

Estrutura de Dados

Apresentação da Disciplina

Prof. Paulo de Tarso F. Júnior

1

Quem sou?

- Ex-acadêmico da UFCG
- Mestre em Visão Computacional - UFCG
- Analista de Sistemas Sênior na Accenture
- Tutor à Distância – Introdução à Informática
- Desenvolvedor nas plataformas Java e .Net

Disciplina

- Ementa
 - Estrutura de dados primitivas:
 - Filas, Pilhas, Listas, Árvores e Mapas
 - Algoritmos de manipulação de estrutura de dados:
 - Inserção, eliminação, busca e percurso
 - Algoritmos de ordenação de dados
 - Quick sort, Bubble sort, Merge sort e Heap sort
 - Algoritmos de pesquisa
 - Busca binária e busca linear

Metodologia

- Utilização da linguagem Java
- Aulas práticas
- Listas de exercícios
- Discussão do conteúdo
- Avaliação por Estágio
- Nota
 - Avaliação por Estágio (Peso de 50%)
 - Trabalhos e listas de exercícios (peso de 35%)
 - Participação e envolvimento em sala (peso 15%)

Tecnologias

- JAVA SE 6 (Standard Edition)
- Eclipse 3.7.2

O que é JAVA

- Linguagem de programação
- Uma plataforma:

Linguagem JAVA

- Linguagem de alto nível,
- Orientada a objetos.
- Programas são *compilados e interpretados*.
- Linguagem intermediária chamada *bytecode*
- Java Virtual Machine (JVM)
- programas Java executados em qualquer sistema que tenha uma versão da JVM.

A plataforma Java

- É uma plataforma apenas de software que é executada sobre várias plataformas de hardware. Ela possui três versões:
 - Java SE (Standard Edition)
 - Java EE (Enterprise Edition)
 - Java ME (Micro Edition)

JAVA SE (Standard Edition)

- O Java SE permite desenvolver e implantar aplicativos Java em desktops e servidores, bem como ambientes integrados e em tempo real.
- O Java SE inclui classes que suportam o desenvolvimento de serviços da Web para Java e fornece a base para Java Platform, Enterprise Edition (Java EE).

Java EE (Enterprise Edition)

- A versão corporativa o ajuda a desenvolver e implantar aplicativos Java do lado do servidor transportáveis, robustos, escaláveis e seguros.
- Construído com base em Java SE, o Java EE fornece APIs de serviços da Web, de modelo de componente, de gerenciamento e de comunicações para implementar:
 - Aplicativos SOA
 - Web 2.0 de classe corporativa.

Java ME (Micro Edition)

- O Java ME fornece um ambiente para aplicativos em execução em uma grande variedade de dispositivos móveis e integrados, como:
 - Telefones celulares
 - PDAs
 - TV set-top boxes
 - Impressoras.
- Inclui :
 - interfaces com o usuário flexíveis
 - um modelo de segurança robusto
 - uma grande variedade de protocolos de rede integrados
 - suporte extensivo para aplicativos em rede e off-line que podem ser transferidos por download dinamicamente.

Porque Java é importante?

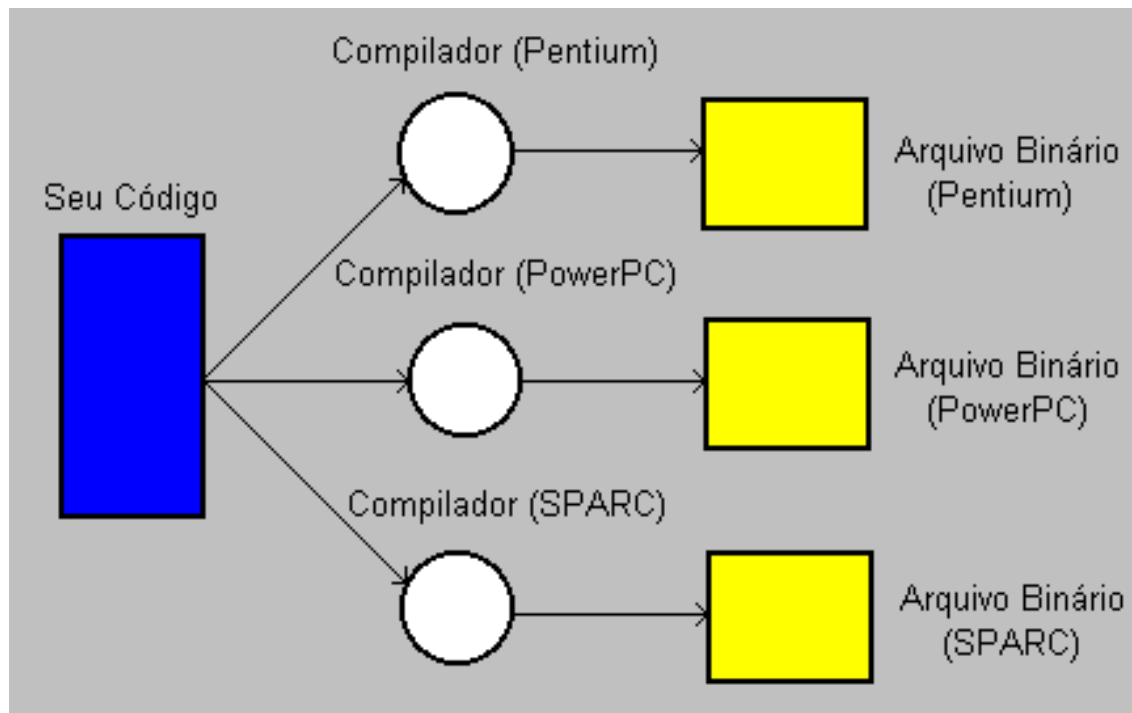
- Principais benefícios da linguagem Java:
 - Portabilidade de aplicativos Java em plataformas de hardware e sistemas operacionais — possível porque a JVM instalada em cada plataforma entende o mesmo bytecode.
 - A linguagem Java e a plataforma têm uma escala impressionante. Os aplicativos podem ser facilmente gravados (ou adaptados de aplicativos de desktop Java existentes) para dispositivos com recursos limitados. recursos da Web.
 - Projetada para execução de maneira segura em redes, a linguagem suporta este nível de segurança ao operar na Internet.

Máquina Virtual

- Em uma linguagem de programação como C e Pascal, temos o seguinte quadro quando vamos compilar um programa:
 - O código fonte é compilado para uma plataforma e sistema operacional específico. Muitas vezes, o próprio código fonte é desenvolvido visando uma única plataforma!
 - Esse código executável (binário) resultante será executado pelo sistema operacional e, por esse motivo, ele deve saber conversar com o sistema operacional em questão.

