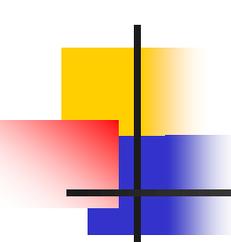


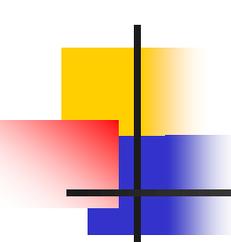
Arrays em Java

Alberto Costa Neto
DComp - UFS



Roteiro

- Arrays
- Arrays de Objetos
- Arrays Multidimensionais



Arrays

- Como armazenar as médias dos 3 alunos que se candidataram para fazer parte do grupo de pesquisa?

```
float media1 = 7.5f;
```

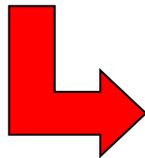
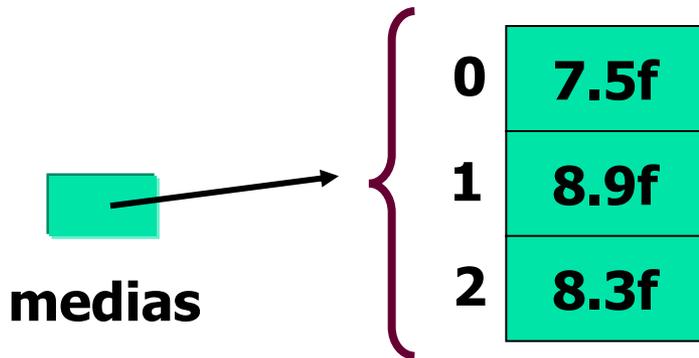
```
float media2 = 8.9f;
```

```
float media3 = 8.3f;
```

Não existiria uma estrutura de dados que armazenasse todos os dados, tendo apenas um nome como referência???

Arrays

- Algo assim...



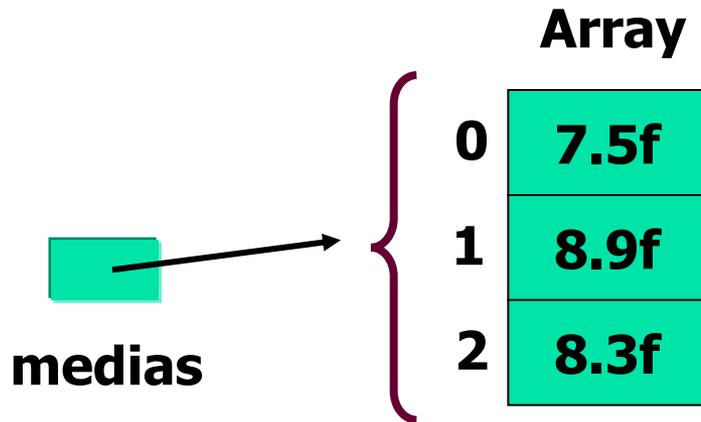
```
medias [0] = 7.5f;
```

```
medias [1] = 8.9f;
```

```
medias [2] = 8.3f;
```

Arrays

- O que é um array?



← Arrays são objetos (Java)

