UNIVERSIDAD SAN SEBASTIAN

VOCACIÓN POR LA EXCELENCIA

Paradigmas de Programación

UNIVERSIDAD SAN SEBASTIAN

VOCACIÓN POR LA EXCELENCIA

Más Tipos de Datos

UNIVERSIDAD SAN SEBASTIAN
VOCACIÓN POR LA EXCELENCIA

Punteros

- La memoria de un computador puede modelarse como una gran colección de cajas donde se almacenan datos, cada caja identificada por un número único.
  - Cada uno de los números es una **dirección** de memoria.

- Una variable en un lenguaje de programación
  - Tiene un tipo y un nombre, asignados por un programador.
  - Al ejecutar el programa, a ese nombre se le asigna una dirección en la memoria de la máquina.

UNIVERSIDAD SAN SEBASTIAN
VOCACIÓN POR LA EXCELENCIA

Punteros

- Los punteros son un mecanismo que permite:
  - Acceder y manipular datos directamente en la memoria.
    - Esto puede mejorar la eficiencia de los programas, especialmente cuando se trabaja con grandes cantidades de datos.
  - Pasar parámetros por referencia a funciones
    - Esto permite que una función modifique el valor de una variable que se encuentra fuera de su ámbito.
  - Crear estructuras de datos dinámicas:
    - Los punteros permiten crear listas enlazadas, árboles y otras estructuras que pueden crecer o decrecer durante la ejecución del programa.

UNIVERSIDAD SAN SEBASTIAN
VOCACIÓN POR LA EXCELENCIA

Punteros

- El contenido usual de una caja es un dato a manipular por el programa.
  - En otras palabras: una variable “normal” almacena el valor del dato.
- En el caso de los punteros, una caja contiene la dirección de otra caja (direccionamiento indiirecto).
-

UNIVERSIDAD SAN SEBASTIAN
VOCACIÓN POR LA EXCELENCIA

Punteros

- Dos operadores asociados:
  - Obtener la dirección de una variable: **&**
    - Si `p` es una variable, `&p` obtiene su dirección (referencia).
    - La referencia puede ser almacenada en una variable para uso posterior.
  - Usar el contenido de un puntero como dirección de una variable: *****
    - Para una variable `q`, `*q` representa el contenido de la memoria apuntado por `q` (dereferencia).
- Los punteros deben declararse con el tipo de dato al que apuntan.

UNIVERSIDAD SAN SEBASTIAN
VOCACIÓN POR LA EXCELENCIA

Punteros en C

```c
#include <stdio.h>

int main() {
    int numero = 10;
    int *puntero;              // Declaración de un puntero a entero

    puntero = &numero;         // Asignación de la dirección de 'numero' al puntero

    printf("Valor de numero                : %d\n", numero);
    printf("Dirección de numero            : %p\n", &numero);
    printf("Valor del puntero              : %p\n", puntero);
    printf("Valor apuntado por el puntero : %d\n", *puntero);

    *puntero = 20;             // Modificación de 'numero' a través del puntero

    printf("Nuevo valor de numero: %d\n", numero);

    return 0;
}
```

UNIVERSIDAD SAN SEBASTIAN
VOCACIÓN POR LA EXCELENCIA

Punteros explicación

- Se declara una variable entera llamada 'numero' y se le asigna el valor 10.
- Se declara un puntero a entero llamado 'puntero'.
- Se utiliza el operador '&' para obtener la dirección de memoria de 'numero' y se asigna esa dirección al puntero.
- Se imprime el valor de 'numero', su dirección de memoria, el valor del puntero (que es la dirección de 'numero') y el valor apuntado por el puntero (que es el valor de 'numero').
- Se utiliza el operador '*' para acceder al valor apuntado por el puntero y se modifica ese valor a 20.
- Se imprime el nuevo valor de 'numero', que ha sido modificado a través del puntero.

UNIVERSIDAD SAN SEBASTIAN
VOCACIÓN POR LA EXCELENCIA

Punteros: Uso en lenguajes:

- C y C:
  - Permiten una manipulación de memoria muy cercana al hardware
  - Ideales para el desarrollo de sistemas operativos y software de alto rendimiento.
  - Rust, también utiliza el concepto de punteros, pero de una forma mucho más segura y controlada.
- Java o Python:
  - La gestión de memoria es automática (a través de recolectores de basura), lo que oculta parcialmente los punteros al programador.
  - Sin embargo, comprender los punteros sigue siendo valioso para comprender cómo funciona la memoria y para optimizar el rendimiento.

Punteros: Aritmética

- En el fondo, un puntero no es más que un valor entero.
  - Por ello, se les puede asignar valores fijos.
    - En general, mala política. Por ejemplo, el valor 0 se refiere a un área de memoria generalmente protegida => *segmentation fault*.
  - Se puede hacer aritmética con ellos.
    - A los arrays se les asigna una dirección inicial. Sus componentes se ubican a través de sumas (restas) que, en el fondo, no son más que cálculos con direcciones

UNIVERSIDAD SAN SEBASTIAN
VOCACIÓN POR LA EXCELENCIA

Punteros: Riesgos y Consideraciones

- Errores de Segmentación: Intentar acceder a una ubicación de memoria no válida puede provocar un error de segmentación (*segmentation fault*), que generalmente aborta la ejecución del programa.
- Punteros Colgantes: Un puntero colgante (*dangling pointer*) apunta a una ubicación de memoria que ya no está asignada, lo que puede provocar errores impredecibles.
- Fugas de Memoria: Si la memoria asignada dinámicamente no se libera correctamente, se produce una fuga de memoria (*memory leak*), lo que agota los recursos del sistema.

UNIVERSIDAD SAN SEBASTIAN

VOCACIÓN POR LA EXCELENCIA

Otros tipos de datos

UNIVERSIDAD SAN SEBASTIAN
VOCACIÓN POR LA EXCELENCIA

Otros tipos de datos

- Enumeración
  - Nombre simbólico para un conjunto de valores.
    - Las enumeraciones se tratan como tipos de datos
    - se pueden usar a fin de crear conjuntos de constantes para su uso con variables y propiedades.
- Conjunto
- Bit, Byte

UNIVERSIDAD
SAN SEBASTIAN
VOCACIÓN POR LA EXCELENCIA

FIN