

SENAI



entra2i
blusoft | fapesec

REACT NATIVE

Parte 1

- Array

Parte 2

- Ordenação de Arrays

Parte 3

- Exercícios



Array's ESTRUTURAS DE DADOS

Trabalhar com array, significa trabalhar com **estruturas de repetições**.

O motivo é simples:

Trabalharemos com muitas variáveis e deixaremos o trabalho mais pesado para a estrutura de repetição.

A melhor estrutura de repetição para trabalhar com array é o **para**, pelo simples motivo de sempre termos o número exato dos elementos do array.

Imagine o seguinte problema:

Você precisa criar um algoritmo que lê o nome e as 4 notas de 50 alunos, calcular a média de cada aluno e informar quais foram aprovados e quais foram reprovados.

Conseguiu imaginar quantas variáveis você vai precisar? Muitas né?

Vamos fazer uma conta rápida:

50 variáveis para armazenar os nomes dos alunos

+

(4 * 50 =) 200 variáveis para armazenar as 4 notas de cada aluno

+

50 variáveis para armazenar as médias de cada aluno.

TOTAL : 300 variáveis

Isso sem contar a quantidade de linhas de código que você vai precisar para ler todos os dados do usuário, calcular as médias e apresentar os resultados.



Mas eu tenho uma boa notícia pra você. Nós não precisamos criar **300** variáveis!
Podemos utilizar **Arrays** para isso.

O QUE SÃO ARRAY ?'S

Um array é uma variável que armazena várias variáveis do mesmo tipo.

No problema apresentado anteriormente, nós podemos utilizar um array de 50 posições para armazenar os nomes dos 50 alunos.

ARRAY COM NOMES DOS ALUNOS

Índices 

| [0] | [1] | [2] | [.....] | [49] |
|------|-------|-----------|---------|------|
| João | Maria | Francisco | | José |