Sintaxe especial



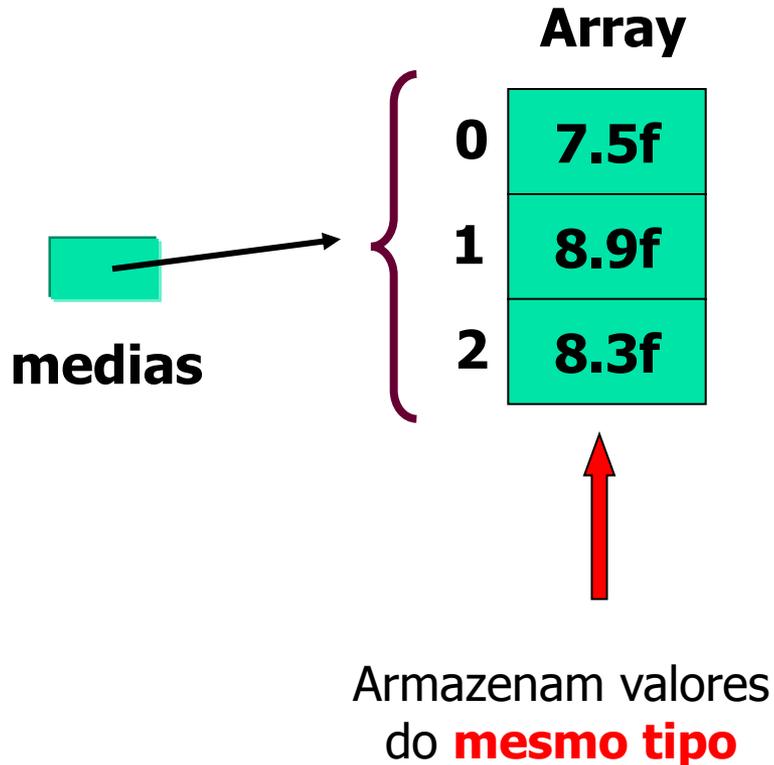
```
medias [0] = 7.5f;
```

```
medias [1] = 8.9f;
```

```
medias [2] = 8.3f;
```

Arrays

■ Características de um array



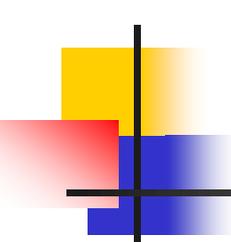
Elementos:

- **Identificados** pelo **mesmo nome**
- **Diferenciados** apenas pelo **índice**

medias [0] = 7.5f;

medias [1] = 8.9f;

medias [2] = 8.3f;



Arrays

- Como declarar um array?

```
float [ ] medias;
```

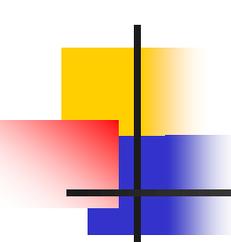
```
float medias [ ];
```

```
int [ ] matriculas;
```

```
int matriculas [ ];
```

```
String [ ] nomes;
```

```
String nomes [ ];
```



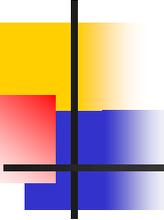
Arrays

- Como criar um array?
 - Usando "new"

```
String [ ] nomes;  
nomes = new String [3];  
  
float [ ] medias = new float [3];
```

- Usando inicializadores

```
String[ ] nomes = { "Maria", "Pedro", "João" };  
  
float [ ] medias = {7.5f , 8.9f , 8.3f } ;
```

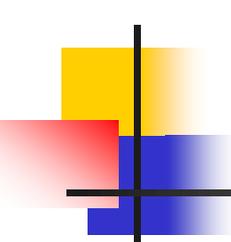


Arrays

- Atribuindo valores

```
String[ ] nomes = new String [3];  
nomes[0] = "Maria";  
nomes[1] = "Pedro";  
nomes[2] = "João" ;
```

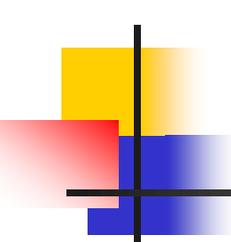
```
float [ ] medias = new float [3];  
medias [0] = 7.5f;  
medias [1] = 8.9f;  
medias [2] = 8.3f;
```



Arrays

- Outro exemplo

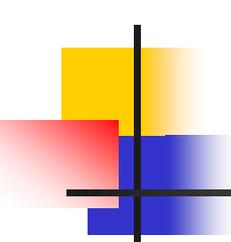
```
String[] nomes = new String [3];  
Scanner e = new Scanner(System.in);  
for (i = 0 ; i < 3; i++)  
    nomes[i] = e.next();
```



Arrays

- Acessando os valores armazenados

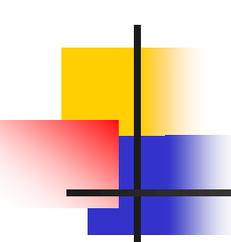
```
String[] nomes = new String [3];  
for (i=0 ; i < nomes.length ; i++)  
    System.out.println( nomes[i] );
```



