laboratório de Programação Orientada a Objetos



**IME-USP**

