

Linguagem de Programação Java - Introdução

Jose.wellington@uniceub.br

Calendário

◀ agosto de 2013 ▶

D	S	T	Q	Q	S	S
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

◀ setembro de 2013 ▶

D	S	T	Q	Q	S	S
25	26	27	28	29	30	31
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5

◀ outubro de 2013 ▶

D	S	T	Q	Q	S	S
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2

◀ novembro de 2013 ▶

D	S	T	Q	Q	S	S
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

◀ dezembro de 2013 ▶

D	S	T	Q	Q	S	S
24	25	26	27	28	29	30
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

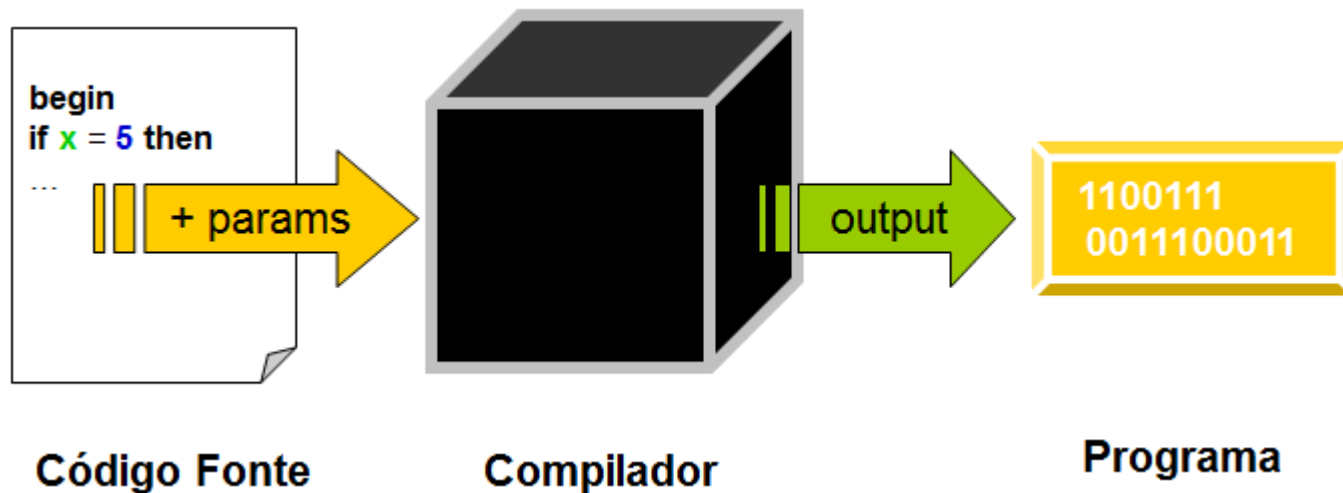
Agenda

- **Interpretador**
- **Compilador x Interpretadores**
- **Máquina Virtual – Java**
- **Ambiente de Trabalho**
- **Meu Primeiro Programa**
- **Exercício**

