

LISTA SOBRE PONTEIROS

1. Pretende-se que seja feito um programa que faça a gestão de inscrições de alunos em um determinado curso prático. Cada aluno tem um número de identificação (um número inteiro) e, obviamente, um nome (uma cadeia de caracteres com, no máximo, 60 caracteres). Sabendo-se que a capacidade máxima do curso é de 30 alunos. As operações a implementar são as seguintes:

- * inscrição de um aluno
- * listagem dos alunos do turno, por ordem alfabética
- * remoção de um aluno, dado o seu número.

2. Escreva um programa que declare uma matriz 100x100 de inteiros. Você deve inicializar a matriz com zeros usando ponteiros para endereçar seus elementos. Preencha depois a matriz com os números de 1 a 10000, também usando ponteiros.

3. Verifique o programa abaixo. Encontre o seu erro e corrija-o para que escreva o número 10 na tela.

```
#include <stdio.h>
int main()
{
    int x, *p, **q;
    p = &x;
    q = &p;
    x = 10;
    printf("\n%d\n", &q);
    return(0);
}
```

4. Aprendemos, que o valor de uma variável ou expressão do tipo vetor é o endereço do elemento zero do vetor. Seja **a[]** um vetor qualquer, independente de tipo e tamanho, e **pa** um ponteiro para o mesmo tipo de **a[]**. Responda **V** ou **F**, justificando as que forem falsas:

- a) () Qualquer expressão escrita por meio do nome do vetor e de um índice é equivalente a escrever uma outra expressão por meio de um apontador e um deslocamento;
- b) () Após a atribuição **pa=&a[0]**; **pa** e **a** possuem valores idênticos
- c) () A atribuição **pa=&a[0]**; pode ser escrita como **pa=a**;
- d) () Uma referencia a **a[i]** pode ser escrita como ***(a+i)**
- e) () **&a[i]** e **a+i** são idênticos
- f) () **a+i** é o endereço do i-ésimo elemento após **a**
- g) () **pa[i]** é idêntico a ***(pa+i)**

- h) () **pa=a** é uma operação válida
- i) () **pa++** e' uma operação válida
- j) () **a=pa** é uma operação válida
- k) () **a++** é uma operação valida

5. O programa abaixo está errado. Por quê?

```
void main(void)
{
    int x, *p;
    x = 10;
    p = x;
    printf ("%d", *p);
}
```

6. Defina o que o programa abaixo faz e faça uma versão dele para uso de ponteiros.

```
void copia(s1,s2)
char s1[], s2[];
{
    int i = 0;
    while((s1[i]=s2[i]) != '\0')
        i++;
}
void main()
{
    char st1[10], st2[10];
    printf("digite uma string: %s\n");
    scanf("%s",st1);
    printf("digite uma string: %s\n");
    scanf("%s",st2);
    printf("os valores sao %s e %s\n",st1,st2);
    copia(st1,st2);
    printf("st1 e %s",st1);
}
```

- 7. Faça um programa que busque um elemento em um vetor de inteiros com 20 posições. O algoritmo deve imprimir quantas vezes o elemento procurado aparece e em que posição(ões) ele se encontra. **Observação:** seu programa deve usar ponteiros na busca e impressão.
- 8. Considere 2 vetores ordenados A e B, com 5 posições cada um. Crie um programa em C que coloque os elementos de A e B em um novo vetor Z de 10 posições de forma que o vetor Z também esteja ordenado.

9. A criptografia é um importante recurso da computação moderna. A língua do P, brincadeira comum das crianças, é um modo simples de "criptografar" uma mensagem. Ela consiste em substituir todas as consoantes da frase por P. Crie um programa em C que dada uma frase de no máximo 100 caracteres passe a mesma para a língua do P. (As frases não devem conter acentuação, mas podem conter pontuação).
10. Dada uma matriz B, determine a linha de B que possui a maior soma de seus elementos, use ponteiros para resolução desse problema.
11. Seja R uma matriz 4x5. Faça uma função que determine o maior elemento de R e a sua posição. **Observação:** use ponteiros para resolução desse problema.
12. Uma matriz quadrada inteira é chamada de "quadrado mágico" se a soma dos elementos de cada linha, a soma dos elementos de cada coluna e a soma dos elementos das diagonais principal e secundária são todos iguais. Exemplo: A matriz abaixo representa um quadrado mágico:

| 8 0 7 |

| 4 5 6 |

| 3 10 2 |

Escreva uma função que utilize ponteiros para verificar se uma matriz de **n** linhas e **n** colunas representa um quadrado mágico. Construa também a função **main** para uso da sua função.

13. Faça uma função para mostrar o **n**ésimo elemento **n**, de um vetor **v**. A função deve receber por referencia, tanto **v**, quanto **n**.
14. Faça um programa que leia um vetor de inteiros **v** do usuário, e armazene em outro vetor **v2** o quadrado de cada um dos números de **v** na respectiva posição. **Observação:** seu programa deve fazer uso de ponteiros para resolução do problema.
15. Uma variável do tipo ponteiro é estática ou dinâmica? Explique.
16. Escreva um programa que leia uma frase de até 80 caracteres e a imprima em ordem reversa convertendo todos os caracteres minúsculos para maiúsculos. **Observação:** seu programa deve fazer uso de ponteiros para resolução do problema.

17. Escreva uma função que receba um ponteiro para uma string e troque todo caracter existente após um espaço em branco, pelo seu equivalente maiúsculo. **Observação:** seu programa deve fazer uso de ponteiros para resolução do problema.
18. Faça uma função que verifique se a expressão apontada por **sub_str** está presente na cadeia apontada por **str**. Essa função deve retornar a posição em que a subcadeia aparece na cadeia. A função possui o seguinte protótipo:
int posicao(char *substr, char *str);
19. Escreva um programa que simule uma pilha usando vetores. O programa deve implementar as seguintes operações na pilha:
- Inserir
 - Remover
 - Listar todos os elementos da Pilha.