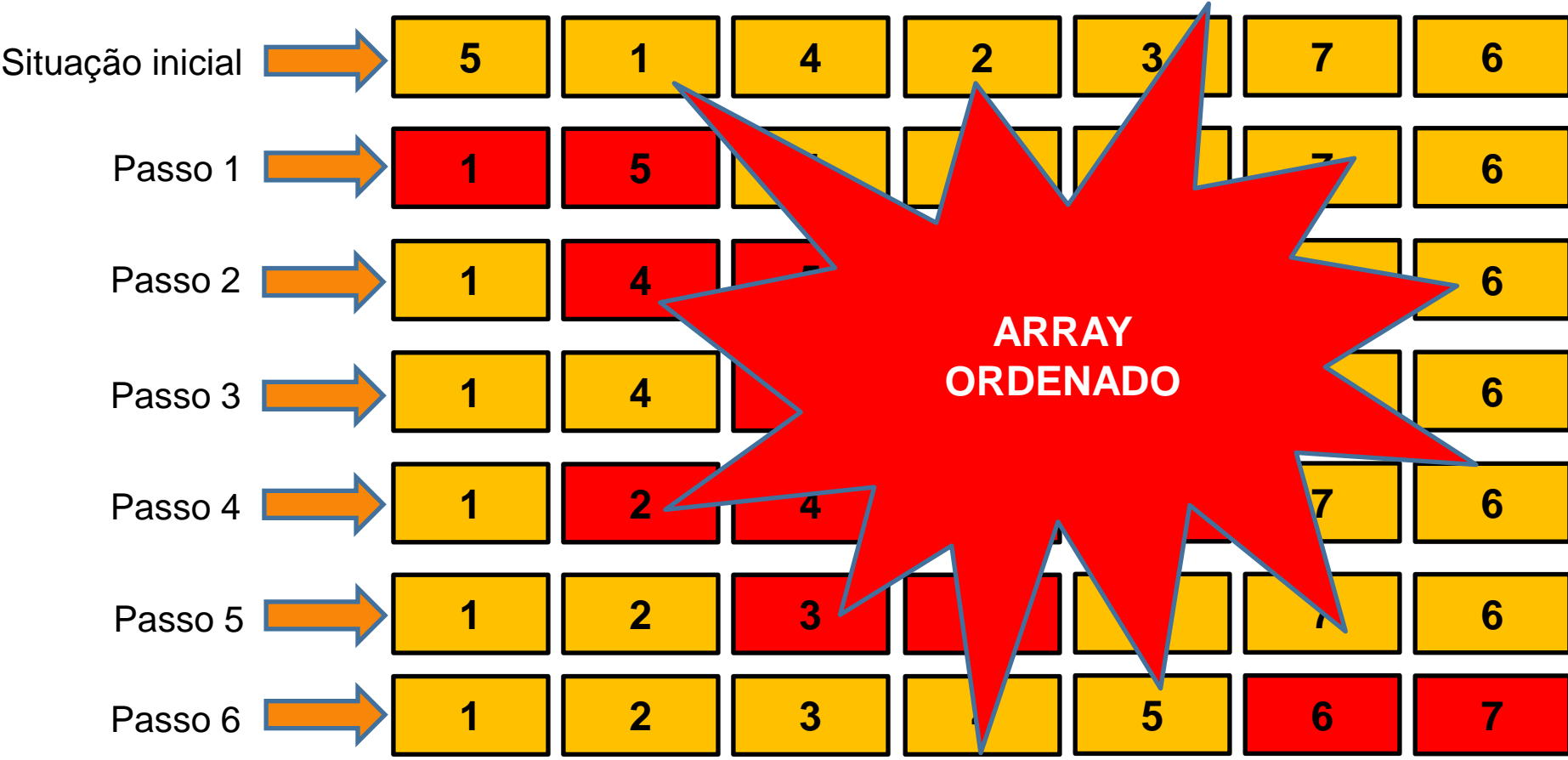
 **Dados**

Como fazer para ordenar os valores dentro de um array.

Para isso utilizamos um método chamado de “**MÉTODO BOLHA**” ou “**BUBBLE SORT**”.

Não é o método mais eficiente, digamos assim, mas preza pela simplicidade e facilidade de implementação e manutenção.

A idéia é percorrer as ocorrências dentro do array repetidas vezes comparando os elementos vizinhos da ocorrência analisada, e se está fora de posição, colocar na posição correta. Veja o exemplo abaixo:



Mas como implementar isso num código ?

Vamos ver um passo a passo no JavaScript

MÉTODO BOLHA DE ORDENAÇÃO DE ARRAYS USANDO NÚMEROS

Vimos como podemos classificar um ARRAY utilizando números.

Mas é possível localizar um número de um array ?

SIM. É possível ! Vamos ver como funciona

VAMOS À PRÁTICA ?



1. Escreva um algoritmo que solicite ao usuário a entrada de 5 números, e que exiba o somatório desses números na tela. Após exibir a soma, o programa deve mostrar também os números que o usuário digitou, um por linha.
2. Escreva um algoritmo que solicite ao usuário a entrada de 5 nomes, e que exiba a lista desses nomes na tela. Após exibir essa lista, o programa deve mostrar também os nomes na ordem inversa em que o usuário os digitou, um por linha.
3. Crie um programa que solicite a entrada de 10 números pelo usuário, armazenando-os em um array, e então monte outro array com os valores do primeiro multiplicados por 5. Exiba os valores dos dois array's na tela, simultaneamente, em duas colunas (um em cada coluna), uma posição por linha.
4. Leia um array de 16 posições e troque os 8 primeiros valores pelos 8 últimos e vice-e-versa. Escreva ao final o array obtido.
5. Leia um array de 40 posições. Contar e mostrar quantos valores pares ele possui.
6. Leia um array de 40 posições e atribua valor 0 para todos os elementos que forem ímpares. Ao término, mostrar como ficou o vetor depois das alterações.

7. Crie um programa que armazene 10 números digitados pelo usuário em dois array's: um somente para números pares, e outro somente para números ímpares. Após, exiba os valores dos dois arrays na tela, em sequência.

Obs.: As posições que não receberem valores exibirão o número zero. Não se preocupe com isso por enquanto.

8. Modifique o programa anterior para não aceitar a entrada do número zero, e requisitar a digitação de outro número neste caso.

9. Modifique novamente o programa anterior, de modo a não exibir na saída os números zero que são mostrados para todas as posições, e que não receberam nenhum valor durante a atribuição (e portanto estão vazias).

10. Carregue um array de inteiros com 20 posições e em seguida informe um valor X qualquer. Seu programa devera fazer uma busca do valor de X no array lido e informar a posição em que foi encontrado ou se não foi encontrado.