

Estrutura de Dados

Arrays

Prof. MSc. David Alain do Nascimento
IFPE Campus Garanhuns

Aula 1

```
int[] array = {3, 41, 28, 79, 7};
```

Memória RAM

valores:

3

41

28

79

7

índices:

0

1

2

3

4

Aula 1

- ▶ Arrays
- ▶ Geração de valores aleatórios com Random

Aula 2

1. Método que recebe um array de inteiros e imprime os valores em uma linha.
 - ▶ Assunto: Acessar os índices de um array
2. Método que recebe um array de inteiros e imprime os valores de trás pra frente em uma linha.
 - ▶ Assunto: Acessar os índices de maneira invertida ($\text{array.length} - 1 - i$)
3. Método que recebe um array de inteiros e imprime os elementos que estão nos índices pares.
 - ▶ Assunto: Verificação se o índice é ímpar ou par
3. Método que recebe um array de inteiros e imprime os elementos cujos valores são ímpares.
 - ▶ Assunto: Verificação se o valor no índice é ímpar ou par
4. Método que recebe um array de inteiros e imprime os elementos de trás pra frente começando o $i = 0$.
 - ▶ Assunto: Acessar os índices finais de trás pra frente usando índice começando em 0
5. Método que cria o array de N inteiros com valores aleatórios entre X e Y e retorna a quantidade de ocorrências de um valor K no array. (*X, Y, N e K são parâmetros*)
 - ▶ Assunto: Gerar valores aleatórios e contar quantas vezes acontece valores dentro de um intervalo
 - ▶ ($\text{valor} \geq X \ \&\& \ \text{valor} \leq Y$)

Aula 2

6. Método que recebe um array de inteiros e retorna a soma de todos os valores.
 - ▶ Assunto: Soma de valores de um array
7. Método que recebe um array de números decimais e retorna a quantidade valores dentro de um intervalo entre X e Y [inclusive]. *(X e Y são parâmetros decimais)*
 - ▶ Assunto: Contar ocorrências de valores dentro de um intervalo
 - ▶ *(valor >= X && valor <= Y)*
8. Método que recebe um array de números decimais e retorna a quantidade valores fora de um intervalo entre X e Y [inclusive]. *(X e Y são parâmetros decimais)*
 - ▶ Assunto: Contar ocorrências de valores fora de um intervalo
 - ▶ *(valor < X || valor > Y)*
9. Método que recebe um array de char e retorna a quantidade de valores fora de um intervalo entre X e Y [inclusive]. *(X e Y são parâmetros char)*
 - ▶ Assunto: Contar ocorrências de valores fora de um intervalo
 - ▶ *(valor < X || valor > Y)*

Aula 3

1. Método que recebe dois array de inteiros e retorna um novo que é a concatenação dos.
2. Método que recebe um array e retorna aquele array invertido.
3. Método que recebe dois arrays e retorna a concatenação do array1 com o array2 invertido. (apenas o array2 é que deve ser invertido).
4. Método que recebe um array de char e retorna quantos caracteres estão fora do intervalo: ['a'-'z'] e ['A'-'Z'].
5. Método que recebe um array de String e retorna quantas vezes a String "ifpe" aparece neste array.
6. Método que recebe um array de inteiros e retorna a soma dos valores ímpares deste array.
7. Método que recebe dois arrays de byte, compara e retorna true se são iguais.
8. Método que recebe dois arrays de String, compara e retorna true se são iguais.
9. Método que recebe dois arrays de char e retorna o índice onde os dois começam a ser diferentes. Caso sejam iguais deverá retornar -1.

Bônus: Contar quantas vezes a palavra “ifpe” aparece em uma String.

Dica: converter para array.

Aula 4

1. Método que recebe um array de números decimais e retorna o maior valor.
 - ▶ `maior = array[0]`
2. Método que recebe um array de números inteiros e retorna o segundo maior valor.
 - ▶ `maior = a[0], segundoMaior = a[1]`
 - ▶ `If a[i] > maior else if ... segundoMaior`
3. Método que recebe um array de String e retorna a String de menor comprimento (quantidade de caracteres).
 - ▶ `str.length()`
4. Método que recebe uma String e retorna true se é um palíndromo.
 - ▶ arara, ovo, osso, radar, reviver, rodador, salas, socos, atorredaderrota, oceusueco
5. Método que recebe um array de inteiros e retorna true está ordenado de maneira crescente.
 - ▶ `a[i] > a[i + 1]`
6. Método que recebe um array de inteiros, verifica sua ordenação e retorna +1 se está ordenado de maneira crescente, -1 se está ordenado de maneira decrescente, e 0 se não estiver ordenado.