Dever de Sala

1) Escreva um programa em Java que leia dez número inteiros e em seguida calcule e mostre:

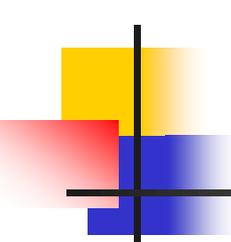
- a) Todos os números pares
- b) Todos os números ímpares
- c) A quantidade de números pares e ímpares



Dever de Sala

- 2) Escreva um programa em Java que armazena o nome de 5 alunos e de suas respectivas médias. O programa deve exibir:
- a) Nome do aluno com maior média (desconsiderar empates)
 - b) Nome dos alunos reprovados (média < 5)

Obs.: Usar 2 arrays (um p/ nomes e outro p/ médias)



Arrays de Objetos

- O que faz este código?

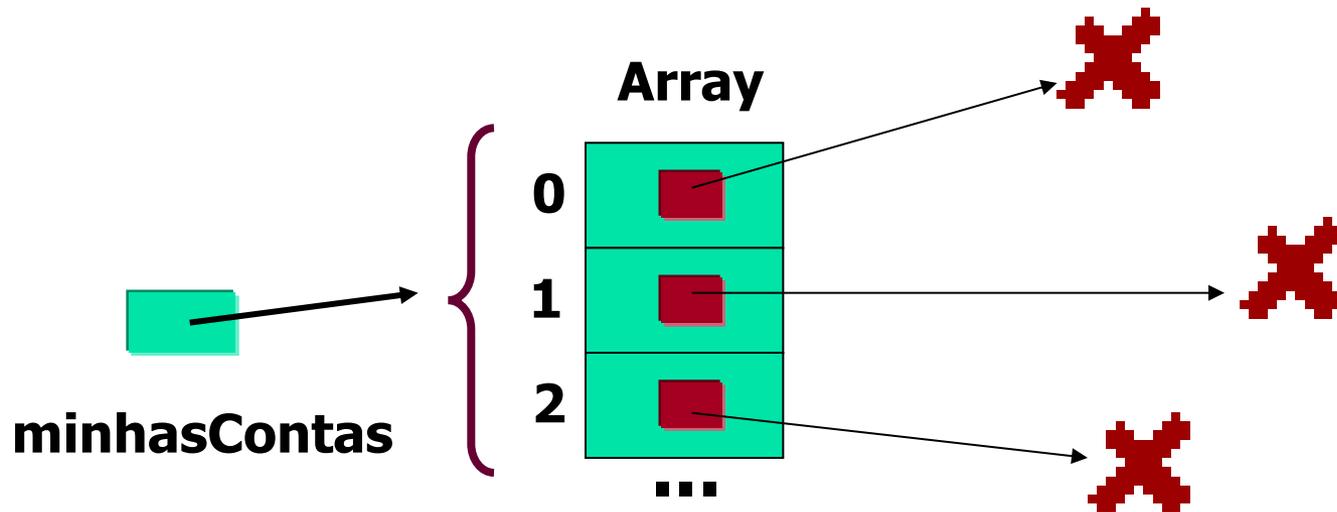
```
Conta[] minhasContas;  
minhasContas = new Conta[10];
```

- Quantas contas foram criadas?

