1. O que é e para que serve um ponteiro?

Ponteiro é um tipo especial de variável que serve para armazenar um endereço de memória de outra variável, podendo manipula-la.

1. Qual a sintaxe correta para declarar um ponteiro?

tipo[int,char,float,double] \*ponteiro;

1. Diferencie os operadores de ponteiros & e \* em C.

& acessa a posição de memória da variável apontada ,independentemente se está sendo apontada ou não.

\*pega o conteúdo da variável apontada

1. Assumindo que o endereço da variável i foi atribuída ao ponteiro p, qual a forma correta de referenciar i através do ponteiro?

p=&i;

\*p

1. Explique qual a diferença entre um ponteiro do tipo float, int e char.

Ponteiro tipo int aponta somente com numeros inteiros.

Char aponta para variaveis do tipo char.

Float aponta para variaveis com numeros que tenham casas decimais.

1. Supondo que c seja um ponteiro para char e i um ponteiro para int, explique a diferença entre realizar as operações i++ e c++.

Operação i++: incrementará fazendo com que o ponteiro avança para a próxima posição do vetor.

Exemplo: Supondo que i =vet; após o incremento i irá ficar i=vet[1].

Char c=vet; após fica c=vet[[1];

1. Supondo o código abaixo:

int \*p;

int num;

p = &num;

Diga se cada uma das comparações é verdadeira ou falsa, e justifique sua resposta:

a) num == &p;  
Resposta: FALSA, pois não há valor definido para num, portanto não é possível afirmar que o valor de num é igual ao numero do endereço de memória que está armazenado o ponteiro p;

b) num == \*p; VERDADEIRA,

dizer que o valor que está contido em num é o mesmo que dizer que é igual ao valor da variável para qual p está apontando, e no caso, p aponta para num, ou seja, sim, conteúdo de num é igual a conteúdo de num;

c) p == \*num;

FALSA, pois neste caso num não é um ponteiro.

d) p == &num;

Verdadeiro: p recebe a posição de memória da variável num.

1. O trecho de código abaixo falta uma linha para funcionar. Insira essa linha no código e justifique o porquê dela ser necessária:

int num, \*pont;

p=&num;

\*pont = 10;

printf("o número é %d", num);

1. Escreva um trecho de código que divida o conteúdo de uma variável X por 2 sem utilizar o X na expressão.

Resposta no arquivo 9.c

1. . Supondo que p seja um ponteiro para float, explique a diferença entre os comandos p++, \*p++, (\*p)++, e \*(p++);

* p++: Aponta para a próxima posição de memória do vetor (4 bytes a frente)
* \*p++: O conteúdo em que p está apontando, neste caso é incrementado, por exemplo se o conteúdo apontado por p vale 2, ele recebe o incremento e vira 3.
* (\*p)++: Faz a mesma coisa que \*p++;
* \*(p++) O ponteiro acessa o conteúdo da próxima posição de memória do vetor.

11Explique qual a relação que um vetor tem com um ponteiro.

Vetor tem como relação a um ponteiro as suas posições memória, o ponteiro facilitada o acesso a alguma posição do vetor que deve ser usada no meu código.

12)Supondo que vet seja um vetor, e p seja um ponteiro do mesmo tipo do vetor. Qual a diferença entre se fazer a atribuição p = vet e p = &vet[0]? Explique sua resposta.

Não há nenhuma diferença pois nos dois casos o ponteiro irá receber da mesma maneira, a primeira posição do vetor.

13)Supondo que um ponteiro p esteja apontando para um vetor do tipo int. Logo após é apresentada a seguinte linha:

printf("%d",\*(p+7)); O que aparecerá na tela? Por quê?

Vai aparecer na tela o conteúdo da posição 7 do vetor, pois p+7 faz com que o ponteiro incremente e avance da posição 0 até a 7° e o \* faz com que ele acesse o conteúdo desta posição ficando \*(p+7).

14)Supondo um vetor do tipo char chamado vet. A quinta posição de um vetor pode ser acessada pelo comando \*(vet+5)? Caso contrário justifique e sugira uma solução nos mesmos moldes. E se o vetor fosse do tipo int, o que mudaria? Justifique.

Não, pois neste caso o ponteiro está acessando a 6° posição do vetor, o correto para acessar a 5° posição seria \*(v+4). E se o vetor fosse do tipo int não iria fazer diferença pois ele estaria acessando a 6° posição.

15)Explique detalhadamente o que o trecho de código abaixo faz:

*int vet[10],\*pv,i=0;*

*pv = vet;*

*while (pv < &vet[10]){*

*i++;*

*\*pv = i;*

*pv++;*

*}*

RESPOSTA: arquivo 15.c

16)O que a função sizeof() faz? O que irá retornar caso pedíssemos para mostrar na tela o valor de retorno de sizeof(int)? e de sizeof(char)? Explique.

sizeof() retorna o tamanho que o tipo de variável ocupa, ou seja cada tipo de variável,int,char,float,double tem o seu tamanho espefícifico. Por exemplo:

sizeof(int) retorna 4bytes;

sizeof(char) retorna 1byte;

17)Usando a função sizeof(), como podemos saber qual o tamanho de um vetor qualquer, sabendo somente seu tipo e nome? Escreva a linha de código que retorna o tamanho deste vetor.

Resposta: arquivo 17.c

18) O código abaixo contém um erro. Explique-o e faça a correção:

*int \*p;*

*int i;*

*p = &i;*

*scanf("%d",\*p);*

Resposta: arquivo 18.c

19)Como é feita a declaração de um ponteiro para ponteiro e para que eles servem? Dê um exemplo de uso.

DECLARAÇÃO/EXEMPLO:

Int/char/float \*\*ppnome;

int \*p;

int \*pp;

int num;

p=&num;

pp=p;

SERVE:

Para referenciar e armarzenar o endereço de memória de outro ponteiro (ficou incompleta minha resposta :/)