



Fundamentos de Programación 101

By Ernie

Ernesto José Canales Guillén

Círculos de estudio UCA

Ciclo Virtual 01/2021



Referencias y Punteros



Direcciones y referencias

Cuando una variable se declara, se asocian tres atributos fundamentales con la misma: su nombre, su tipo y su dirección en memoria.

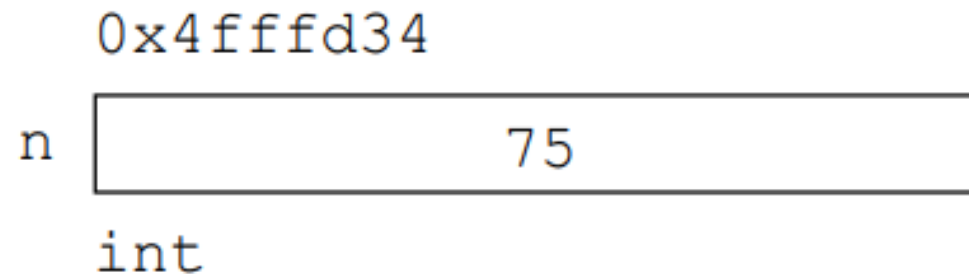
```
int n;          // asocia al nombre n, el tipo int y la dirección de
                // alguna posición de memoria donde se almacena el
                // valor de n
```

0x4fffd34
n
int



Direcciones y referencias

Esta caja representa la posición de almacenamiento en memoria. El nombre de la variable está a la izquierda de la caja, la dirección de variable está encima de la caja y el tipo de variable está debajo en la caja. Si el valor de la variable se conoce, se representa en el interior de la caja.





Direcciones y referencias

- Al valor de una variable se accede por medio de su nombre.

```
cout << n;
```

- A la dirección de la variable se accede por medio del operador de dirección &.

```
cout << &n;
```



¿Y por qué es útil conocer la dirección de la memoria?

Brindan la capacidad de manipular los datos en la memoria de la computadora, lo que puede reducir el código y mejorar el rendimiento.



Referencias

Una referencia es un alias de otra variable. Se declara utilizando el operador de referencia (&) que se añade al tipo de la referencia.

Las variables `n` y `r` tienen la misma dirección de memoria.

```
1 int n = 75;  
2 int& r = n; // r es una referencia para n  
3  
4 cout << "&n = " << &n << ", &r = " << &r << endl; //&n = 0x4fffd34, &r = 0x4fffd34
```



Los dos identificadores `n` y `r` son nombres diferentes para la misma variable.



```
1 int n = 75;  
2 int& r = n; // r es una referencia para n  
3  
4 cout << "n = " << n << ", r = " << r << endl; //n = 75, r = 75
```




Puntero (Apuntador)

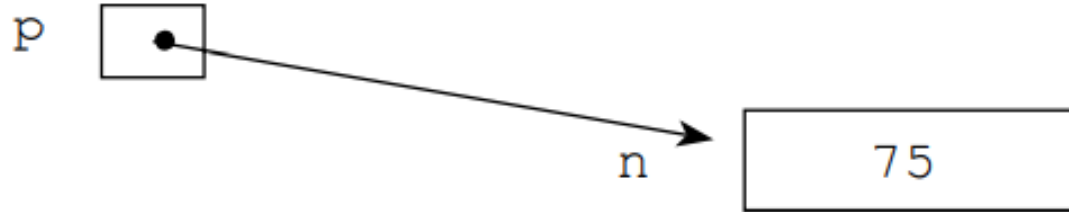
El concepto de punteros tiene correspondencia en la vida diaria. Cuando se envía una carta por correo, su información se entrega basada en un puntero que es la dirección de esa carta.

Los punteros se rigen por estas reglas básicas:

- Un puntero es una variable como cualquier otra;
- Una variable puntero contiene una dirección que apunta a otra posición en memoria;
- En esa posición se almacenan los datos a los que apunta el puntero;
- Un puntero apunta a una variable de memoria.



Puntero (Apuntador)



El valor de un puntero es una dirección. La dirección depende del estado de la computadora en la cual se ejecuta el programa.



Declaración de punteros

- Si se utiliza cuando se declara un puntero: (indirección)
 - *<tipo de dato apuntado> *<identificador de puntero>*
 - Indica que dicha variable se utilizará como un puntero (y no como una variable normal), es decir, su uso será almacenar la dirección en memoria de otra variable.
- Si se utiliza junto a un puntero ya declarado: (desreferencia)
 - `cout << (*puntero) << endl;`
 - `*puntero = 8;`
 - Devuelve el valor contenido (o asigna uno nuevo) en la dirección apuntada por dicho puntero.



Puntero (Apuntador)

```
1 string food = "Pizza";
2 string* ptr = &food;
3
4 cout << food << "\n"; // Output the value of food (Pizza)
5 cout << &food << "\n"; // Output the memory address of food (0x6dfed4)
6
7 //Dereference
8 cout << *ptr << "\n"; // Access the memory address of food and output its value (Pizza)
9
10 //Dereference
11 *ptr = "Hamburger"; // Change the value of the pointer
12
13 //Dereference
14 cout << *ptr << "\n"; // Output the new value of the pointer (Hamburger)
15 cout << food << "\n"; // Output the new value of the food variable (Hamburger)
```



Puntero (Apuntador)

Operador	Propósito
&	Obtiene la dirección de una variable.
*	Declara una variable como puntero.
*	Obtiene el contenido de una variable puntero.



Puntero vs Referencia

- Punteros: un puntero es una variable que contiene la dirección de memoria de otra variable. Es necesario desreferenciar un puntero con el operador `*` para acceder a la ubicación de memoria a la que apunta.
- Referencias: una variable de referencia es un alias, es decir, otro nombre para una variable ya existente. Una referencia, como un puntero, también se implementa almacenando la dirección de un objeto.



- L. J. Aguilar, Programación en C++. Algoritmos, estructuras de datos y objetos, Aravaca (Madrid): McGRAW-HILL, 2006.
- D. Malik, C++ Programming: From Problem Analysis to Program Design, Boston, MA: Cengage Learning, 2003.