

Exercícios de Arrays Unidimensionais

Exercício 1: Para cada conjunto de valores abaixo, escreva o código Java, usando laço(s), que preencha um array com os valores:

a) 10 9 8 7 6 5 4 3 2 1

b) 0 1 4 9 16 25 36 49 64 81 100

c) 1 2 3 4 5 10 20 30 40 50

d) 3 4 7 12 19 28 39 52 67 84

Exercício 2: Escreva um trecho Java que exiba os valores de um array *a* **double** numa mesma linha.

Exercício 3: Escreva um trecho Java que leia 10 valores **double** do teclado e armazene-os num array *d*.

Exercício 4: Crie um método que recebe um array de inteiros e retorna a quantidade de elementos do array que são números negativos.

Exercício 5: Crie um método que recebe um array de inteiros *a* e um valor inteiro *x* e retorna a quantidade de vezes que *x* aparece no array *a*.

Exercício 6: Escreva um método que recebe um array de inteiros *a* e devolve um array de boolean onde, cada posição indique true se o elemento da posição correspondente de *a* é *positivo* e false caso seja negativo ou zero.

Exercício 7: Escreva um método que recebe um array de números e devolve a posição onde se encontra o maior valor do array. Se houver mais de um valor maior, devolver a posição da primeira ocorrência.

Exercício 8: Faça um método que devolve um array de números lidos do teclado.

Exercício 9: Crie um método que recebe um array de inteiros positivos e substitui seus elementos de valor ímpar por -1 e os pares por +1.

Exercício 10: Crie a classe Carta, que possui um nome e um naipe. Crie os métodos GET e SET para os atributos. Crie agora uma classe Baralho, que possui 52 cartas. No construtor de Baralho, inicialize as 52 cartas (juntamente com os 4 coringas, que devem se chamar “coringa” e o naipe é “coringa”). Escreva os seguintes métodos:

a) *embaralha()* – usando o método `Math.random` dá para trocar as cartas dentro do baralho, misturando-as (sorteia duas posições e troque-as. Repita este processo diversas vezes).

b) *daCarta()* – devolve uma carta, retirada do topo do baralho, se não estiver vazio, ou **null**.

c) *temCarta()* – verifica se tem carta no baralho, devolvendo `true` ou `false`.

A classe *Baralho*, assim construída, será útil para programar diversos jogos de cartas, através de novas classes que os implementem.

d) *imprimeBaralho()* – imprime as cartas para verificar como estão dispostas (se estão embaralhadas por exemplo).