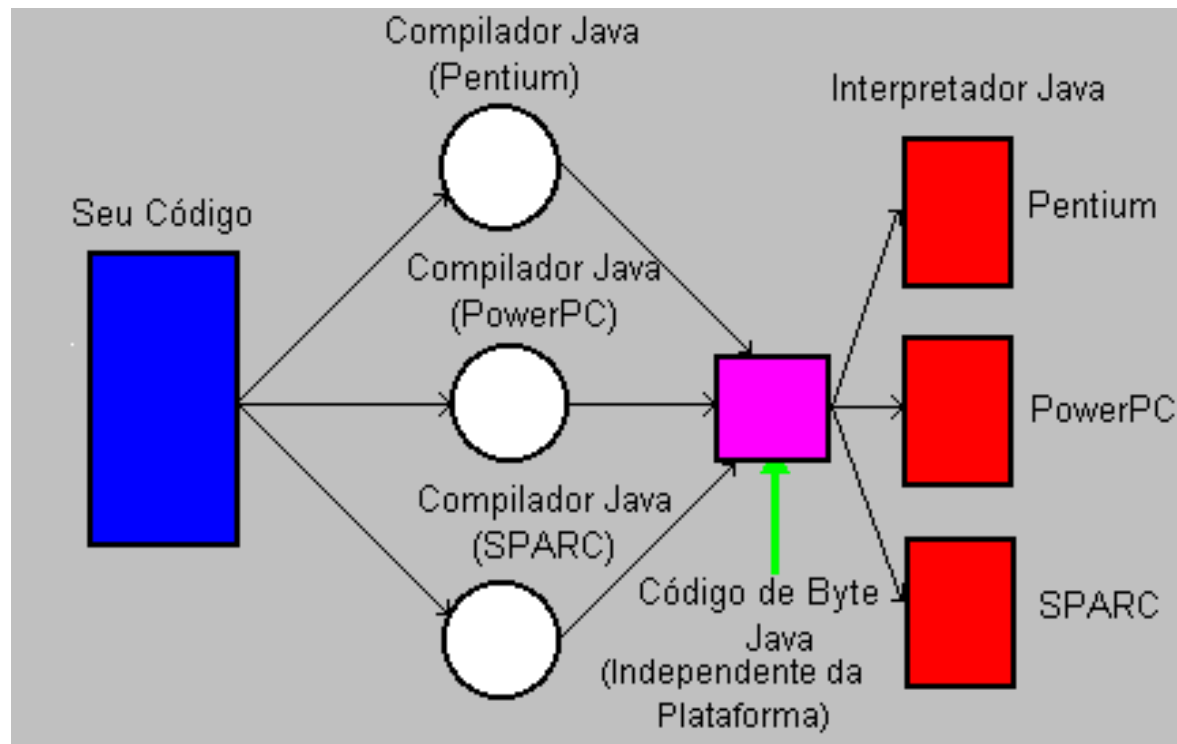
Máquina Virtual

- Linguagem C



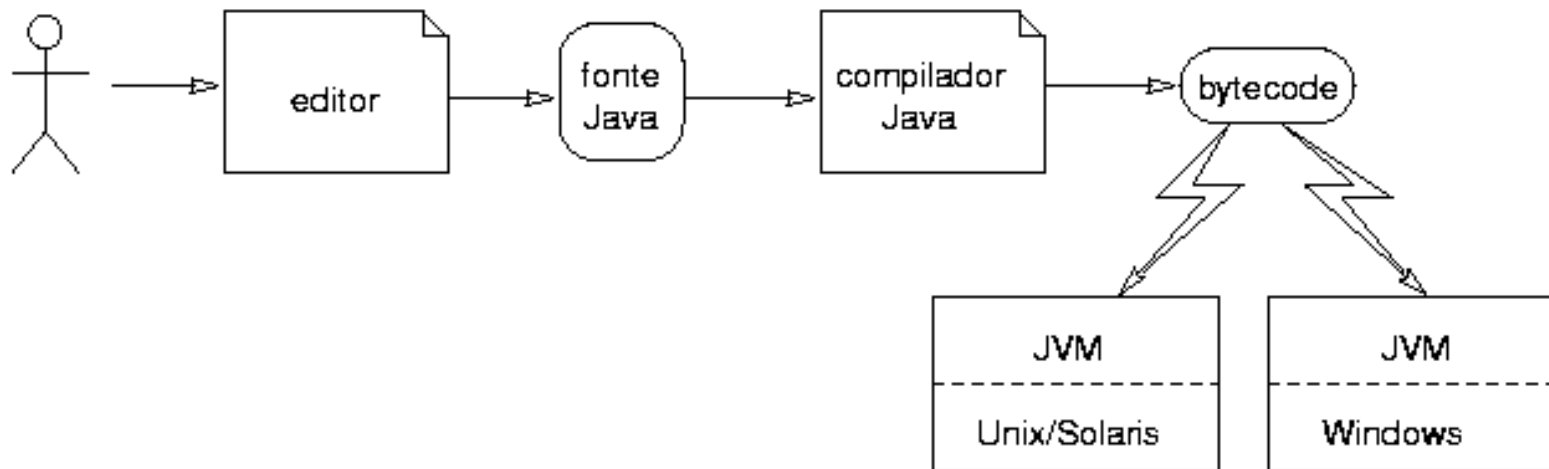
Máquina Virtual

- Linguagem JAVA



Máquina Virtual

- Linguagem JAVA



Estrutura de um programa JAVA

<code>/* Primeiro Programa em Java */</code>	comentários
<code>package pacote;</code>	definição do pacote
 <code>import java.util.Scanner;</code>	 biblioteca de E/S
 <code>Public class MeuPrograma {</code>	 definição da classe
 <code>// função principal inicio</code>	
<code>public static void main(String[] args)</code>	
{	marca início da função
 <code>// função para escrever na tela</code>	
<code>System.out.println ("olá Java!!");</code>	
 }	marca fim da função
}	

Eclipse

- O eclipse é uma IDE *Open Source* (código fonte aberto) que serve para o desenvolvimento em muitas linguagens:
 - PHP
 - Java
 - Python
 - HTML
 - C/C++
- A IDE Eclipse foi iniciado pela forte IBM, e depois foi doada para uma comunidade software livre.
- A grande vantagem do Eclipse é que seu desenvolvimento é todo baseado em plugins, isso nos ajuda muito na hora de utilizado para outras linguagens.

