**O que são ponteiros para ponteiros**

Um ponteiro pode guardar o endereço de outro ponteiro?

Pode sim! E isso é muito utilizado em estrutura de dados, chama-se ponteiro de ponteiro

Mas para criar um ponteiro para ponteiro  você precisa aumentar o número de asteriscos na declaração.

seria assim int **\*\***Ponteiro2 = &Ponteiro1;

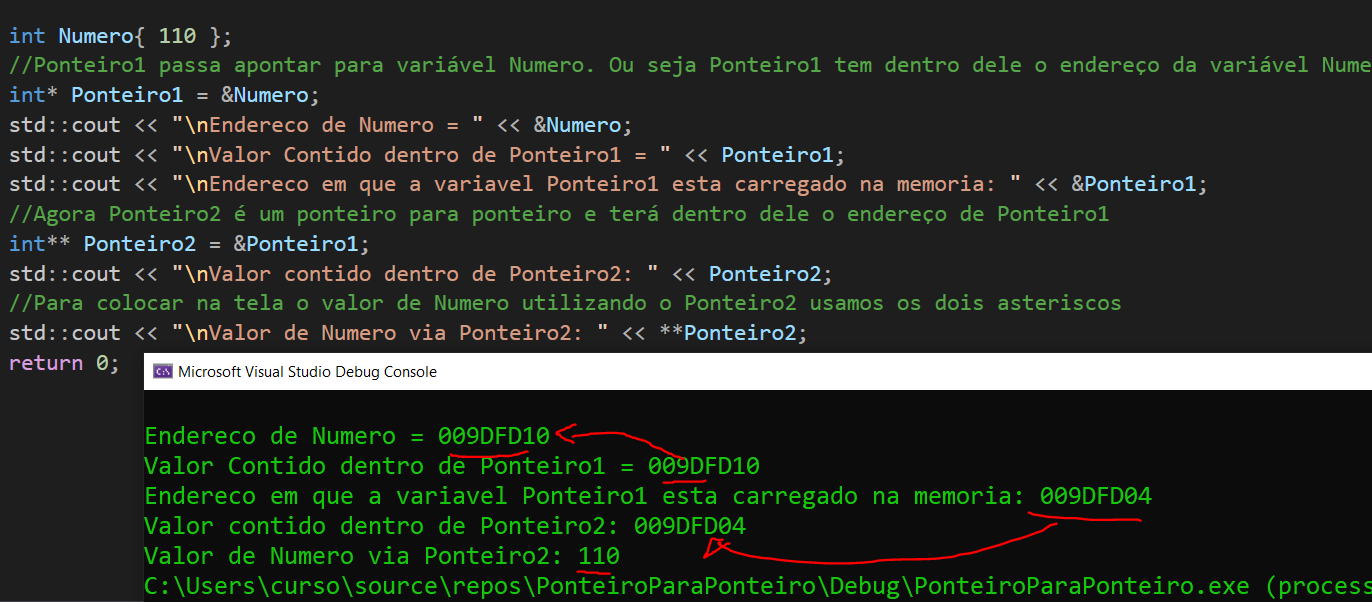
Para acessar o valor desejado apontado por um ponteiro para ponteiro, o operador asterisco deve ser aplicado duas vezes \*\*

Ou seja o valor contido no apontado pelo Ponteiro1 através  do Ponteiro2 será obtido com **\*\*Ponteiro2**

Vamos ver  um exemplo comentado e explicado só que agora usando estas novas notações \*\*

1. #include <iostream>
3. int main()
5. {
6. int Numero{ 110 };
7. //Ponteiro1 passa apontar para variável Numero. Ou seja Ponteiro1 tem dentro dele o endereço da variável Numero
8. int\* Ponteiro1 = &Numero;
9. std::cout << "\nEndereco de Numero = " << &Numero;
10. std::cout << "\nValor Contido dentro de Ponteiro1 = " << Ponteiro1;
11. std::cout << "\nEndereco em que a variavel Ponteiro1 esta carregado na memoria: " << &Ponteiro1;
12. //Agora Ponteiro2 é um ponteiro para ponteiro e terá dentro dele o endereço de Ponteiro1
13. int\*\* Ponteiro2 = &Ponteiro1;
14. std::cout << "\nValor contido dentro de Ponteiro2: " << Ponteiro2;
15. //Para colocar na tela o valor de Numero utilizando o Ponteiro2 usamos os dois asteriscos
16. std::cout << "\nValor de Numero via Ponteiro2: " << \*\*Ponteiro2;
17. return 0;
19. }

Resultado



Se algo que não ficou claro não hesite em perguntar :)

Abraço