

PONTEIROS

1. Quais serão os valores de x, y e p ao final do trecho de código abaixo?

```
int x, y, *p; y = 0;
p = &y;
x = *p;
x = 4;
(*p)++;
--x;
(*p) += x;
```

2. Os programas (trechos de código) abaixo possuem erros. Qual(is)? Comodeveriam ser?

a)

```
void main() {
int x, *p;
x = 100;
p = x;
printf("Valor de p: %d.\n", *p);}
```

b)

```
void troca (int *i, int *j) {
int *temp;
*temp = *i;
*i = *j;
*j = *temp;}
```

c)

```
char *a, *b;

a = "abacate";

b = "uva";

if (a < b)

printf ("%s vem antes de %s no dicionário", a, b);

else

printf ("%s vem depois de %s no dicionário", a, b);
```

3) Suponha que os elementos do vetor v são do tipo int e cada int ocupa 8 bytes no seu computador. Se o endereço de v[0] é 55000, qual o valor da expressão v + 3?

4)Escreva uma função mm que receba um vetor inteiro v[0..n-1] e os endereços de duas variáveis inteiras, digamos min e max, e deposite nessas variáveis o valor de um elemento mínimo e o valor de um elemento máximo do vetor. Escreva também uma função main que use a função mm.

5)Suponha que v é um vetor. Descreva a diferença conceitual entre as expressões v[3] e v + 3.

6)(sem usar o computador) Qual o conteúdo do vetor a depois dos seguintes comandos.
`int a[99]; for (i = 0; i < 99; ++i) a[i] = 98 - i; for (i = 0; i < 99; ++i) a[i] = a[a[i]];`

7)Escreva uma função chamada troca que troca os valores dos parâmetros recebidos. Sua assinatura deve ser: `void troca(float *a, float *b);`

8)Crie uma função que receba uma string como parâmetro (de tamanho desconhecido) e retorne uma cópia da mesma. A assinatura da função deve ser: `char *strcpy(char *str);`

9)Escreva uma função que recebe como parâmetros um vetor de inteiros v, o número de elementos dele N e ponteiros para variáveis nas quais devem ser armazenados os valores máximo e mínimo do vetor. Sua assinatura deve ser: `void maximoMinimo(int *v, int N, int *maximo, int *minimo);`

10) Qual o resultado do código abaixo?
Explique cada linha.
`int x = 100, *p, **pp;
p = &x;
pp = &p;
printf("Valor de pp: %d\n", **pp);`

11)Escreva uma função que recebe uma string de caracteres e uma letra e devolve um vetor de inteiros contendo as posições (índices no vetor da string) onde a letra foi encontrada) e um inteiro contendo o tamanho do vetor criado (total de letras iguais encontradas). Utilize

o retorno de um vetor para retornar os índices e um ponteiro para guardar o tamanho do vetor.

12. O que significa o operador asterisco em cada um dos seguintes casos:

- a) `int *p;`
- b) `cout << *p;`
- c) `*p = x*5;`
- d) `cout << *(p+1);`

13. Qual é a saída deste programa?

```
#include <iostream.h>

void main( ) {

int i=5, *p;

p = &i;

cout << p << '\t' << (*p+2) << '\t' << **&p << '\t' << (3**p) << '\t' << (**&p+4);

}
```

STRUCTS

1. Escrever um programa que cadastre o nome, a matrícula e duas notas de vários alunos. Em seguida imprima a matrícula, o nome e a média de cada um deles.

2. Escrever um programa que cadastre o nome, a altura, o peso, o cpf e sexo de algumas pessoas. Com os dados cadastrados, em seguida localizar uma pessoa através do seu CPF e imprimir o seu IMC.

3. Escrever um programa que cadastre vários produtos. Em seguida, imprima uma lista com o código e nome de cada produto. Por último, consulte o preço de um produto através de seu código.

4. Escreva um programa que simule contas bancárias, com as seguintes especificações:

- Ao iniciar o programa vamos criar contas bancárias para três clientes.
 - Cada conta terá o nome e o CPF do cliente associado a ela.
 - No ato da criação da conta o cliente precisará fazer um depósito inicial.
- Após as contas serem criadas, o sistema deverá possibilitar realizações de saques ou depósitos nas contas.
 - Sempre que uma operação de saque ou depósito seja realizada, o sistema deverá imprimir o nome do titular e o saldo final da conta.