Arrays de Objetos

- Foi criado um array para **guardar referências** para contas

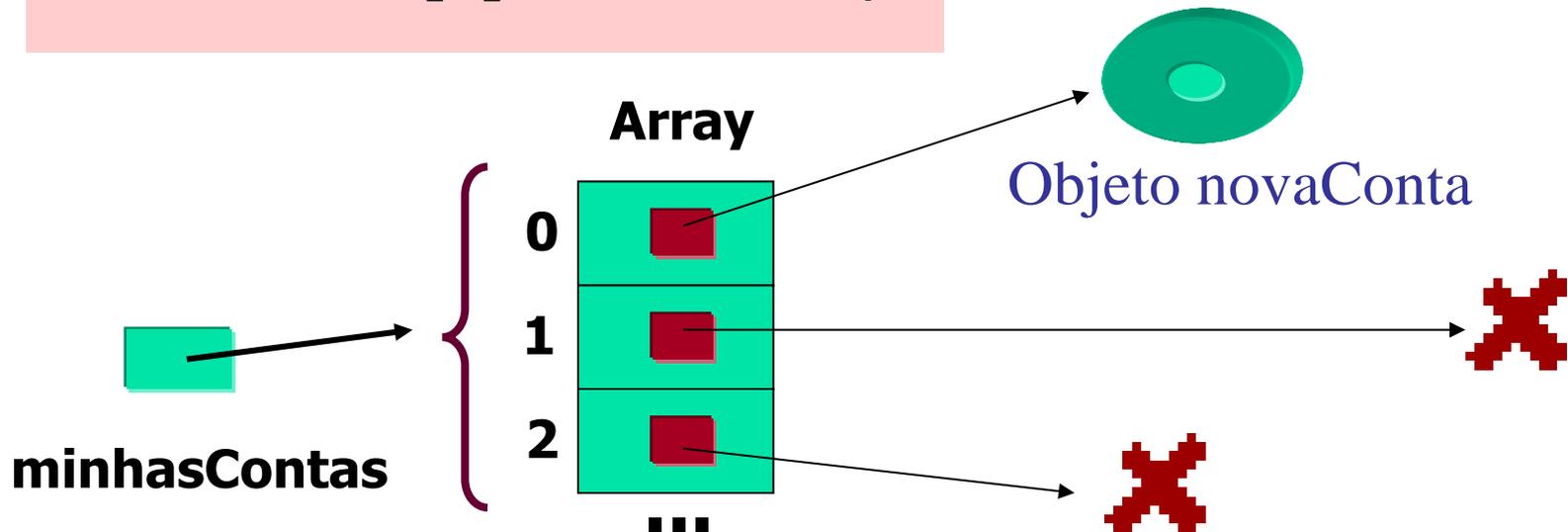
```
Conta[] minhasContas;  
minhasContas = new Conta[10];
```



Arrays de Objetos

- Populando o array

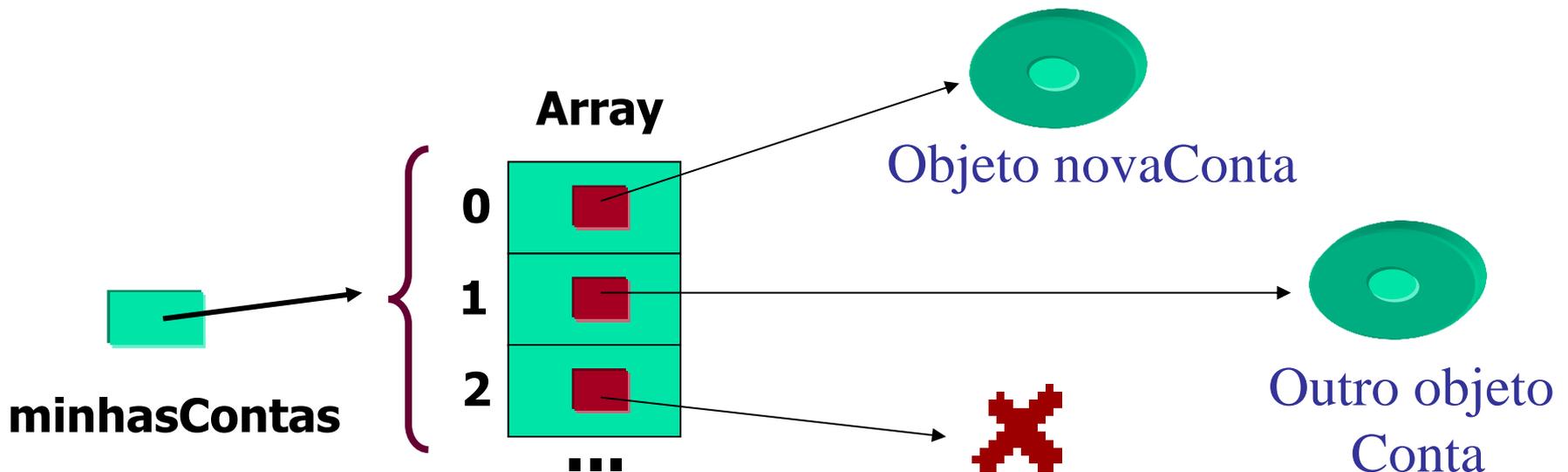
```
Conta[] minhasContas;  
minhasContas = new Conta[10];  
Conta novaConta = new Conta();  
novaConta.creditar(1000f);  
minhasContas[0] = novaConta;
```



