Roteiro Ponteiro

1. Quais são os valores exibidos em tela antes e depois da atribuição feita em \*pt ? Explique sua resposta:

R.

Valor exibido em tela antes da atribuição de \*pt:



Valor exibido em tela após a atribuição em \*pt:



No primeiro momento são exibidos os valores da variável x que recebe 10.2 através da própria variável e por conseguinte através do ponteiro \*pt que aponta para essa variável tendo acesso ao seu endereço e conteúdo.

No segundo momento são exibidos os valores da variável x alterado através do ponteiro \*pt que recebe 11.4, o qual conforme já mencionado tem acesso ao seu endereço e conteúdo o que possibilita realizar a alteração do conteúdo dessa variável.

* 1. O que acontece se acrescentarmos os comandos abaixo?

printf("%d\n",&x);

printf("%d\n",pt);



Ao acrescentarmos os comandos podemos mostrar um valor que representa o endereço de memória da variável x, no primeiro passo através da própria variável x e no segundo passo através do ponteiro.

1. Observe que o conteúdo das variáveis se alterou, mas não houve alteração no endereço de memória apresentado nos dois “prints”. Por quê?

R. Porque a variável cont foi alocada no endereço de memória que aparece no print, logo a mudança de conteúdo não altera a “posição” dessa variável na memória do computador e somente o valor contido nela.

* 1. O que acontece se alterarmos o comando do printf para o comando abaixo?

printf("cont = %d; pt = %d; endereco de (cont,pt) = (%d,%d) \n", cont, \*pt, &cont, &pt);



R. Nesse exemplo o printf mostra entre parênteses o endereço da variável cont e o endereço do ponteiro pt.

1. Observer que:
   1. 0 printf - pt não aponta para nenhum endereço de memória. Por que isso gera erro? Como eliminar o problema?

R. Porque o ponteiro não foi inicializado, logo o programa vai buscar as informações solicitadas no printf, mas não irá encontrar nenhum conteúdo e endereço de memória para \*pt e pt. Para eliminar o problema é necessário atribuir alguma variável para o ponteiro antes de dar o printf.

* 1. Por que o conteúdo exibido em cont e \*pt são iguais? (Segundo comando printf)

R. Porque antes do segundo comando printf o ponteiro pt recebe o endereço de memória da variável cont, logo ele passa a ter acesso ao endereço e conteúdo dessa variável.

* 1. Os conteúdos de “num, cont e \*pt” são iguais ou diferentes?

Por um lado sim, pois através do ponteiro pt é possível chegar ao conteúdo da variável cont e depois atribuir indiretamente o mesmo conteúdo a variável num, porém o conteúdo do ponteiro é um endereço de memória também, portanto o conteúdo não totalmente igual.

* 1. Por que o último endereço de memória impresso em tela é diferente dos demais?

R. Porque se trata do endereço de memória da variável num que é diferente do endereço de memória da variável cont que é apresentado nos printfs anteriores.

1. O que você consegue observar na impressão depois dos comandos “pt++” e “pt--”?

R. É possível perceber que no comando pt++ o ponteiro passa a apontar para o próximo endereço de memória mais próximo do último que ele foi associado.

E com o comando p-- ele volta a apontar para o endereço de memória inicial que havia sido associado antes do comando pt++.

1. Os dois laços de repetição exibem o mesmo resultado? Qual é a diferença entre eles? Explique.

R. Sim, apresentam os mesmos resultados. A diferença entre eles é que no primeiro laço for estamos mostrando na tela de forma direta os conteúdos preenchidos no vetor V[] e no segundo laço temos o ponteiro p que recebe o vetor V[] sendo incrementado e passando por cada posição do vetor V[] e mostrando na tela.

1. Qual o objetivo do comando malloc?

R. Alocar espaço de memória de forma dinâmica durante a execução do programa, faz o pedido de memória para o computador e retorna um ponteiro com o endereço do início do espaço de memória que foi alocado.

1. No exemplo abaixo foi criado algum vetor? Qual é a diferença entre ponteiro e vetor?

R. Sim, foi criado um vetor através de um ponteiro, não existe diferença, um ponteiro é um vetor unidimensional, assim como um ponteiro de ponteio é um vetor bidimensional, ou seja, uma matriz.

1. Qual é a diferença entre os comandos “malloc” e “realloc”?

R. O comando malloc serve para alocar espaço de memória de forma dinâmica, de forma dinâmica na memória, o comando realloc serve para alocar e realocar espaço memória durante a execução do programa. Por isso é possível observar que a sintaxe dos comandos são diferentes, Enquanto o comando malloc recebe como parâmetro apenas o tamanho em bytes do tamanho de memória necessário, o comando realloc recebe também o ponteiro que já foi alocado, para receber o outro espaço de memória a ser definido, ou seja, a realocação da memória.