Universidad Nacional Autónoma de México
Facultad de Ingeniería

Asignatura: Estructura de Datos y Algoritmos I

Actividad 5: Apuntadores en C

Alumno: Miranda González José Francisco

Fecha: Miércoles 24 de Marzo del 2021

¿Qué es un apuntador?

Un apuntador es una variable que contiene la dirección de una variable, es decir, hace referencia a la localidad de memoria de otra variable. Debido a que los apuntadores trabajan directamente con la memoria, a través de ellos se accede con rapidez a un dato.

La sintaxis para declarar un apuntador y para asignarle la dirección de memoria de otra variable es, respectivamente:

TipoDeDato *apuntador, variable;

apuntador = &variable;

La declaración de una variable apuntador inicia con el carácter *. Cuando a una variable le antecede un ampersand, lo que se hace es acceder a la dirección de memoria de la misma (es lo que pasa cuando se lee un dato con scanf).

Los apuntadores solo pueden apuntar a direcciones de memoria del mismo tipo de dato con el que fueron declarados; para acceder al contenido de dicha dirección, a la variable apuntador se le antepone *.

Bibliografía: MADO-17_FPv2 Manual de practicas Fundamentos de programación

¿Dónde se aplica?

Los apuntadores (punteros) tienen varias aplicaciones, incluyendo:

Crear códigos eficientes y rápidos

Proporcionan asignación de memoria dinámica

Hacen expresiones compactas y concisas

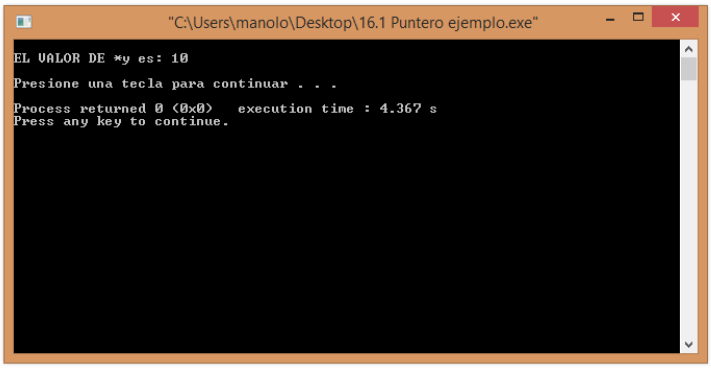
Protegen datos pasados como parámetros a una función

Proporcionan la capacidad de pasar estructuras de datos mediante un puntero sin ocasionar un exceso de código conocido como “overhead”

Bibliografía: https://dignal.com/importancia-de-utilizar-punteros-en-lenguaje-c/

Ejemplo de uso en lenguaje C

```c
//EJEMPLO DE USO DEL PUNTERO

#include <stdio.h>

int main(){

    int x,*y;

    x=10;
    y=&x; //En "y" se almacena el valor de la direccion de x
    printf("\n");
    printf("EL VALOR DE *y es: %i \n",*y);
    printf("\n");
    //*y esta señalando al valor de x
    //Los * señalan
    //Los & son la direccion

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




```c
//EJEMPLOS DE USO DEL PUNTERO
//INTERCAMBIAR LOS VALORES DE DOS VARIABLES CON UNA FUNCION

#include <stdio.h>

void intercambio(int *a, int *b);//funcion void intercambio definida antes de la principal

int main(){

    int x,y;//definimos 2 variables tipo entero

    printf("\n\n");
    printf("VALOR DE X : ");
    scanf("%i",&x);//nos regresara el valor introducido de x
    printf("VALOR DE Y : ");
    scanf("%i",&y);//nos regresara el valor introducido de y

    intercambio(&x,&y);//manda a llamar a la funcion intercambio para ejecutarse en la principal
    //se usa para el espacio de memoria.

    printf("\n");
    printf("AHORA EL VALOR DE X ES: %i ",x);
    printf("\n\n"); //los valores de las varibales se intercambian
    printf("AHORA EL VALOR DE Y ES: %i ",y);
    printf("\n");

    system ("pause");
    return 0;
}

void intercambio(int *a, int *b){//se realiza todo el proceso en la funcion
    //void intercambio y se ejecutara el la principal "main"
    // a=&x y con lo tanto *a = x
    // b=&y y con lo tanto *b = y

    int aux;//en aux guardamos un valor para no tener cambios

    aux=*a;
    *a=*b;
    *b=aux;
}
```

```
VALOR DE X : 15
VALOR DE Y : 30

AHORA EL VALOR DE X ES: 30

AHORA EL VALOR DE Y ES: 15

Presione una tecla para continuar . . .

Process returned 0 (0x0)   execution time : 81.996 s
Press any key to continue.
```

: