



PROGRAMAÇÃO I

VETORES I

Prof. Dr. Daniel Caetano

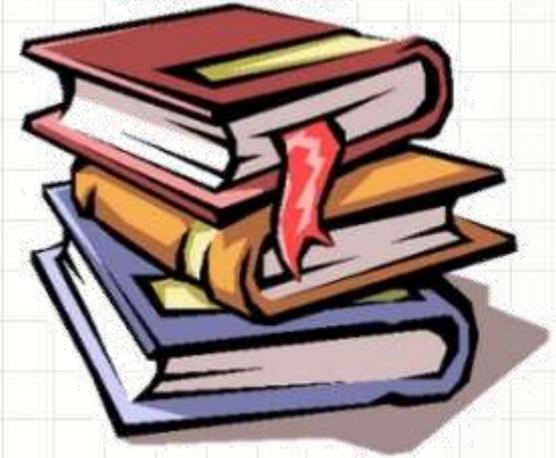
2017 - 1

Objetivos

- Compreender o conceito de vetores
- Conhecer os vetores em Java
- Conhecer o uso do “for” como “for-each”
- Compreender a String como um análogo aos vetores



Material de Estudo



Material

Acesso ao Material

Apresentação

<http://www.caetano.eng.br/>
(Programação I – Aula 6)

Material Didático

Programação I – Espalhando nas Págs 65 a 70



VETORES

Vetores

- O que é um vetor?
 - Estrutura que armazena sequência de dados
 - Analogia?

PlanilhasVoos_Avianca.xlsx - Microsoft Excel

Arquivo | Página Inicial | Inserir | Layout da Página | Fórmulas | Dados | Revisão | Exibição

L11

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Vc	Tip	Aerona	Orige	Destin	Dia	Hora	Dia	Hora					
2	6189	1	31807	BSB	GRU	1	09:11	1	10:50					
3	6308	1	31903	BSB	NAT	1	11:46	1	14:37					
4	6320	1	31805	BSB	SSA	1	11:48	1	14:07					
5	6374	1	32002	BSB	FOR	1	11:56	1	14:39					
6	6300	1	31904	BSB	REC	1	12:02	1	14:36					
7	6306	1	31804	BSB	MCZ	1	12:28	1	14:53					
8	6171	1	31805	BSB	CGH	1	18:26	1	20:08					
9	6219	1	31804	BSB	GIG	1	18:54	1	20:22					
10	6221	1	31904	BSB	SDU	1	19:04	1	21:00					
11	6318	1	32002	BSB	GRU	1	19:12	1	21:00					
12	6173	1	31903	BSB	CGH	1	19:15	1	20:52					
13	6161	1	31807	BSB	FLN	1	20:35	1	22:36					

Pronto

Vetores

- Diferença pro Excel
 - Só podemos guardar um único tipo de dado
 - Temos qual o tamanho antecipadamente
- Há dois jeitos de criar vetores
 - Declaração direta: todos os valores conhecidos
 - Declaração postergada: não conheço os valores

Vetores – Declaração Direta

- Método mais simples
 - Preciso conhecer todos os valores antes

- Formato

```
tipo[] nome = { v1, v2, ..., vn };
```

- Exemplo:

```
int[] umVetor = { 1, 2, 4, 8 };
```

- Como isso fica na memória?

umVetor

Índice	0	1	2	3
Conteúdo	1	2	4	8

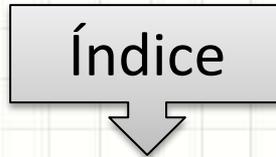
Vetores – Declaração Direta

- Como acessar os dados desse vetor?

umVetor

Índice	0	1	2	3
Conteúdo	1	2	4	8

- É fácil!



```
int x = umVetor[0];  
System.out.println(umVetor[2]);
```

- E se eu indicar um índice inexistente?
 - Vai dar um problemão!

Vetores – Declaração Postergada

- Método mais comum
 - Não conheço os dados na declaração!

- Formato

```
tipo[] nome = new tipo[tamanho];
```

- Exemplo:

```
int[] umVetor = new int[4];
```

- Como isso fica na memória?

umVetor

Índice	0	1	2	3
Conteúdo	?	?	?	?

Vetores – Declaração Postergada

- Como guardar dados nesse vetor?

umVetor

Índice	0	1	2	3
Conteúdo	?	?	?	?

- É fácil!

Índice
↓

```
umVetor[1] = 100;
```

Vetores – Declaração Postergada

- Como guardar dados nesse vetor?

umVetor

Índice	0	1	2	3
Conteúdo	?	100	?	?

- É fácil!

```
umVetor[1] = 100;
```

```
umVetor[3] = 55;
```

Vetores – Declaração Postergada

- Como guardar dados nesse vetor?

umVetor

Índice	0	1	2	3
Conteúdo	?	100	?	55

- É fácil!

```
umVetor[1] = 100;
```

```
umVetor[3] = 55;
```

- E se eu indicar um índice inexistente?
 - Vai dar um problemão!

Vetores – Declaração Postergada

- Como acessar os dados desse vetor?

umVetor

Índice	0	1	2	3
Conteúdo	?	100	?	55

- É fácil!

Índice

```
int x = umVetor[3];
```

```
System.out.println(umVetor[2]);
```

- E se eu indicar um índice inexistente?
 - Vai dar um problemão!

Vetores: tamanho

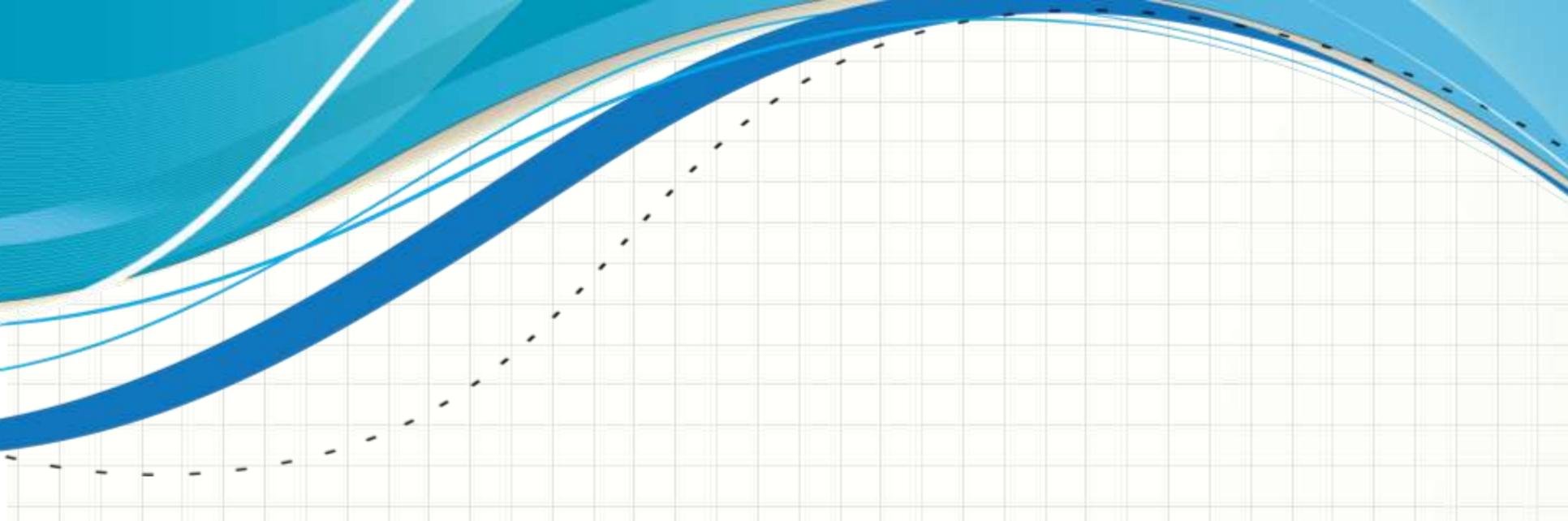
- Como vimos, é importante saber o tamanho!
 - “atributo” length!

```
int[] umVetor = { 1, 2, 4, 8 };  
int tamanho = umVetor.length;  
System.out.println(tamanho);
```

4

umVetor

Índice	0	1	2	3
Conteúdo	1	2	4	8



PERCORRENDO OS ELEMENTOS DE UM VETOR

Imprimindo um Vetor

- Existem várias formas...
 - Desde a manual (imprimir um por um)
 - Passando pelo for...

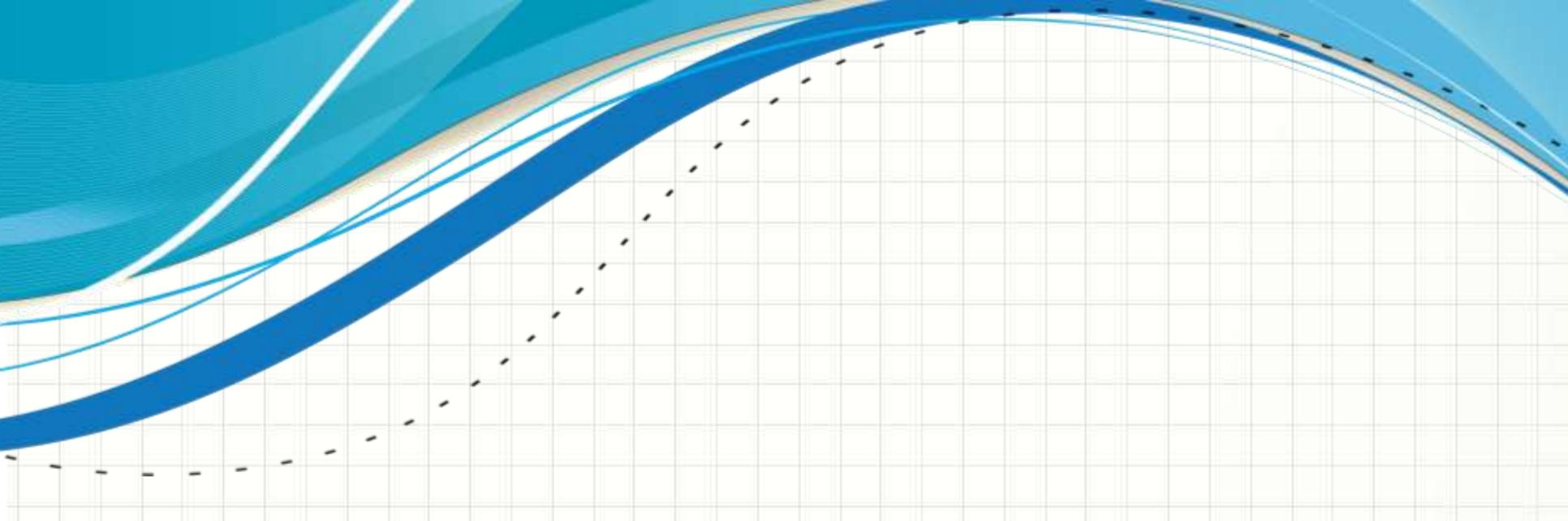
```
int[] v = { 1, 2, 4, 8 };  
for (int i=0; i < v.length; i=i+1) {  
    System.out.println(v[i]);  
}
```

Imprimindo um Vetor

- Existem várias formas...
 - “for each”

```
int[] v = { 1, 2, 4, 8 };  
for (int n : v) {  
    System.out.println(n);  
}
```

- Limitação?
 - Não serve para mudar o vetor (só imprimir)



COPIANDO UM VETOR

Copiando um Vetor

- Processo simples:

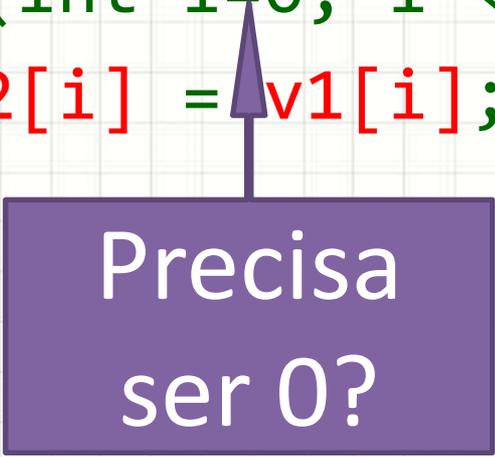
```
int[] v1 = { 1, 2, 4, 8 };  
int[] v2 = new int[4];
```

Precisa
ser 4?



```
for (int i=0; i < v1.length; i=i+1) {  
    v2[i] = v1[i];  
}
```

Precisa
ser 0?



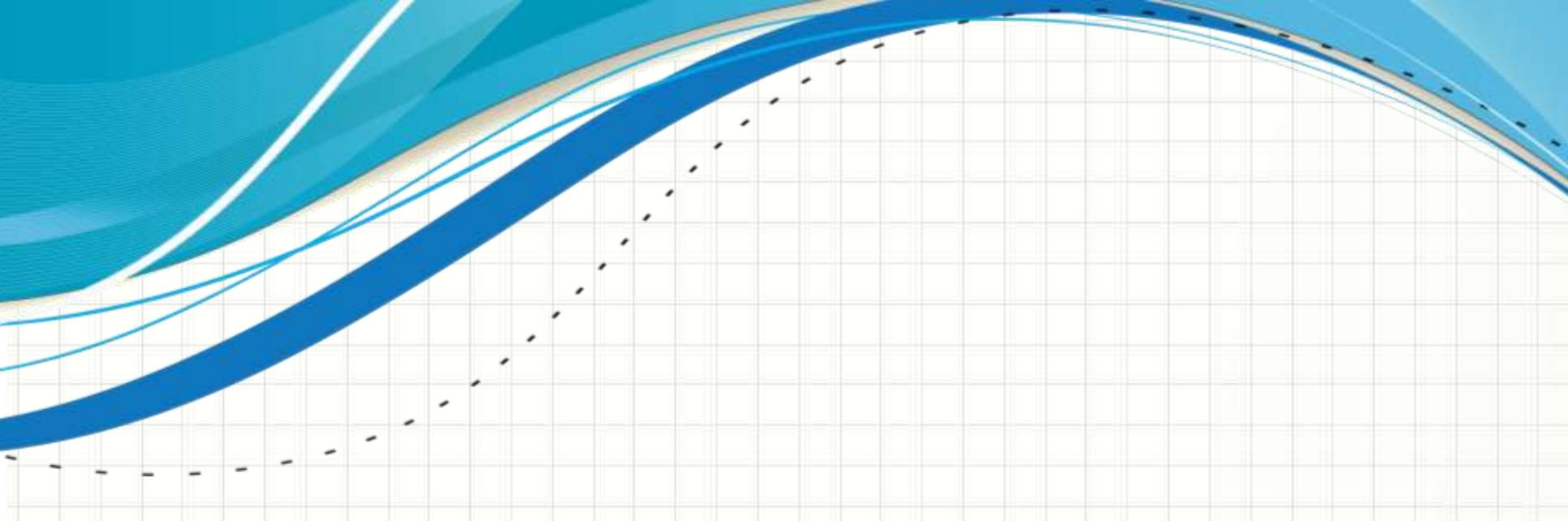
Copiando um Vetor

- Alternativa

```
int[] v1 = { 1, 2, 4, 8 };  
int[] v2;
```

```
v2 = Arrays.copyOf(v1, v1.length);
```

- Limitação
 - Sempre do início ao “comprimento”
 - Vetor destino sempre terá o tamanho indicado



PARALELO ENTRE STRING E VETOR

Métodos da Classe String

- `String tmp = "abacate";`

`tmp`

Índice	0	1	2	3	4	5	6
Conteúdo	a	b	a	c	a	t	e

- `tmp.length()`
- `tmp.charAt(pos)`
- Imagine imprimir letra por letra...!



MATRIZES

Matrizes

- Mais parecido com o Excel:
 - Linhas e colunas

	0	1	2	3
0				
1				
2				
3				

- Como criar uma matriz?

Matrizes – Criando

- Método mais simples
 - Preciso conhecer todos os valores antes
- Formato

```
tipo[][] nome = {  
    { u1, u2, ..., un },  
    { v1, v2, ..., vn },  
    ...  
    { w1, w2, ..., wn }  
};
```

Matrizes – Criando

- Método mais simples
 - Preciso conhecer todos os valores antes
- Exemplo:

```
int[][] matrix = {  
    {1, 2, 3}, {2, 4, 8} };
```

- Como isso fica na memória?

matrix

	0	1	2
0	1	2	3
1	2	4	8

Matrizes – Acessando

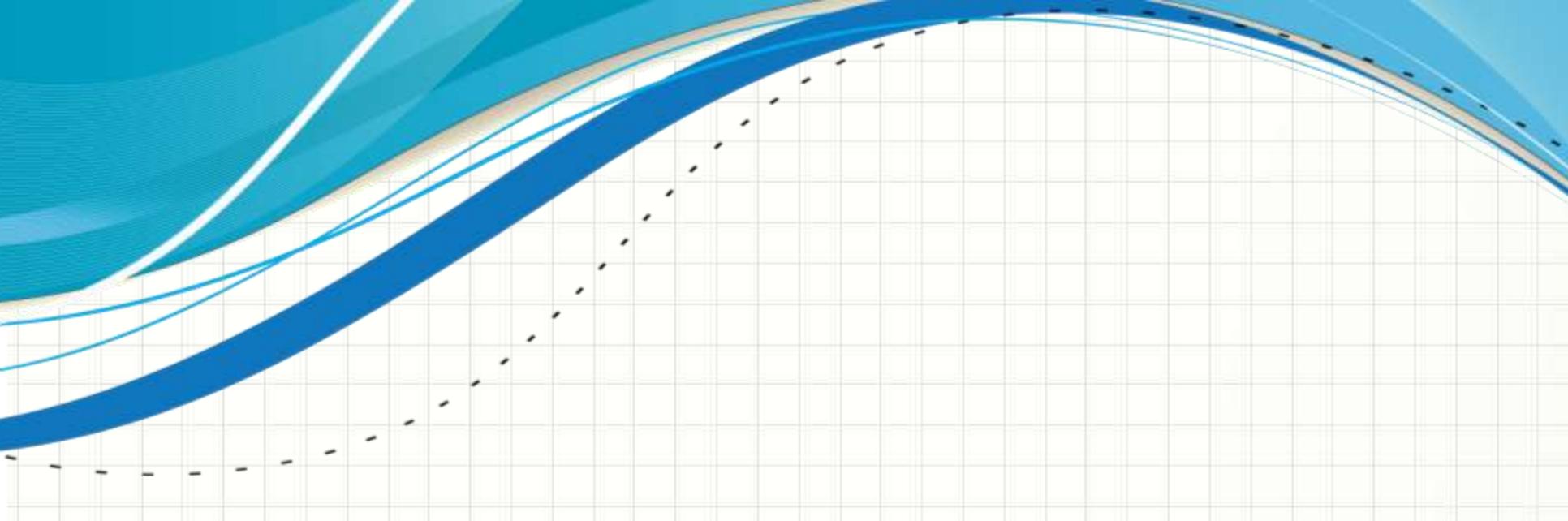
- Como acessar?

matrix

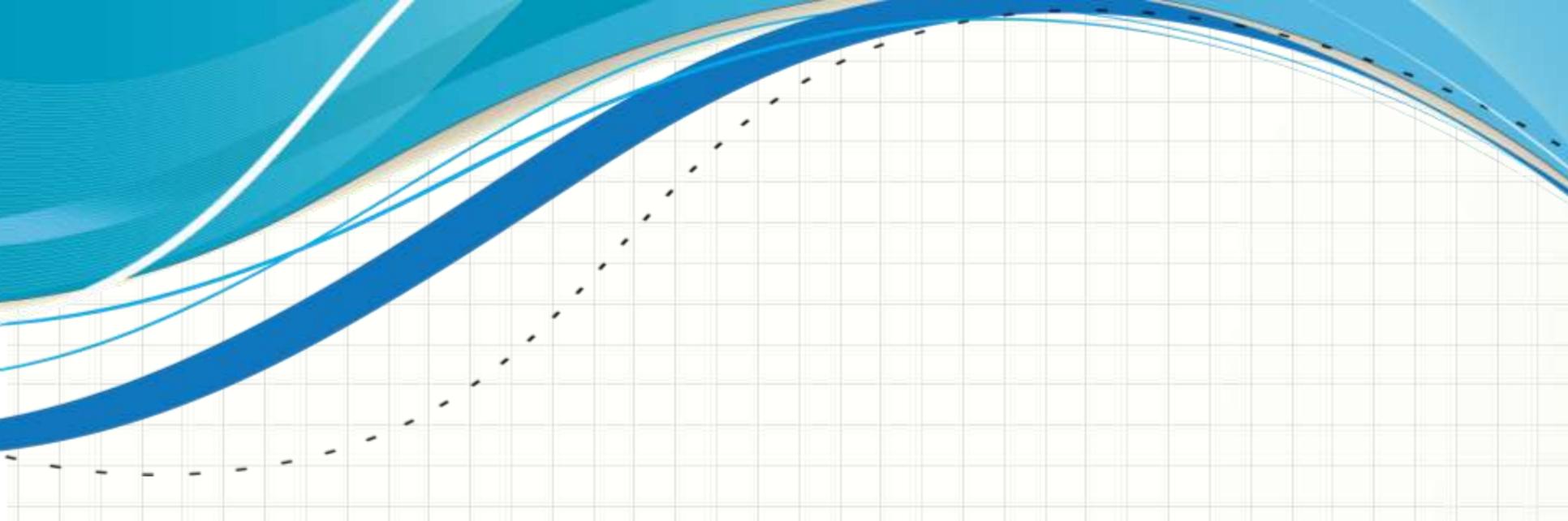
	0	1	2
0	1	2	3
1	2	4	8

- Fácil:

```
int x = matrix[1][2];  
System.out.println(matrix[1][2]);
```



PERGUNTAS?



