

CLASE MARTES 12 SEPTIEMBRE ASTROINFORMÁTICA

memoria ram aleatoria, no se limpia constantemente

(limpiarla x____x)

inicialización determinar qué valor quiero en una variable

se pueden juntar como x ej: long unsigned int

más rango para que el entero pueda abarcar

enteros: simples 4 bites o double de 8 bits

C introduce "struct" dato especial para la creación de nuevos tipos de datos,

permite tomar x ej 2 float y combinarlos y hacer un tipo de dato nuevo

esto permitió la orientación a objetos

operadores matemáticos de C se rigen x las mismas reglas de papomudas

joasjoas

float si lo meto a int el código redondeará a entero el resultado

además tiene operadores lógicos: and, or, not o de comparación igual q python:

igual, distinto, mayor menor igual que

flujo de código: trayectoria

control de flujo: agregar decisiones en la trayectoria del algoritmo

pseudo código

desafiarse: programar un triángulo equilátero

condicionales iguales a python

if acompañada de (donde meto mi condición)

RECORDAR LÓGICA

x ej: meter un or donde debería haber un and

elif: if anidado

si la instrucción es única no necesito {}

cerrar { en la misma indentación }

if se puede usar sin else

control de flujo: iteraciones: loops

C introduce 3 tipos de loops

1 for y 2 while

for necesita de 3 componentes: **inicio**: se pueden dar instrucciones; **condición** q

se tiene q cumplir para q el loop se siga iterando; instrucción final de cada **paso**.

i++ que toma algo y le suma 1

python maneja 2 tipos de string:

C no define strings, uno lo pone literalmente con %d

while: while con condición y después pongo las condiciones

****es i++**** no i solita, al final de la diapo

el otro while es: do while

dif: while solito no mete la variable si no cumple la condición al tiro, el do while si

permite que entre la variable al do pero en el while lo tiran pajueraaaaaa

antes de la instrucción: prefijo o después de la instrucción postfijo

operadores ternarios: `?:`, establece condición y tener 2 posibles resultados

operadores son funciones que reciben parámetros

punto es un operador de membresía, para revisar miembros de

arrays y memoria estática

`[]` : operador de indexación: permite tomar elementos de

aquí `[]` es para ponerle el tamaño del array

Arrays en C: Colección de elementos, donde cada elemento tiene un tipo de datos y todos son del mismo tipo

La declaración del array se hace con parentesis cuadrados.

En caso de array estaticos, se reserva la memoria de manera consecutiva, por lo cual es muy eficiente

C busca que los elementos estén consecutivos

arrays están indexados partiendo desde 0

arrays multidimensionales: no hay límite

eficiencia: múltiples cálculos q estén en una sola línea

for en una lista es lento !!! o en cualquier objeto que tenga iteradores en python, pq está saltando todo el tiempo

no hacer un for dentro de un for

C no tiene iteradores, ni tampoco tiene saltos

stdout : pantalla

puntero

en C son la característica central para acceder a las propiedades de bajo nivel y al manejo versátil de la memoria

es una variable en C cuyo contenido es una dirección de memoria del sistema

almacenar la dirección de memoria

RAM : stack : almacena toda la info implícita estática del programa: x ej nombre de funciones o así

stack es una lista

entonces reservo un espacio para mi int algo

esa es mi dirección, y con el puntero indico esa dirección, variable que almacena el espacio

crear puntero `*` y para decir cual es la memoria de algo `&`

`&` da el espacio físico en hexagesimal

el `%p` variable tipo puntero

pa es la variable puntero

`*pa` es el contenido

