

Programación en Python y C++

Apuntadores

Rafael De Luna Loredó

Facultad de Ciencias, UASLP



Índice

① Introducción

② Operadores

③ Aritmética de
apuntadores

④ Arreglos y
apuntadores

⑤ Apuntadores a
funciones

⑥ Referencias

① Introducción

② Operadores

③ Aritmética de apuntadores

④ Arreglos y apuntadores

⑤ Apuntadores a funciones

⑥ Referencias

Introducción

- Hacen referencia a direcciones de memoria
- Pueden apuntar a direcciones de memorias de variables y funciones
- Su tipo de dato puede cambiar ¹
- También pueden apuntar a arreglos o matrices

¹solo cuando se declaran de tipo void

Ventajas

Debajo de muchos de los lenguajes actuales se hace uso de los apuntadores, sin embargo no permiten hacer un uso mas profundo de ellos por lo que podríamos terminar con código mas pesado o ineficiente, los apuntadores nos permiten:

- Administrar el uso de memoria
- Crear código mas eficiente y rápido
- Permiten evitar o disminuir el overhead ²
- Iterar eficientemente sobre estructuras de datos

²Exceso de tiempo de ejecución y memoria

Desventajas

Aunque los apuntadores son una poderosa herramienta también implican ciertos riesgos o desventajas

- Mala administración de la memoria
- Posible acceso a partes críticas del sistema
- Requiere una previa planeación sobre su implementación en el código

① Introducción

② Operadores

③ Aritmética de
apuntadores

④ Arreglos y
apuntadores

⑤ Apuntadores a
funciones

⑥ Referencias

Operador de dirección &

Nos permite obtener la dirección de memoria de cualquier variable, sea un apuntador o no, hay que tener cuidado al usarlos, ya que podríamos terminar mostrando direcciones importantes del sistema

Operador de indirección *

Nos permite obtener el contenido de la dirección a la que el apuntador hace referencia, es decir nos muestra un valor o dato, aquí también hay que tener cierto cuidado o podríamos terminar mostrando valores importantes del sistema.

① Introducción

② Operadores

③ Aritmética de
apuntadores

④ Arreglos y
apuntadores

⑤ Apuntadores a
funciones

⑥ Referencias

Suma y resta

Podemos hacer sumas o restas en un apuntador como en cualquier otra variable sin embargo debemos considerar que en realidad lo que estaríamos haciendo es cambiar la dirección de memoria a la que se apunta. En el caso de la suma estaríamos avanzando hacia una dirección de memoria que se encuentre mas adelante, según el tipo de dato al que apunte se este apuntando. Para la resta estaríamos retrociendo hacia una dirección de memoria que se encuentre mas atrás

① Introducción

② Operadores

③ Aritmética de
apunadores

④ Arreglos y
apunadores

⑤ Apunadores a
funciones

⑥ Referencias

Relación entre arreglos y apuntadores

Dado que un arreglo es una variable en la que se reserva un n tamaño de memoria, mediante apuntadores podemos acceder a la memoria de los diferentes índices del arreglo, también mediante apuntadores podemos crear arreglos, la diferencia es que con apuntadores el espacio de memoria se reserva en tiempo de ejecución (memoria dinámica)

Matrices con apuntadores

En realidad una matriz con apuntadores, es un arreglo de apuntadores, donde cada apuntador hace referencia a una dirección de memoria, tal vez suene más complicado pero la creación y recorrido es de la misma manera que en una matriz, solo que aplicando aritmética de apuntadores.

Ventajas respecto a arreglos o matrices

- No es necesario conocer el tamaño
- Se reserva el tamaño exacto
- Se puede liberar la memoria en caso de ya no utilizarla
- Se puede redimensionar el tamaño

Desventajas respecto a arreglos o matrices

- A veces es complicado de implementar
- Para redimensionar el tamaño es necesario crear una copia
- No se pueden eliminar elementos intermedios
- Cuando se libera memoria se borra todo el arreglo o matriz
- Se tiene que volver a crear el espacio de memoria en caso de querer redimensionar

① Introducción

② Operadores

③ Aritmética de
apuntadores

④ Arreglos y
apuntadores

⑤ Apuntadores a
funciones

⑥ Referencias

Apuntadores a funciones

- Hay que saber el tipo de dato que retorna la función
- Hay que saber los parámetros de la función
- Pueden hacer referencia a cualquier función que cumpla con los primeros puntos
- Podemos crear código más limpio y legible
- En principio puede ser difícil de implementar

① Introducción

② Operadores

③ Aritmética de
apuntadores

④ Arreglos y
apuntadores

⑤ Apuntadores a
funciones

⑥ Referencias

Referencias I