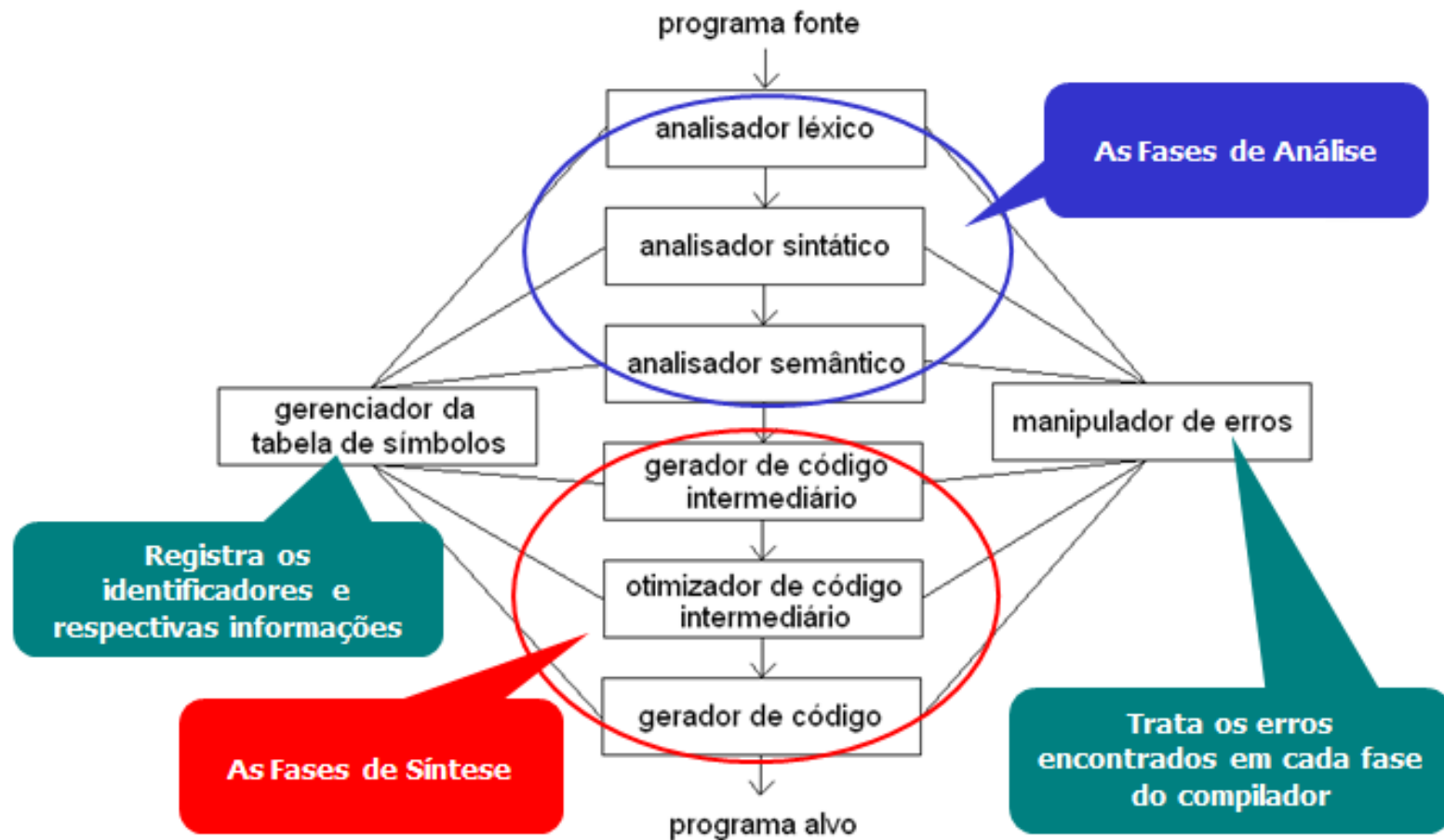
Compilador

Compilador

- Transforma Linguagem de alto nível em linguagem de máquina.



Compilador



Interpretador

Compilador x Interpretadores

■ Compiladores

- Tempo de espera pela compilação
- Execução dos programas com máxima performance, em linguagem de máquina

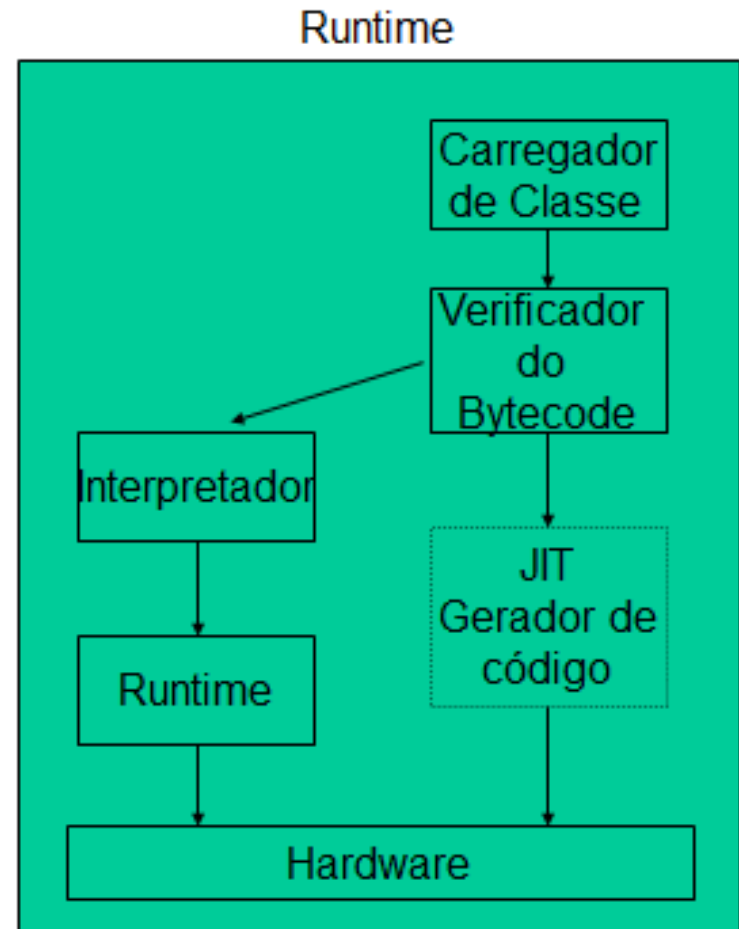
■ Interpretadores

- Resultados imediatos
- Execução (de linguagem de alto nível) até 100 vezes mais lenta que a versão compilada

Compilador x Interpretadores

■ Interpretadores

- Um programa P expresso em uma linguagem S, usando um interpretador de S executando na máquina M?



Compilador x Interpretadores

■ Interpretadores

- O funcionamento dos interpretadores é muito parecido ao dos compiladores.
- O interpretador traduz o código linha a linha.
- O código fonte não é totalmente traduzido antes de ser executado.
- Não existem fases distintas nem se produz código intermediário.
- Passa o tempo todo lendo, traduzindo código e executando.

Compilador x Interpretadores

	Vantagens	Desvantagens
Compiladores	Execução mais rápida	Várias etapas de tradução
	Permite estruturas de programação mais completas	Programação final é maior, necessitando mais memória para a sua execução
	Permite a otimização do código fonte	Processo de correção de erros e depuração é mais demorado
Interpretadores	Depuração do programa é mais simples	Execução do programa é mais lenta
	Consome menos memória	Estruturas de dados demasiado simples
	Resultado imediato do programa ou rotina desenvolvida	Necessário fornecer o programa fonte ao utilizador

Máquina Virtual - Java

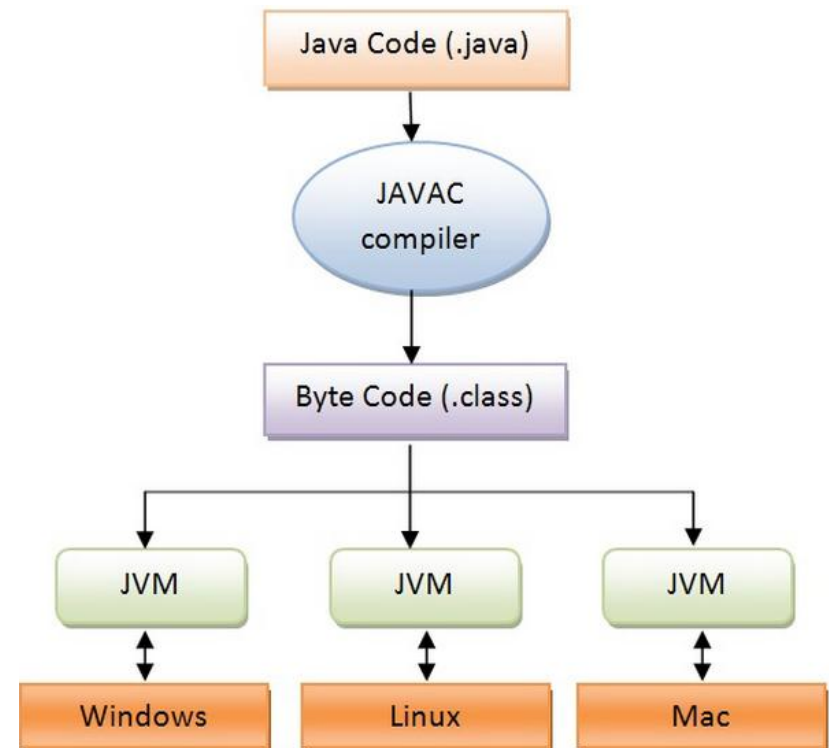
Máquina virtual Java

- É um programa que carrega e executa os aplicativos Java, convertendo os bytecodes em código executável de máquina.
- A JVM é responsável pelo gerenciamento dos aplicativos, à medida que são executados.



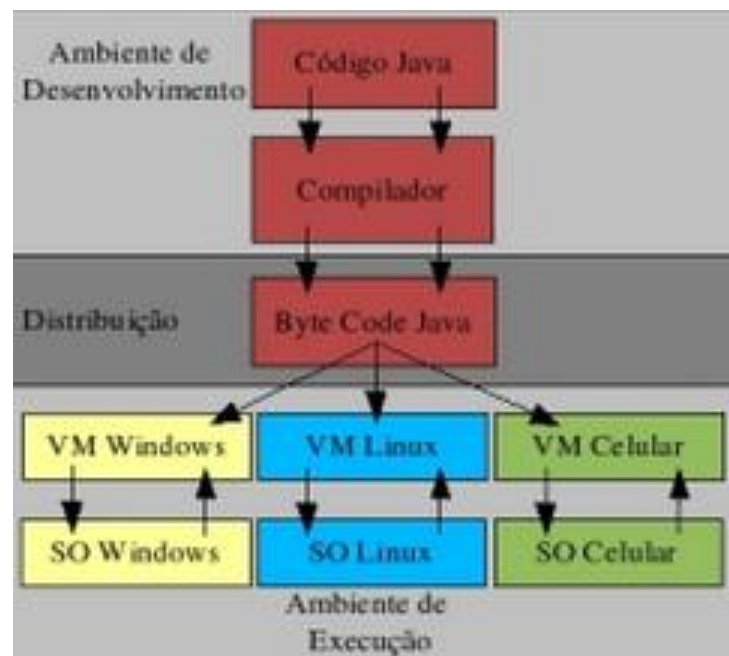
Máquina virtual Java

- A JVM é, em poucas palavras, um mecanismo que permite executar código em Java em qualquer plataforma.



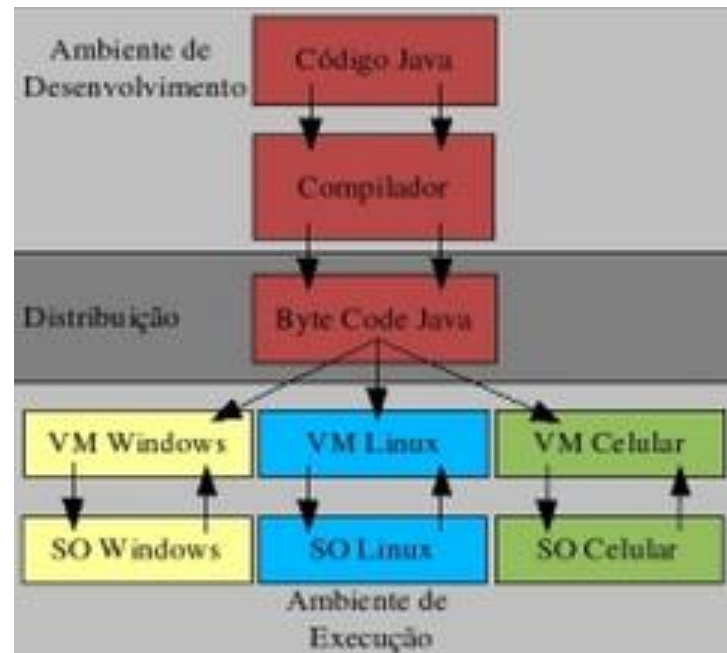
Máquina virtual Java

- **Bytecode** (traduzindo ao pé da letra, **código em bytes** — não confundir com código-máquina) é o resultado de um processo semelhante ao dos compiladores de código-fonte que não é imediatamente executável.



Máquina virtual Java

- O **bytecode** é um estágio intermediário entre o código-fonte (escrito numa linguagem de programação específica) e a aplicação final, sendo a sua vantagem principal a dualidade entre a portabilidade — o *bytecode* irá produzir o mesmo resultado em qualquer arquitetura — e a ausência da necessidade do pré-processamento típico dos compiladores.



Ambiente de Trabalho

O ambiente de desenvolvimento JDK e Netbeans

- Netbeans 7.2 ou superior, em <http://www.oracle.com>
- Instalar o JDK apropriado para o seu ambiente. A última versão homologada encontra-se em:
 - ⇒ <http://www.java.com>
 - ⇒ <http://www.oracle.com>
- Inicializar apropriadamente as variáveis de ambiente. Para a versão jdk1.7.0, segue:
 - ⇒ SET PATH=C:\JDK1.7.0\BIN; %PATH%
 - ⇒ SET CLASSPATH=c:\JDK1.7.0\JRE\LIB\RT.JAR;%CLASSPATH%;
 - ⇒ SET JAVA_HOME=C:\JDK1.7.0

Meu Primeiro Programa

```
// Primeiro Programa

class Olah {
    public static void main (String[] args) {
        System.out.println("Olah mundo");
    }
}
```

Meu Primeiro Programa

- Certifique-se de ter adicionado a sua lista de path's o path do compilador e interpretador Java, Javac e Java respectivamente.
- Crie o arquivo ao lado em um diretório qualquer e salve com o nome: `Olah.Java`
- Chame o compilador Java para este arquivo: `javac Olah.Java`
- Seu diretório deve ter recebido um novo arquivo após essa compilação: **`Olah.class`**
- Chame o interpretador Java para este arquivo (omite a extensão `.class` de arquivo): `java Olah`
- Observe o resultado na tela: `Olah mundo !!!`

Agenda

- **Interpretador**
- **Compilador x Interpretadores**
- **Máquina Virtual – Java**
- **Ambiente de Trabalho**
- **Meu Primeiro Programa**
- **Exercício**