



Unidad 7

ciclos

Ciclos

- Los ciclos son también llamados iteraciones , se usan en programación para ejecutar el mismo conjunto de instrucciones hasta que se cumpla cierta condición.

Tres instrucciones diseñadas para los ciclos

- while
- do-while
- for

Ciclo while

- While (expresión)
 - Instrucción;
- While (expresión)
- {
 - Instrucción1;
 - Instrucción2;
 -
 - Instrucción n;}

Devuelve cero si la condición es falsa y no se evalúa la expresión

Uso de un ciclo while

- Escribe un programa utilizando un ciclo while para leer continuamente y desplegar un carácter después de introducirlo siempre y cuando no sea el carácter 'x'.
- Escribe las indicaciones apropiadas en pantalla.
- Tip inicializa la variable
 - `c=' '`
 - `c= getc(stdin)`
 - `putchar(c)`

do-while

- Do
- {
 - Instrucción 1
 - Instrucción 2
 - Instrucción 3
 - }
 - While (expresión);
- Se garantiza que se efectúen la(s) instruccione(s) al menos 1 vez antes de evaluar la expresión.
- Note que al final de do-while termina con ; y en while no!

Uso de do-while

- Escriba un programa que despliegue los caracteres así como sus respectivos valores numéricos, mediante un ciclo do-while que repite la impresión y el incremento.
- Tip A es 65

Ciclos usando la instrucción for

- for (expresión 1; expresión 2 ; expresión 3)
 - Instrucción

- for (expresión 1; expresión 2 ; expresión 3)
 - {
 - Instrucción 1
 - Instrucción 2
 - Instrucción 3
 - ..
 - }

Uso del ciclo for

- Reescriba el programa de conversión del 0 al 15 a números hexadecimales usando un ciclo for.
- Tip: `for (i=0; i<16; i++)`

Instrucción nula

- La instrucción for (while) no tienen no llevan ; al final, es decir
- for (i=0; i<16; i++)
- Sum+= I;
- Si se pone ; entonces for (i=0; i<16; i++); esto sería igual a
- for (i=0; i<16; i++)
- ;
- Donde ; es la instrucción nula. Así que el ciclo iterar pero sin hacer ninguna instrucción.

IMPORTANTE

- Si por accidente se coloca ;
- for (i=0; i<16; i++) ;
- Sum+= I;
- Esto e
- for (i=0; i<16; i++)
- ;
- Sum+= I
- La instrucción sum nunca se realiza dentro del ciclo.

Uso de expresiones complejas en una instrucción for

- `for (i=0, j=10; i!=j; i++, j--)`
- `{instrucciones}`

- Escriba un programa que sume $i + j$

- OTRO EJEMPLO
- `for (i=0, j=1; i<8; i++, j++)`
- `{instrucciones}`
- Escriba un programa que reste $j - i$

Ciclos anidados

- Contiene un ciclo externo y uno interno
- Escribe un programa que muestre las iteraciones de los dos ciclos
- Tip:
- for (..)
 - printf(“Inicio ciclo externo: iteración #..
 - for (..)
 - printf(“Inicio ciclo interno: iteración #..
 - printf(“Fin ciclo externo: iteración #..

importante

- No confundir
- for (i=0, j=1; i<=8; i++, j++)
- Con
- for (i=0, j=1; i<8; i++, j++)