Aula 5

1. Método que recebe um array de inteiros, filtra e retorna um array que contenha somente os valores positivos.
 - ▶ Usa dois for's (não é dentro do outro)
2. Método que recebe um array de char e retorna um array contendo apenas a primeira metade do array de entrada.
 - ▶ `i < array.length/2`
3. Método que recebe um array de inteiros e conta quantas vezes acontece de ter números pares adjacentes.
 - ▶ `a[i] ... a[i+1] ...`
4. Método que recebe um array de inteiros e retorna um array de inteiros contendo os valores pares que estavam nos índices ímpares do array de entrada.
 - ▶ Usa dois for's (não é dentro do outro)
5. Método que recebe um array de char e retorna um array que estejam trocadas as letra maiúsculas [A-Z] pelas minúsculas [a-z] e vice-versa e coloque espaço em branco quando não é uma letra.
 - ▶ :D
6. Método que recebe um array de char, e dois inteiros **ini** e **fim** e retorna um array contendo apenas os elementos a partir do índice inicial **ini** até o índice final **fim** do array de entrada.
 - ▶ Ex.: `entrada=[0,2,4,6,7,9,4],ini=1,fim=4` irá produzir a saída=`[2,4,6,7]`
7. Método que recebe dois arrays de char **array1** e **array2**, o índice inicial **ini1** do array1, a quantidade de elementos **tam**, o índice inicial do array2 **ini2** e copia **tam** elementos do array1 a partir do índice **ini1** para o array2 a partir do índice **ini2**. **Você deve checar se é possível copiar de acordo com os tamanhos dos arrays.**
 - ▶ Ex.: `array1=[0,2,4,6,7,9,4],ini1=1,tam=3,ini2=2,array2=[0,0,0,0,0]` irá alterar array2 e deixá-lo assim: `[0,0,2,4,6]`

Aula 6

Matriz - Array bidimensional

Matriz = array de arrays

É um array em que cada elemento deste array é também um array

Sintaxe:

```
int[][] matriz; //Equivalente a (int[])[] matriz
```

Aula 6

Matriz - Array bidimensional

Exemplos:

```
int[][] array =  
    {  
        {1, 2, 1, 2},  
        {1, 3, 1, 3},  
        {1, 1, 1, 1}  
    };
```

```
int[][] array;  
int linhas = 3;  
int colunas = 2;  
  
array = new int[linhas][colunas];
```

```
int[][] array = new int[5][];  
array[0] = new int[5];  
array[1] = new int[6];  
array[2] = new int[4];  
array[3] = new int[3];  
array[4] = new int[7];
```

```
int linhas = 1;  
int colunas = 2;  
int[][] m = new int[linhas][colunas];  
m[0][0] = 3;  
m[0][1] = 6;  
  
for(int l : m)  
    System.out.println(Arrays.toString(l));
```

```
//mostrar os valores  
for(int lin = 0 ; lin < array.length ; lin++){  
    for(int col = 0 ; col < array[0].length ; col++){  
        System.out.print(array[lin][col] + "\t");  
    }  
    System.out.println();  
}
```

Aula 7

Matriz - Array bidimensional

1. Um método que recebe uma matriz de inteiros e retorna o maior valor contido na matriz.
2. Um método que recebe uma matriz de inteiros e retorna um array contendo respectivamente a linha e coluna onde está o menor valor.
3. Um método que recebe duas matrizes de inteiros e retorna true se são iguais.
4. Um método que recebe duas matrizes de inteiros e retorna a soma das matrizes.
5. Um método que recebe uma matriz de inteiros e retorna a soma de todos os valores.
6. Um método que recebe uma matriz de inteiros e retorna a matriz transposta.
7. Um método que recebe uma matriz de inteiros e retorna true se é uma matriz simétrica.
8. Um método que recebe uma matriz de inteiros e retorna um array contendo a soma das **linhas**. Em que no índice 0 do array vai ter a soma dos valores da linha 0 da matriz e assim por diante.
9. Um método que recebe uma matriz de inteiros e retorna um array contendo a soma das **colunas**. Em que no índice 0 do array vai ter a soma dos valores da **coluna** 0 da matriz e assim por diante.
10. Um método que recebe uma matriz de inteiros (com apenas 0 e 1) e verifica se todos os valores da borda são 1 e outros são 0.
11. Um método que recebe uma matriz de inteiros (com apenas 0 e 1) e retorna quantas vezes aparece a submatriz $\{\{0, 1\}, \{1, 0\}\}$.
12. Um método que recebe uma matriz de inteiros e retorna a matriz rotacionada em 90 graus pra direita (sentido horário).