5.

Considerando uma lista encadeada de valores inteiros definida pelo tipo abaixo:

```
struct lista {  
    int info;  
    struct lista* prox;  
};  
typedef struct lista Lista;
```

Escreva uma função que retire o último elemento de uma dada lista. Esta função deve ter como valor de retorno o ponteiro para a lista alterada e deve obedecer o seguinte protótipo:

```
Lista* retira_ultimo (Lista* l);
```

6.

Considere estruturas de listas encadeadas que armazenam valores inteiros. O tipo que representa um nó da lista é dado por:

```
struct lista {  
    int info;  
    struct lista* prox;  
};  
typedef struct lista Lista;
```

Implemente uma função que receba um vetor de valores inteiros com n elementos e construa uma lista encadeada armazenando os elementos do vetor nos nós da lista. Assim, se for recebido o vetor $v[5] = \{3, 8, 1, 7, 2\}$, a função deve retornar uma nova lista cujo primeiro nó tem a informação 3, o segundo a informação 8, e assim por diante. Se o vetor tiver zero elementos, a função deve ter como valor de retorno uma lista vazia. O protótipo da função é dado por:

```
Lista* constroi (int n, int* v);
```

7.

Considere a implementação de uma lista encadeada para armazenar as notas dos alunos de uma turma. O tipo `Lista` representa a estrutura do nó da lista conforme declaração abaixo:

```
typedef struct lista Lista;
```

Pede-se:

- Defina `struct lista` de tal forma que a informação associada a cada nó da lista seja composta por:
Nome do aluno: cadeia com até 80 caracteres
Nota do aluno: número real
- Escreva uma função para inserir um novo elemento na lista. A ordem em que o elemento será inserido na lista é irrelevante. Esta função deve ter como valor de retorno o ponteiro para a lista alterada e deve obedecer o seguinte protótipo:
`Lista* insere (Lista* l, char* nome, float nota);`

8. Dada uma lista duplamente encadeada e todos os seus nós tem campos de inteiros, para a guarda de informação, prox, ponteiro que aponta para o próximo da lista, e ant, que é um ponteiro que aponta para o anterior da lista. Temos a inserção de a, b, c, d, x, y e z. A lista fica da seguinte maneira:

a-> b-> c-> x-> y-> z

A lista tem um ponteiro pri que aponta para o seu primeiro elemento (a, no caso). Quais serão os nós acessados nos seguintes casos?

- a) pri->prox->prox->ant->prox->prox
- b) pri->ant->ant->prox->ant->prox
- c) pri -> ant-> prox -> prox -> prox -> ant

9. Dada uma lista duplamente encadeada com o seguinte formato:

55 -> 17 -> 5 -> 12 -> 0 -> 40 -> 8 -> 3

Considerando que pri aponta para o primeiro elemento, qual será o valor de x?

- a) $x = \text{pri} * \text{pri} \rightarrow \text{prox} \rightarrow \text{prox} \rightarrow \text{ant} - \text{pri} \rightarrow \text{ant}$
- b) $x = \text{pri} \rightarrow \text{ant} * \text{pri} \rightarrow \text{prox} - \text{pri} \rightarrow \text{ant} \rightarrow \text{ant} \rightarrow \text{ant}$
- c) $x = (\text{pri} \rightarrow \text{ant} \rightarrow \text{ant} \rightarrow \text{ant} / \text{pri} \rightarrow \text{ant} \rightarrow \text{ant} == \text{pri} \rightarrow \text{prox} \rightarrow \text{prox})$

10. Considere a função abaixo.

```
bool Func (ListaDEncad *pri){  
    if(pri -> prox -> prox == pri -> ant -> ant)  
        return true;  
  
    return false;  
  
}
```

Suponha que todos os nós desta lista são diferentes sempre. Quantos nós a lista deve ter para que esta função retorne verdadeiro? Justifique.