Arrays de Objetos

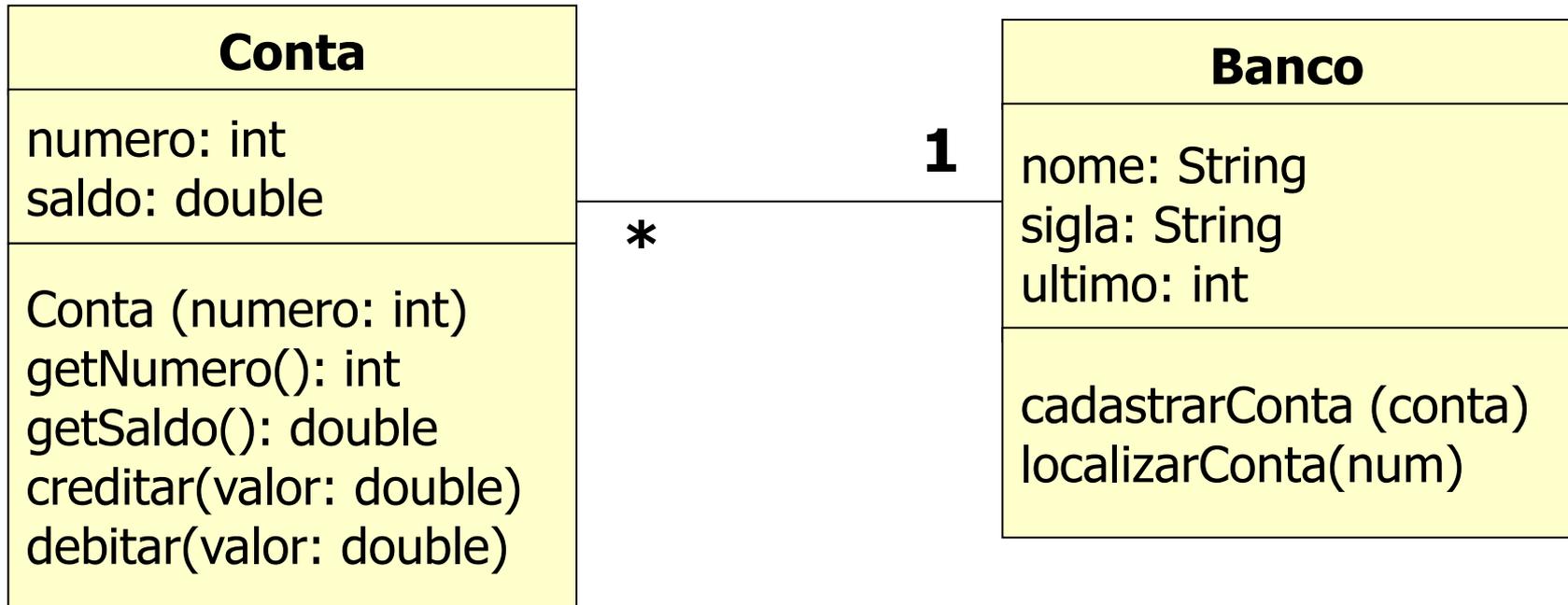
- Outra forma

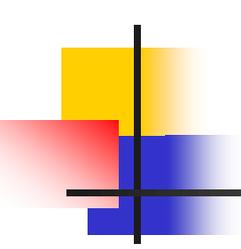
```
minhasContas[1] = new Conta();  
minhasContas[1].creditar(500f);
```



Arrays de Objetos

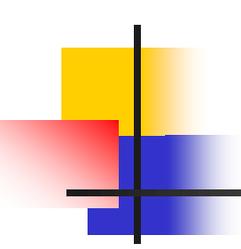
- Expandindo...





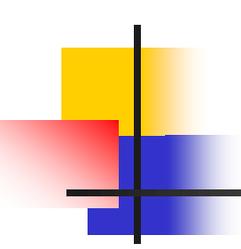
Arrays de Objetos

```
class Banco {  
    ...  
    private Conta[] contas;  
    private int ultimo = 0;  
  
    public Banco( ) {  
        contas = new Conta[100];  
    }  
  
    public void cadastrar(Conta c) {  
        contas[ultimo] = c;  
        ultimo = ultimo + 1;  
    }  
}
```



Arrays de Objetos

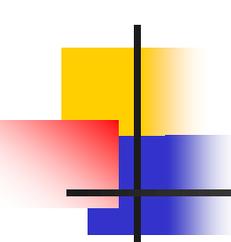
```
private Conta localizarConta (int n) {  
    for (int i = 0; i < ultimo; i++)  
        if (contas[i].getNumero() == n)  
            return conta[i];  
    }  
    return null;  
}  
...  
...
```



Arrays de Objetos

- Cadastro flexível

```
public void cadastrar(Conta c) {  
    if (ultimo==contas.length) {  
        Conta[] aux = new Conta[ultimo+10];  
        for (int i = 0; i < contas.length ; i++)  
            aux[i]= contas[i];  
        contas = aux;                // muda a referência  
    }  
    contas[ultimo] = c;  
    ultimo = ultimo + 1;  
}
```



Dever de Sala

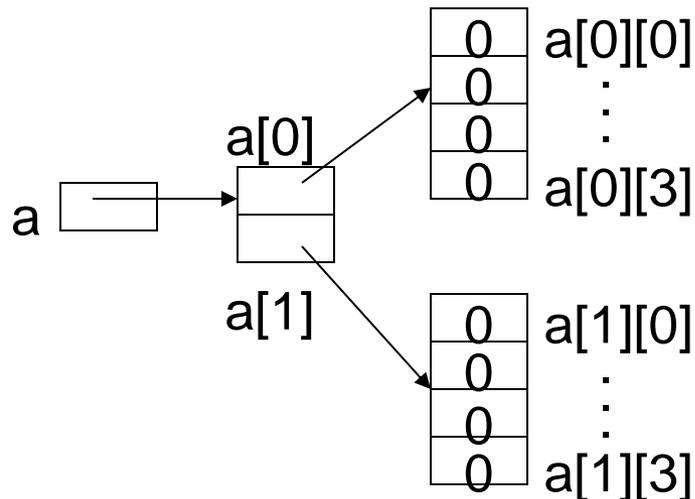
- 3) Escreva um programa em Java que armazena o nome de 5 alunos e de suas respectivas médias. O programa deve exibir:
- a) Nome do aluno com maior média (desconsiderar empates)
 - b) Nome dos alunos reprovados (média < 5)

Obs.: Usar 1 array contendo as referências para objetos Aluno

Arrays Multidimensionais

- O que faz este código?

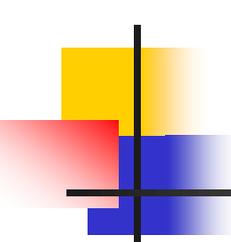
```
byte[ ][ ] a = new byte[2][4];
```



O tipo da variável 'a' é byte[][]

O tipo de a[0] e a[1] é byte[]

