**Exercícios com Arrays, For e If (sem While, Do While ou Matrizes)**

**Nível 1:**

1. **Criando e Preenchendo um Array:** Crie um array de 10 elementos do tipo int e preencha-o com valores aleatórios entre 1 e 100. Utilize um loop for para percorrer o array e imprimir cada elemento na tela.
2. **Somando Elementos de um Array:** Crie um array de 10 elementos do tipo int e preencha-o com valores aleatórios entre 1 e 100. Utilize um loop for para percorrer o array e calcular a soma de todos os elementos. Imprima a soma na tela.
3. **Buscando um Elemento em um Array:** Crie um array de 10 elementos do tipo int e preencha-o com valores aleatórios entre 1 e 100. Solicite ao usuário que digite um número. Utilize um loop for para percorrer o array e verificar se o número digitado pelo usuário está presente no array. Se o número for encontrado, imprima a mensagem "Elemento encontrado!" e a posição do elemento no array. Caso contrário, imprima a mensagem "Elemento não encontrado!".

**Nível 2:**

1. **Ordenando um Array (Bubble Sort):** Crie um array de 10 elementos do tipo int e preencha-o com valores aleatórios. Utilize o algoritmo de ordenação Bubble Sort para ordenar o array em ordem crescente. Imprima o array ordenado na tela.
2. **Calculando a Média de um Array:** Crie um array de 10 elementos do tipo int e preencha-o com valores aleatórios entre 1 e 100. Utilize um loop for para percorrer o array e calcular a média de todos os elementos. Imprima a média na tela.
3. **Invertendo um Array:** Crie um array de 10 elementos do tipo int e preencha-o com valores aleatórios entre 1 e 100. Utilize um loop for para percorrer o array e inverter a ordem dos elementos. Imprima o array invertido na tela.

**Nível 3:**

1. **Removendo Elementos Duplicados de um Array:** Crie um array de 10 elementos do tipo int e preencha-o com valores aleatórios entre 1 e 100. Utilize um loop for para percorrer o array e remover os elementos duplicados. Imprima o array final sem elementos duplicados na tela.
2. **Combinando Dois Arrays:** Crie dois arrays de 5 elementos do tipo int e preencha-os com valores aleatórios entre 1 e 100. Crie um novo array que seja a combinação dos dois arrays originais, intercalando os elementos. Imprima o array combinado na tela.
3. **Contando Elementos Pares e Ímpares em um Array:** Crie um array de 10 elementos do tipo int e preencha-o com valores aleatórios entre 1 e 100. Utilize dois contadores para contabilizar a quantidade de elementos pares e ímpares no array. Imprima a quantidade de elementos pares e ímpares na tela.

**Nível 4:**

1. **Encontrando o Maior e Menor Elemento de um Array:** Crie um array de 10 elementos do tipo int e preencha-o com valores aleatórios entre 1 e 100. Utilize variáveis para armazenar o maior e menor valor do array. Utilize loops for e instruções if para encontrar o maior e menor elemento. Imprima o maior e menor valor na tela.
2. **Criando um Array de Strings:** Crie um array de 5 elementos do tipo String e preencha-o com nomes de frutas. Utilize um loop for para percorrer o array e imprimir cada fruta na tela.
3. **Concatenando Strings em um Array:** Crie um array de 5 elementos do tipo String, onde cada elemento represente uma parte de uma frase. Utilize um loop for e concatenação de strings para formar a frase completa. Imprima a frase final na tela.

**Nível 5:**

1. **Criando um Menu Interativo com Array:** Crie um array de opções de menu (por exemplo, "Calcular Soma", "Exibir Média", "Sair"). Utilize um loop for para mostrar as opções do menu ao usuário. Utilize um loop while e instruções if para capturar a escolha do usuário e executar a

**Nível 4 (continuação):**

1. **Buscando Elementos por Valor e Posição:** Crie um array de 10 elementos do tipo int e preencha-o com valores aleatórios entre 1 e 100. Solicite ao usuário que digite um número e um índice (posição) no array. Utilize loops for e instruções if para verificar se o número digitado existe no array e se o índice está dentro dos limites do array. Imprima mensagens informativas sobre a existência do número e a validade do índice.
2. **Calculando Estatísticas de um Array:** Crie um array de 10 elementos do tipo double e preencha-o com valores aleatórios. Utilize loops for e variáveis para calcular a soma, média, variância e desvio padrão do array. Imprima os valores calculados na tela.

**Desafio 1: Jogo de Adivinhação**

Crie um jogo de adivinhação onde o computador pensa em um número aleatório entre 1 e 100 e o jogador tenta adivinhar o número. Utilize os conceitos de arrays, for, if e instruções random para:

1. Gerar o número aleatório a ser adivinhado.
2. Permitir que o jogador insira chutes até acertar o número secreto.
3. Fornecer feedback ao jogador após cada chute, indicando se o chute foi maior, menor ou igual ao número secreto.
4. Imprimir uma mensagem de parabéns quando o jogador acertar o número secreto.

**Desafio 2: Análise de Dados de Vendas**

Imagine que você possui um array que armazena dados de vendas de um e-commerce em um determinado período. Cada elemento do array é um objeto que contém informações como ID da venda, data da venda, valor total da venda e nome do cliente. Utilize os conceitos de arrays, for, if e funções para:

1. Calcular o total de vendas no período.
2. Encontrar a venda com o maior valor total.
3. Identificar o cliente que mais comprou no período, considerando o total de suas compras.
4. Exibir um relatório na tela com as informações calculadas (total de vendas, venda de maior valor, cliente que mais comprou).

**Desafio 3: Sistema de Gerenciamento de Tarefas**

Desenvolva um sistema simples de gerenciamento de tarefas utilizando arrays, for, if e funções para:

1. Permitir ao usuário adicionar novas tarefas (com descrição e prioridade).
2. Exibir uma lista de todas as tarefas cadastradas, ordenadas por prioridade.
3. Permitir ao usuário marcar tarefas como concluídas.
4. Remover tarefas concluídas da lista.
5. Salvar as tarefas em um arquivo de texto para persistirem mesmo após o término da execução do programa.

**Dicas:**

* Utilize arrays para armazenar as tarefas, seus atributos e o status de conclusão.
* Crie funções para cada ação principal do sistema (adicionar, exibir, marcar como concluída, remover, salvar).
* Utilize loops for para percorrer os arrays e realizar as operações desejadas.
* Utilize instruções if para controlar o fluxo do programa e tomar decisões.
* Para salvar as tarefas em um arquivo de texto, utilize classes como FileWriter e BufferedWriter do Java.