
Laboratorio 14

Punteros a punteros

1. Competencias

1.1. Competencias del curso

Conoce, comprende e implementa programas con punteros a punteros en el lenguaje de programación C++.

1.2. Competencia del laboratorio

Conoce, comprende e implementa programas usando punteros a punteros del lenguaje de programación C++.

2. Equipos y Materiales

- Un computador.
- IDE para C++.
- Compilador para C++.

3. Marco Teórico

3.1. Introducción

Puntero a puntero es una variable que contiene la dirección de memoria de un puntero, el cual a su vez contiene la dirección de memoria de un tipo de dato. Se usa ** para definir un puntero a un puntero. Recuerden que un puntero sigue siendo un espacio en memoria, pero en vez de almacenar un valor almacena una dirección.

Estos punteros múltiples tienen el mismo tamaño que un puntero simple, la diferencia sólo está en el número de indirecciones para obtener el dato. El siguiente programa es un ejemplo en el que se manipulan punteros con varios niveles de indirección:

```
#include "stdafx.h"
#include "stdio.h"
#include "iostream"
using namespace std;

int main()
{
    int i;
    int *ptrToi;           /* Puntero a entero */
    int **ptrToPtrToi;     /* Puntero a puntero a entero */

    ptrToPtrToi = &ptrToi; /* Puntero contiene dirección de puntero */
    ptrToi = &i;           /* Puntero contiene dirección de entero */

    i = 10;                /* Asignación directa */
    *ptrToi = 20;           /* Asignación indirecta */
    **ptrToPtrToi = 30;     /* Asignación con doble indirección */

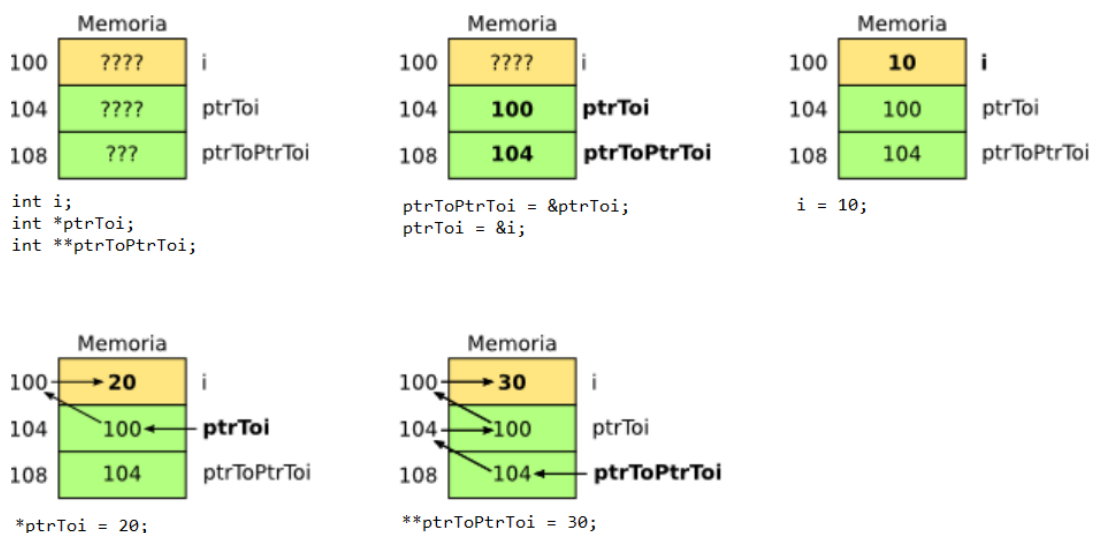
    printf("%d\n", &i);
    printf("%d\n", &ptrToi); /* Dirección de memoria */
    printf("%d\n", &ptrToPtrToi);

    printf("%d\n", i);
    printf("%d\n", ptrToi); /* Referencia de la dirección de memoria */
    printf("%d\n", ptrToPtrToi);

    printf("%d\n", i);
    printf("%d\n", *ptrToi); /* Contenido de La referencia de la DM */
    printf("%d\n", **ptrToPtrToi);

    system("pause");
    return 0;
}
```

La siguiente figura muestra la evolución del valor de todas las variables durante la ejecución del código. Nótese que, aunque no hay ningún valor asignado a la variable *i*, la dirección de cualquier variable sí puede ser obtenida y almacenada sin problemas.



1. Ejercicios

Resolver los siguientes ejercicios planteados:

Implemente un programa utilizando puntero a punteros que gestione los datos de stock de una tienda de ropa, la información a recoger será: código, nombre y precio del producto, asimismo la cantidad en stock. La tienda dispone de 10 productos distintos. El programa debe ser capaz de:

1. Dar de alta un producto nuevo.
2. Dar de baja a un producto. (No hay stock).
3. Buscar un producto por su nombre o código.
4. Modificar el nombre, precio o cantidad de un producto dado.

2. Entregables

Al final estudiante deberá:

1. Compactar el código elaborado y subirlo al aula virtual de trabajo. Agregue sus datos personales como comentario en cada archivo de código elaborado.
2. Elaborar un documento que incluya tanto el código como capturas de pantalla de la ejecución del programa. Este documento debe de estar en formato PDF.
3. El nombre del archivo (comprimido como el documento PDF), será su LAB14_GRUPO_A/B/C_CUI_1erNOMBRE_1erAPELLIDO.

(Ejemplo: LAB14_GRUPO_A_2022123_PEDRO_VASQUEZ).

4. Debe remitir el documento ejecutable con el siguiente formato:

LAB14_GRUPO_A/B/C_CUI_ EJECUTABLE_1erNOMBRE_1erAPELLIDO

(Ejemplo: LAB14_GRUPO_A_EJECUTABLE_2022123_PEDRO_VASQUEZ).

En caso de encontrarse trabajos similares, los alumnos involucrados no tendrán evaluación y serán sujetos a sanción.