

Exercícios de Programação em C - Arrays Unidimensionais e Multidimensionais

Danillo Roberto Pereira

Exercícios

1. Declaração, inicialização e representação interna de arrays unidimensionais

Exercício 1: Declare um array de 10 inteiros e inicialize-o com valores de 0 a 9. Escreva um programa em C que imprima todos os elementos do array.

Exercício 2: Declare um array de 5 caracteres e inicialize-o com as letras 'a', 'b', 'c', 'd' e 'e'. Imprima os elementos do array.

Exercício 3: Declare um array de 7 inteiros e inicialize-o com os valores dos primeiros 7 números primos. Escreva um programa em C que imprima todos os elementos do array.

Exercício 4: Declare um array de 8 números de ponto flutuante e inicialize-o com os valores dos primeiros 8 números da sequência de Fibonacci. Imprima todos os elementos do array.

2. Acesso a elementos de forma indexada

Exercício 5: Escreva um programa em C que leia 5 valores inteiros, armazene-os em um array e depois imprima os valores na ordem inversa.

Exercício 6: Escreva um programa em C que leia 10 valores inteiros e calcule a soma de todos os elementos do array.

Exercício 7: Escreva um programa em C que leia 6 valores de ponto flutuante e calcule a média dos valores.

Exercício 8: Escreva um programa em C que leia 8 caracteres e verifique quantos são vogais.

3. Geração de números aleatórios

Exercício 9: Escreva um programa em C que preencha um array de 20 inteiros com números aleatórios entre 1 e 100. Em seguida, imprima todos os elementos do array.

Exercício 10: Escreva um programa em C que preencha um array de 10 inteiros com números aleatórios entre 50 e 150. Calcule e imprima a média dos elementos do array.

Exercício 11: Escreva um programa em C que preencha um array de 15 inteiros com números aleatórios entre 0 e 50. Em seguida, encontre e imprima o maior valor do array.

Exercício 12: Escreva um programa em C que preencha um array de 12 inteiros com números aleatórios entre -20 e 20. Em seguida, conte e imprima quantos números são negativos.

4. Relação entre ponteiros e arranjos unidimensionais

Exercício 13: Escreva um programa em C que declare um array de 5 inteiros. Use um ponteiro para acessar e imprimir todos os elementos do array.

Exercício 14: Escreva um programa em C que declare um array de 5 inteiros e um ponteiro para inteiro. Use o ponteiro para modificar os valores dos elementos do array. Imprima o array resultante.

Exercício 15: Escreva um programa em C que declare um array de 7 inteiros e use um ponteiro para calcular a soma dos elementos do array.

Exercício 16: Escreva um programa em C que declare um array de 10 caracteres e use um ponteiro para contar quantos desses caracteres são letras maiúsculas.

5. Acesso a componentes usando aritmética de ponteiros

Exercício 17: Escreva um programa em C que declare um array de 5 inteiros e use aritmética de ponteiros para somar 10 a cada elemento do array. Imprima o array resultante.

Exercício 18: Escreva um programa em C que declare um array de 10 inteiros e use aritmética de ponteiros para calcular a soma de todos os elementos do array. Imprima a soma.

Exercício 19: Escreva um programa em C que declare um array de 8 números de ponto flutuante e use aritmética de ponteiros para calcular a média dos valores.

Exercício 20: Escreva um programa em C que declare um array de 12 caracteres e use aritmética de ponteiros para inverter a ordem dos caracteres no array.

6. Resolução de problemas envolvendo arranjos unidimensionais

Exercício 21: Escreva um programa em C que encontre o maior e o menor elemento em um array de 10 inteiros.

Exercício 22: Escreva um programa em C que conte o número de elementos pares e ímpares em um array de 20 inteiros.

Exercício 23: Escreva um programa em C que determine se um array de 10 inteiros é simétrico (um palíndromo).

Exercício 24: Escreva um programa em C que verifique se todos os elementos de um array de 5 inteiros são positivos.

7. Manipulação de strings - uso da biblioteca <string.h>

Exercício 25: Escreva um programa em C que leia uma string e conte o número de caracteres, palavras e linhas na string.

Exercício 26: Escreva um programa em C que leia duas strings e concatene-as. Imprima a string resultante.

Exercício 27: Escreva um programa em C que leia uma string e verifique se ela é um palíndromo.

Exercício 28: Escreva um programa em C que leia uma string e substitua todas as ocorrências de um caractere por outro.

8. Arrays de ponteiros

Exercício 29: Escreva um programa em C que use um array de ponteiros para armazenar e imprimir 5 strings.

Exercício 30: Escreva um programa em C que use um array de ponteiros para armazenar 3 strings, ordene as strings em ordem alfabética e as imprima.

Exercício 31: Escreva um programa em C que use um array de ponteiros para armazenar 5 números inteiros e encontre o maior valor entre eles.

Exercício 32: Escreva um programa em C que use um array de ponteiros para armazenar 4 strings e encontre a string de maior comprimento.

9. Passagem de parâmetros na linha de comandos (argc e argv)

Exercício 33: Escreva um programa em C que receba 3 números inteiros como argumentos na linha de comando e imprima a soma deles.

Exercício 34: Escreva um programa em C que receba uma string como argumento na linha de comando e conte o número de caracteres dessa string.

Exercício 35: Escreva um programa em C que receba um número inteiro como argumento na linha de comando e verifique se ele é par ou ímpar.

Exercício 36: Escreva um programa em C que receba 5 números inteiros como argumentos na linha de comando e imprima o maior deles.

10. Passagem de arranjos (ponteiros) como parâmetros

Exercício 37: Escreva um programa em C que declare um array de 5 inteiros e passe esse array para uma função que calcule a média dos elementos do array.

Exercício 38: Escreva um programa em C que declare um array de 10 inteiros e passe esse array para uma função que encontre o maior elemento do array.

Exercício 39: Escreva um programa em C que declare um array de 6 inteiros e passe esse array para uma função que conte quantos elementos são positivos.

Exercício 40: Escreva um programa em C que declare um array de 8 inteiros e passe esse array para uma função que inverta a ordem dos elementos do array.