

# PUNTEROS EN LENGUAJE “C” ALGORITMIA Y PROGRAMACIÓN

Rodríguez Vélez Diego Leandro - 201959822

Universidad del valle

Sede – zarzal

[diego.leandro.rodriguez@correounivalle.edu.co](mailto:diego.leandro.rodriguez@correounivalle.edu.co)

## ❖ ABSTRACT

The C language is of the structured language type such as Pascal, Fortran, Basic. Its instructions are very similar to other languages including statements like if, else, for, do and while .... Although C is a high-level language (since it is structured and has sentences and functions that simplify its operation) we have the possibility of programming at a low level (as in the Assembler, touching registers, memory, etc.). To simplify the operation of the C language, it includes libraries of functions that can be included by referring to the library that includes them, that is, if we want to use a function to clear the screen, we will have to include in our program the library that has the function to clear the screen.

## ❖ INTRODUCCIÓN

Un puntero es una variable que contiene la dirección de memoria de otra variable. Los punteros permiten código más compacto y eficiente; utilizándolos en forma ordenada dan gran flexibilidad a la programación.

La dirección de memoria de una variable se obtiene con el operador unario &. El operador unario \* permite la desreferencia de un variable puntero; es decir, permite el acceso a lo apuntado por un puntero.

### Ejemplo:

```
int x = 1, y = 2;  
int *ip; /* ip es un puntero a int */  
ip = &x; /* ip apunta a x */  
y = *ip; /* a y se le asigna lo apuntado por  
ip */  
*ip = *ip + 3; /* incrementa lo apuntado  
por ip, x */  
ip = NULL; /* ip apunta a direcc. Especial  
(nada) */
```

La sintaxis de la declaración de un puntero imita a las expresiones en que la variable puede utilizarse; cada puntero apunta a un tipo específico de datos (con la excepción del puntero genérico void).

La dirección especial NULL (o cero) indica que un puntero apunta a “nada” y es usada como centinela para establecer el fin de estructuras autoreferenciadas. Además,

esta es retornada por la función de asignación de memoria, malloc, en el caso de no haber suficiente memoria. El operador new, en cambio, aborta el programa cuando no tiene más memoria que dar.

## ❖ DESARROLLO

### Reglas de uso

-El símbolo \* se usa para definir un tipo puntero.

-El símbolo & se usa para indicar la dirección de una variable.

-El símbolo \* se usa para acceder al valor de un puntero.

-El símbolo & se usa en la invocación de funciones, de forma que se pone delante de los parámetros de E/S

-Cuando el parámetro es de E/S en el prototipo y cabecera se indica con un tipo puntero, esto es, poniendo un \* delante del nombre.

-En el cuerpo d la función los argumentos que son funciones se usan con un \* delante.

-Los & solo se usan en el programa principal.

-Los \* solo se usan en las funciones.

### Procedimiento

Tengo 2 variables y voy a intercambiar sus valores

### Código

```
#include <stdio.h>

void intercambia(int *, int *);

using namespace std;

int main (void) {
    int a = 8, b = 7;
    printf ("antes: a = %d b = %d\n", a, b);
    intercambia(&a, &b);
    printf("Despues: a = %d b = %d\n", a,
b);
}
```

```
void intercambia(int *a, int *b){
    int D;
    D = *a;
    *a = *b;
    *b = D;
}
```

Haciendo que los parámetros formales de la función intercambia no reciban una copia de los valores a y b, sino las direcciones de memoria de a y b. de esta manera, las funciones intercambia tendría

acceso a poder cambiar los valores de a y b.

### **Al ejecutarse debería dar como salida**

Antes: a = 7 b = 8

Después: a = 8 b = 7

### **❖ CONCLUSIÓN**

Accediendo al contenido de una dirección de memoria utilizando punteros, para lenguaje C proporciona un operador que actuando sobre una variable dirección de memoria devuelve su contenido, este operador es \* de esta forma devuelve el valor almacenado en la dirección para la función intercambia.

### **❖ REFERENCIAS**

<http://lsi.vc.ehu.es/pablogn/docencia/manuales/C/Punteros%20en%20C.html>

[https://www.ecured.cu/C\\_\(lenguaje\\_de\\_programaci%C3%B3n\)#:~:text=Lenguaje%20de%20programaci%C3%B3n%20en%20C.,son%20Pascal%20Fortran%20Basic.](https://www.ecured.cu/C_(lenguaje_de_programaci%C3%B3n)#:~:text=Lenguaje%20de%20programaci%C3%B3n%20en%20C.,son%20Pascal%20Fortran%20Basic.)

<https://www.youtube.com/watch?v=RRCh4eFZ7Sg>