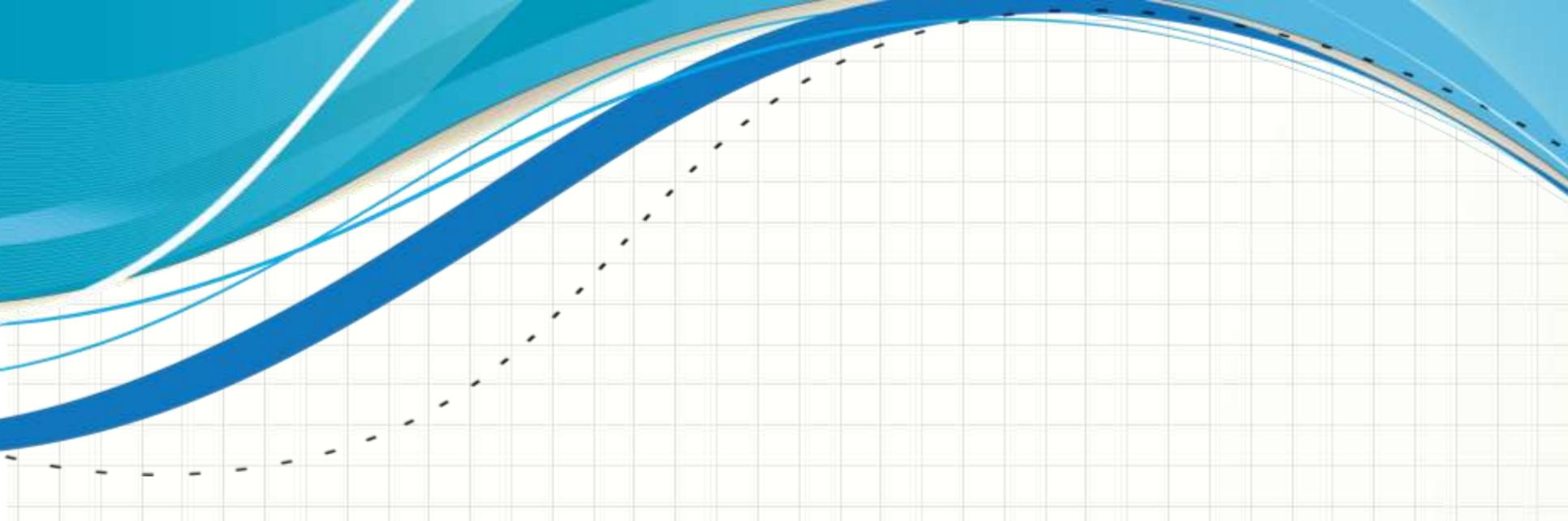
PARTE PRÁTICA

Experimentando

- Vamos construir um programa **Turma** que leia a nota dos alunos e calcule a média
- O programa deve imprimir as notas dos alunos que tiveram nota superior à média

Avançando no CadProd

- Vamos implementar o menu no CadProd.



PROJETO / ATIVIDADE

Projeto / Atividade

- Objetivo: criar cadastro de clientes simples
- Quem: Duplas especificadas
- A primeira entrega será 17/04
 - Classe cliente (com validações)
 - Criação/Preenchimento dos dados do cliente por modo texto (Scanner)
- A segunda entrega será 05/06
 - Listagem de clientes
 - Busca de clientes
 - Edição de clientes

Projeto / Atividade

- Hoje: Continue o projeto CadCli
 - Crie um método **cadastrarCliente()**
 - Cria um cliente “vazio”
 - Pede para o usuário digitar cada um dos valores
 - Usa o setter para preencher o atributo
 - Em caso de erro (não passou na validação), repete a pergunta
 - Modifique o método main da classe CadCli
 - Para que quando a opção “Cadastrar Cliente” for escolhida, o método cadastrarCliente() seja chamado.