O tipo dos a[i][j] é byte



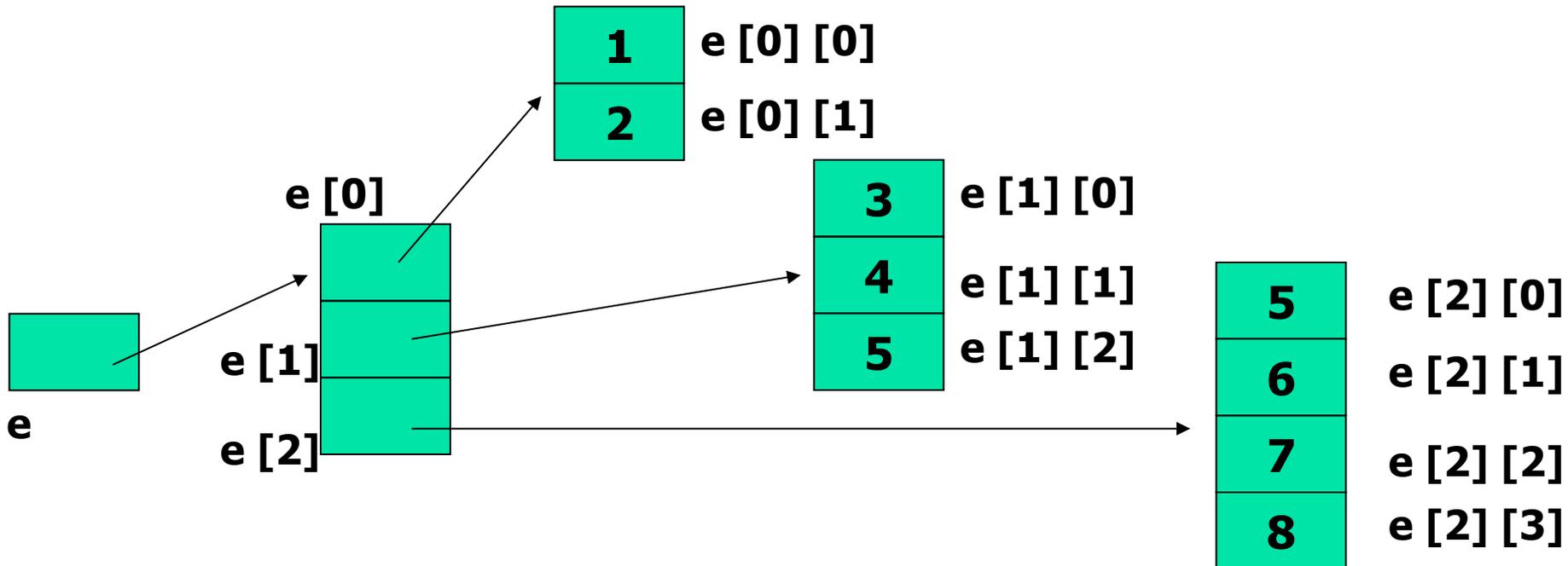
Arrays Multidimensionais

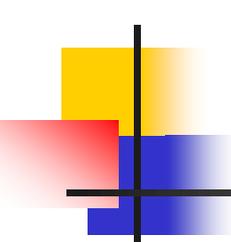
- São implementados como arrays de arrays.
- Para cada dimensão basta especificar um par de colchetes na declaração da variável

Arrays Multidimensionais

- O que faz este código?

```
int[ ][ ] e = {{1, 2}, {3, 4, 5}, {5, 6, 7, 8}};
```



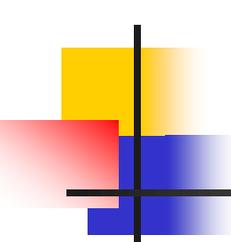


Arrays Multidimensionais

- Qual a diferença?

```
byte [ ][ ] e = new byte[ 2 ][ 4 ];
```

```
byte [ ][ ] e = new byte[ 2 ][ ];  
e [ 0 ] = new byte [ 3 ];  
e [ 1 ] = new byte [ 5 ];
```



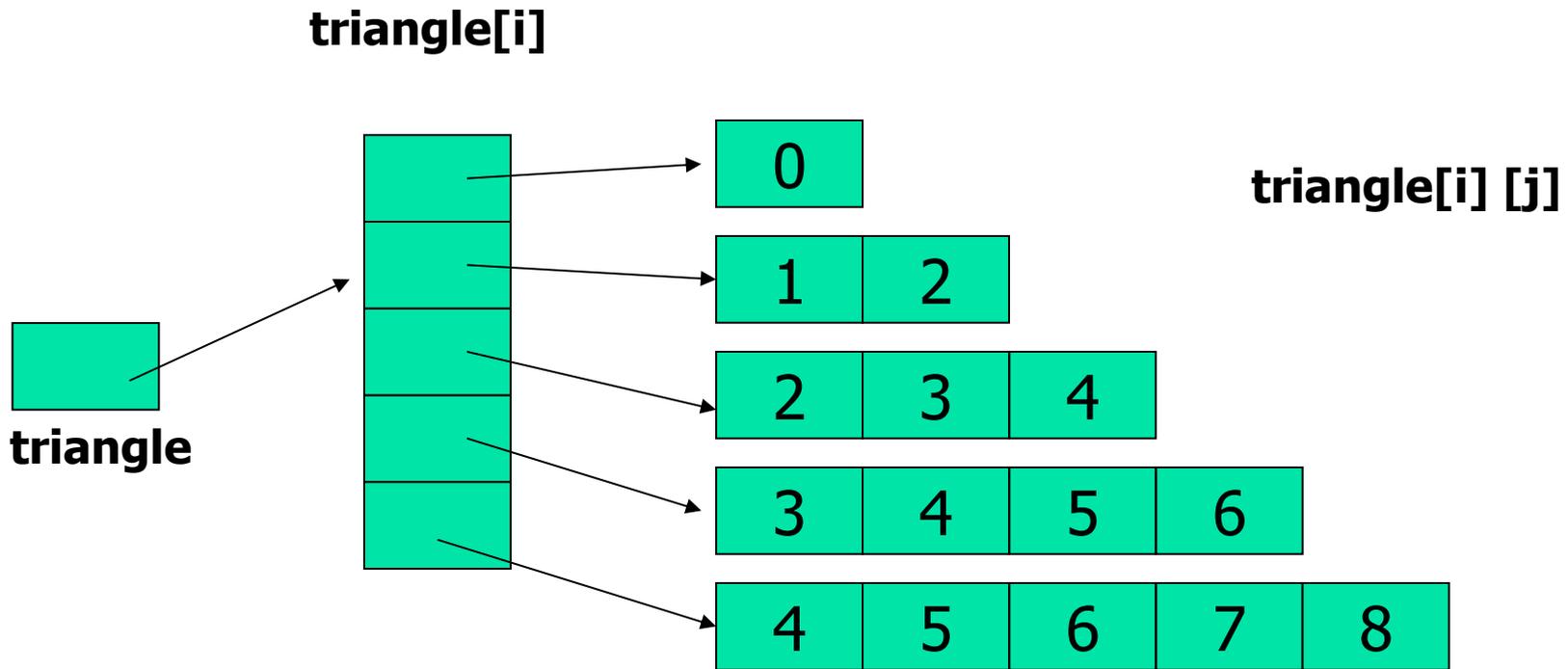
Arrays Multidimensionais

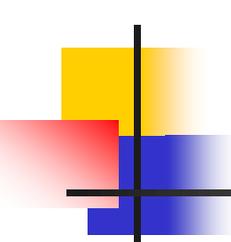
- O que faz este código?

```
short[ ][ ] triangle = new short[5][ ];  
  
for(int i = 0; i < triangle.length; i++) {  
  
    triangle[i] = new short[i+1];  
  
    for(int j=0; j < i+1; j++)  
        triangle[i][j] = (short) i + j;  
  
}
```

Arrays Multidimensionais

- Resultado





Referências

- Slides “Vetores” Prof. Marcos Dósea. UFS. 2010.
- Slides “Arrays em Java”, Prof^a. Débora. UFS. 2010
- Slides “Arrays Java” Prof Giovanni . Java.UFS. 2009.
- Caelum. Java e Orientação a Objetos
 - <http://www.caelum.com.br/apostilas/>
 - Capítulo 5
- Java How to Program
 - Capítulo – 7