

# Apuntadores y direcciones de memoria

jueves, 7 de octubre de 2021 07:07 a. m.

La utilidad de los apuntadores radica en que permiten a un programa ser más dinámico y potente. Debido a que permite el acceso a memoria de manera más eficiente, sin embargo, una mala referencia o gestión a dicha memoria, provocará una salida inesperada en el programa.

Variable

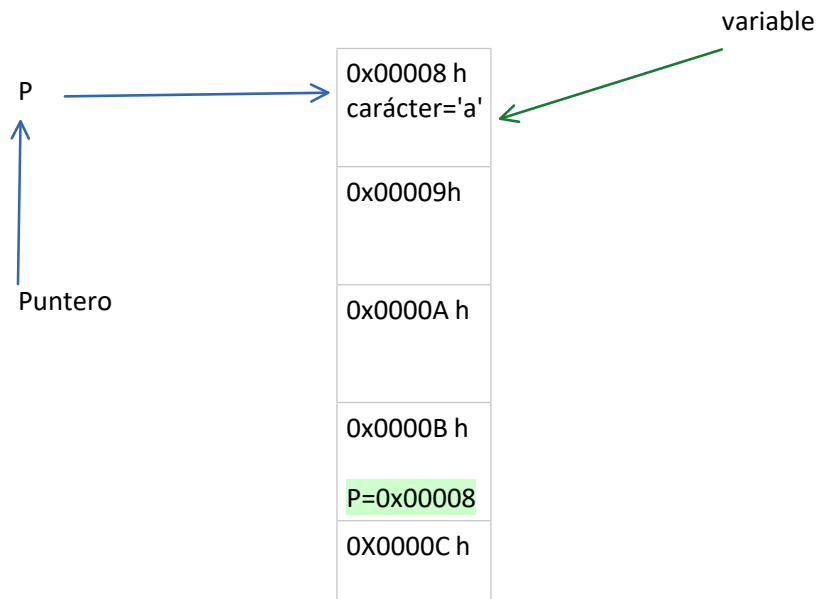
```
int valorEntero=30;
```

0x01	0x02	0x03	0x04	<-----Direcciones de memoria RAM
00000000	00000000	00000000	0001 1110	<-----Dato    valorEntero
int				

Cuando una variable se declara, se le asocian tres atributos fundamentales: identificador, tipo de dato, dirección de memoria

Una variable apuntador o puntero, es una variable que contiene direcciones de memoria de otras variables, es decir almacenan la dirección donde se encuentran los datos asociados a dichas variables

```
char carácter='a'
```



## REGLAS PPARA MANEJO DE APUNTAORES

Los apuntadores en C se rigen por lo siguiente:

1. Un apuntador es una variable como cualquier otra.
2. Una variable apuntador contiene únicamente direcciones de memoria de otras variables en el propio programa.
3. Un apuntador apunta a una variable que se encuentra almacenada en algún lugar físico de la memoria.
4. En las posiciones apuntadas por cualquier apuntador, debe contener datos.
5. El tipo de dato al que se apunta debe ser el mismo que fue declarado para el apuntador.

## Operadores para utilizar apuntadores

&	Operador de dirección	Obtener la dirección de la variable a apuntar.
*	Operador de indirección	Obtener el contenido (el dato) que se encuentra en la localidad de memoria a la que apunta el puntero.
*	Operador de declaración de apuntador	Declarar una variable apuntador.

## Sintaxis

//Declaración de apuntadores

```
<tipo de dato>* identificadorPuntero;
```

//Declaración e inicialización de apuntadores:

```
<tipo de dato>* identificadorPuntero=NULL;
```

```
<tipo de dato>* identificadorPuntero=&variable; //variable anteriormente fue declarada
```

//Sintaxis para lectura/escritura a través de apuntadores

```
<tipo de dato> identificadorVariable;
```

```
<tipo de dato> identificadorVariable2;
```

```
<tipo de dato>* identificadorPtr=&identificadorVariable;
```

```
identificadorVariable2=*identificadorPtr; //Lectura
```

```
*identificadorPtr=56; //Escritura
```

**\*\*Una mala asignación de memoria produce un memoryViolationException**

Ejemplo en C

```
#include<stdio.h>
```

```
void main(){
```

```
    char varC='j';
```

```
    char* ptrVar=&varC;
```

```
    puts("Datos de la variable contador:");
```

```
    printf("Contenido de la variable varC: %c\n",varC);
```

```
    printf("Direccion de memoria de la variable varC: %p\n\n",&varC);
```

```
    puts("Datos de la variable apuntador ptrVar:");
```

```
    printf("Contenido de la variable ptrVar: %p\n",ptrVar);
```

```
    printf("Direccion de memoria de la variable ptrVar: %p\n",&ptrVar);
```

```
    printf("Dato al cual se apunta con el apuntador: %c\n",*ptrVar);
```

```
    *ptrVar='L';
```

```
    printf("Contenido de la variable varC: %c\n",varC);
```

}