Variável

- Variável: “objeto” que pode assumir diversos valores
- Espaço de memória de um certo tipo de dado associado a um nome para referenciar seu conteúdo

```
public static void main(String[] args) {  
    int idade;  
    idade = 30;  
    System.out.println (" A idade é :"+idade);  
}
```

Nomes de variáveis

- Comece com letras ou sublinhado:
 - Seguidos de letras, números ou sublinhados
- Java é sensível ao caso:
 - `Peso` \neq `peso` \neq `pEso`
- Não podemos definir um identificador com o mesmo nome que uma palavra chave
 - `static` `public` `int` `long` `if` `while` `do`, dentre outras

Tipos básicos de dados (Primitivos)

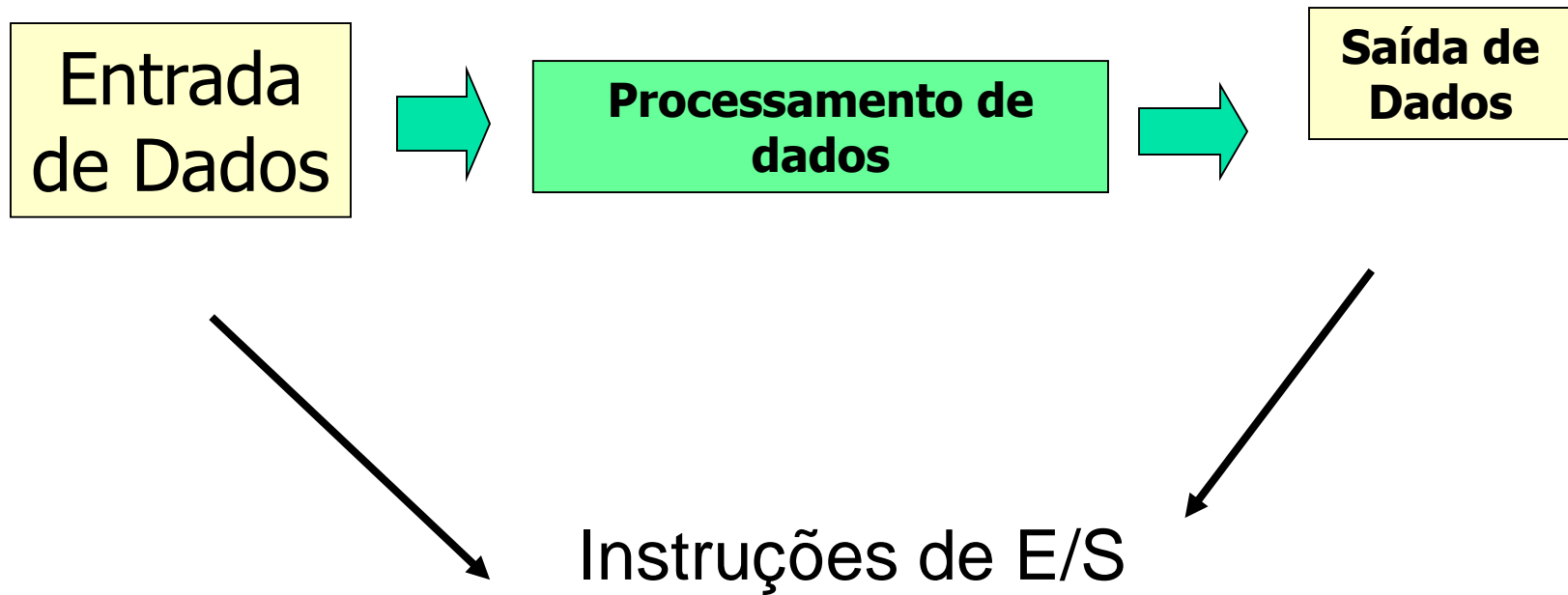
- Determina um conjunto de valores e as possíveis operações realizadas sobre os mesmos
- Informa a quantidade de memória (bits)

