

Exercícios (treino)

1. Escreva um programa que contenha duas variáveis inteiras. Compare seus endereços e exiba o maior endereço.
2. Escreva um programa que contenha duas variáveis inteiras. Leia essas variáveis do teclado. Em seguida, compare seus endereços e exiba o conteúdo do maior endereço.
3. Crie um programa que contenha um array de float com 10 elementos. Imprima o endereço de cada posição desse array.
4. Crie um programa que contenha uma matriz de float com três linhas e três colunas. Imprima o endereço de cada posição dessa matriz.
5. Crie um programa que contenha um array de inteiros com cinco elementos. Utilizando apenas aritmética de ponteiros, leia esse array do teclado e imprima o dobro de cada valor lido.
6. Crie um programa que contenha um array com cinco elementos inteiros. Leia esse array do teclado e imprima o endereço das posições contendo valores pares.
7. Escreva um programa que declare um inteiro, um real e um char, e ponteiros para inteiro, real, e char. Associe as variáveis aos ponteiros (use &). Modifique os valores de cada variável usando os ponteiros. Imprima os valores das variáveis antes e após a modificação.

Exercícios (seminário)

1. Faça um programa que leia dois valores inteiros e chame uma função que receba estas duas variáveis e troque o seu conteúdo, ou seja, esta função é chamada passando duas variáveis A e B por exemplo e, após a execução da função, A conterà o valor de B e B terá o valor de A.
2. Crie uma função que receba dois parâmetros: um vetor e um valor do mesmo tipo do vetor. A função deverá preencher os elementos de vetor com esse valor. Não utilize índices para percorrer o vetor, apenas aritmética de ponteiros.
3. Crie uma função que receba como parâmetro um vetor e o imprima. Não utilize índices para percorrer o vetor, apenas aritmética de ponteiros.

4. Considere a seguinte declaração: `int a,*b,**c,***d`. Escreva um programa que leia a variável `a` e calcule e exiba o dobro, o triplo e o quádruplo desse valor utilizando apenas os ponteiros `b`, `c` e `d`. O ponteiro `b` deve ser usado para calcular o dobro, `c`, o triplo, e `d`, o quádruplo.
5. Faça um programa que leia dois valores inteiros e chame uma função que receba estes 2 valores de entrada e retorne o maior valor na primeira variável, e o menor valor na segunda variável. Escreva o conteúdo das 2 variáveis na tela.
6. Crie um programa que contenha uma função que permita passar por parâmetro dois números inteiros `A` e `B`. A função deverá calcular a soma entre estes dois números e armazenar o resultado na variável `A`. Esta função não deverá possuir retorno, mas deverá modificar o valor do primeiro parâmetro. Imprima os valores de `A` e `B` na função principal.
7. Crie uma função que receba dois parâmetros: um array e um valor do mesmo tipo do array. A função deverá preencher os elementos de array com esse valor. Não utilize índices para percorrer o array, apenas aritmética de ponteiros.
8. Crie uma função para somar dois arrays. Esta função deve receber dois arrays e retornar a soma em um terceiro array. Caso o tamanho do primeiro e segundo array seja diferente, então a função retornará ZERO (0). Caso a função seja concluída com sucesso, a mesma deve retornar o valor UM (1). Utilize aritmética de ponteiros para manipulação do array.
9. Escreva uma função que retorne o maior valor de um array de tamanho `N`. Escreva um programa que leia `N` valores inteiros, imprima o array com `k` elementos por linha, e o maior elemento. O valor de `k` também deve ser fornecido pelo usuário.
10. Escreva um programa em C que leia dois vetores de inteiros com 10 elementos cada. Crie uma função que receba três ponteiros: dois para os vetores de entrada e um para o vetor de saída. Essa função deve calcular a soma elemento a elemento dos dois vetores e armazenar o resultado no vetor de saída. Em seguida, o programa deve exibir o vetor resultante.
11. Escreva um programa que leia uma matriz de inteiros com dimensão 4x4. Em seguida, crie uma função que receba essa matriz por meio de ponteiros e calcule a média dos elementos de cada linha, armazenando os resultados em um vetor de 4 posições. Exiba esse vetor ao final.