Tipo	Tamanho	Escala
byte	8 bits	-128 a 127
short	16 bits	-32.768 a 32.767
int	32 bits	-2.147.483.648 a 2.147.483.647
long	64 bits	-9.223.372.036.854.775.808 a 9.223.372.036.854.775.807
float	32 bits	-3.40292347E+38 a +3.40292347E+38
double	64 bits	-1.79769313486231570E+308 a +1.79769313486231570E+308
char	16 bits	'\u0000' a '\uFFFF'
boolean	1 bit	true ou false

Exemplo – Tipo de Dados

```
public static void main(String[] args) {  
  
    int soma=10;  
    float money= 2.21f;  
    char letra= 'A';  
    double num =2.01E6;  
    String nome = "Paulo"; //não é primitivo, é classe.  
  
    System.out.println("\n valor da soma = "+ soma);  
    System.out.println("\n Valor de Money = "+ money);  
    System.out.println("\n Valor de Letra = "+ letra);  
    System.out.println("\n Valor de Pi = "+ num);  
    System.out.println("\n Valor de nome = "+ nome );  
  
}
```

Comandos Básicos



Comandos E/S

- Apresentação de dados no monitor
- `System.out.println("string de controle");`

Exemplo:

```
Scanner leitor = new Scanner(System.in);
```

```
System.out.println("Digite a sua idade:\n");
```

```
int idade = leitor.nextInt();
```

```
System.out.println("Sua idade é: "+idade);
```

Operadores Aritméticos

Operador	Ação
+	Adição
*	Multiplicação
/	Divisão
%	Resto da divisão inteira
-	Subtração (unário)
--	Decremento
++	Incremento

Operadores Relacionais e Lógicos

Operador	Ação
>	Maior que
>=	Maior ou igual que
<	Menor que
<=	Menor ou igual que
==	Igual a
!=	Diferente de
&&	Condição “E”
	Condição “OU”
!	Não

Operadores de Atribuição

Operador	Exemplo	Ação
<code>+=</code>	<code>x += y</code>	<code>x = x + y</code>
<code>-=</code>	<code>x -= y</code>	<code>x = x - y</code>
<code>*=</code>	<code>x *= y</code>	<code>x = x * y</code>
<code>/=</code>	<code>x /= y</code>	<code>x = x / y</code>
<code>%=</code>	<code>x %= y</code>	<code>x = x % y</code>

Operadores de incremento e decremento

Operador	Exemplo	Ação
++	++a	adicionar 1 à variável a e depois calcular a expressão na qual a reside
	a++	calcular a expressão na qual a reside e depois adicionar 1 à variável a
--	--a	subtrair 1 da variável a e depois calcular a expressão na qual a reside
	a--	calcular a expressão na qual a reside e depois subtrair 1 da variável a

Comparações e testes

- Observemos antes de mais nada que `++x` é diferente de `x++`!

Se

```
x = 10;
```

```
y = ++x;
```

```
/* x=x+1; y=x; */
```

então

```
x = 11 e
```

```
y = 11
```

porém Se

```
x = 10;
```

```
y = x++;
```

```
/* y=x; x=x+1 */
```

então

```
x = 11 e
```

```
y = 10
```

Exercícios

- 1) Faça um programa para ler e escrever na tela o seu nome e a sua idade.
- 2) Faça um programa para ler e escrever na tela os seguintes dados:
 - Nome
 - Endereço
 - Telefone
 - Cidade
 - Cep

Exercícios (Cont.)

- 3) Entre com 3 números, verifique qual o maior dentre os valores informados e imprima na tela a frase “o maior número é X”
 - Onde X é a variável contendo o maior valor
- 4) Entre com 2 números e apresente-os de forma ordenada

Dúvidas



Referências

- GOODRICH, M. T.; TAMASSIA, R. ***Estruturas de Dados e Algoritmos Em Java***. 5ª Edição. Bookman, 2013.
- PUGA, S.; RISSETI, G. ***Lógica de Programação e Estruturas de Dados com Aplicações em Java***. 2ª Edição. Prentice Hall, 2012.
- Algoritmos e Estruturas de Dados com Java. Disponível em: <https://www.caelum.com.br/apostila-java-estrutura-dados/>. Acesso em: 16/05/2016.
- Introdução aos tipos de Estrutura de Dados em JAVA. Disponível em: <http://www.javaprogressivo.net/2012/09/introducao-aos-tipos-de-estrutura-de.html>. Acesso em: 22/05/2016.