

Resumen Ciclos

Profesor:
Julián Andrés Castillo



Estructuras de programación (secuenciales, lógicas y repetitivas)

Agenda



1. Estructuras algorítmicas repetitivas
2. Contadores
3. Acumuladores
4. Ciclo Para – (FOR)
5. Ciclo mientras – (WHILE)
6. Ciclos anidados
(Estructuras de repetición anidadas)

1. Estructuras algorítmicas repetitivas

Estructuras algorítmicas selectivas

Es muy común encontrar en la práctica algoritmos cuyas operaciones se deben ejecutar un número repetido de veces.

El conjunto de instrucciones que se ejecuta repetidamente se llama ciclo.

Todo ciclo debe terminar de ejecutarse luego de un número finito de veces, por lo que es necesario en cada iteración del mismo, evaluar las condiciones necesarias para decidir si se debe seguir ejecutando o si debe detenerse.

En todo ciclo, siempre debe existir una condición de parada o fin de ciclo.

Estructuras algorítmicas selectivas

Las estructuras lógicas repetitivas se utilizan cuando necesitamos realizar un algoritmo en el cual haya que repetir un proceso un número n de veces, las estructuras repetitivas se pueden mezclar con las estructuras selectivas si el problema lo necesita, también se hará uso de expresiones lógicas y operadores relacionales.

2. Contadores

Contadores

La construcción de un contador es una de las técnicas más comunes en la realización de algoritmos.

Es una variable en la memoria que se incrementará en una unidad cada vez que se ejecute el proceso.

El contador se utiliza para llevar la cuenta de determinadas acciones que se pueden solicitar durante la resolución de un problema.

En las instrucciones de preparación se realiza la inicialización del contador o contadores. La inicialización consiste en poner el valor inicial de la variable que representa al contador.

Generalmente se inicializa con el valor 0 para operaciones de suma y en 1 para operaciones de multiplicación.

3. Acumuladores

Acumuladores

Un acumulador es una variable en la memoria cuya misión es almacenar cantidades variables.

Se utiliza para efectuar sumas sucesivas. La principal diferencia con el contador es que el incremento o decremento de cada suma es variable en lugar de constante como en el caso del contador.

2. Contadores

y

3. Acumuladores

Contadores y Acumuladores

El ejemplo mas sencillo para explicar contadores y acumuladores detalla el promedio de n números ingresados a un sistema.

Un contador debe llevar cuenta de la cantidad de números ingresados y el acumulador debe llevar la suma de todos los valores ingresados, al final del procedimiento realizamos la siguiente operación,

$$\text{promedio} = \frac{\text{acumulador}}{\text{contador}}$$

4. Ciclo Para – (FOR)

Ciclo Para – (FOR)

Para, también llamado FOR, es la estructura algorítmica adecuada para utilizar *en un ciclo que se ejecutará un número definido de veces*. El número de veces del ciclo no depende de las proposiciones dentro de el mismo. El número de veces se obtiene del planteamiento del problema o de una lectura que indica que el número de iteraciones se debe realizar para n ocurrencias

Ciclo Para – (FOR)

Para los ciclos Para (FOR) se definen las siguientes variables:

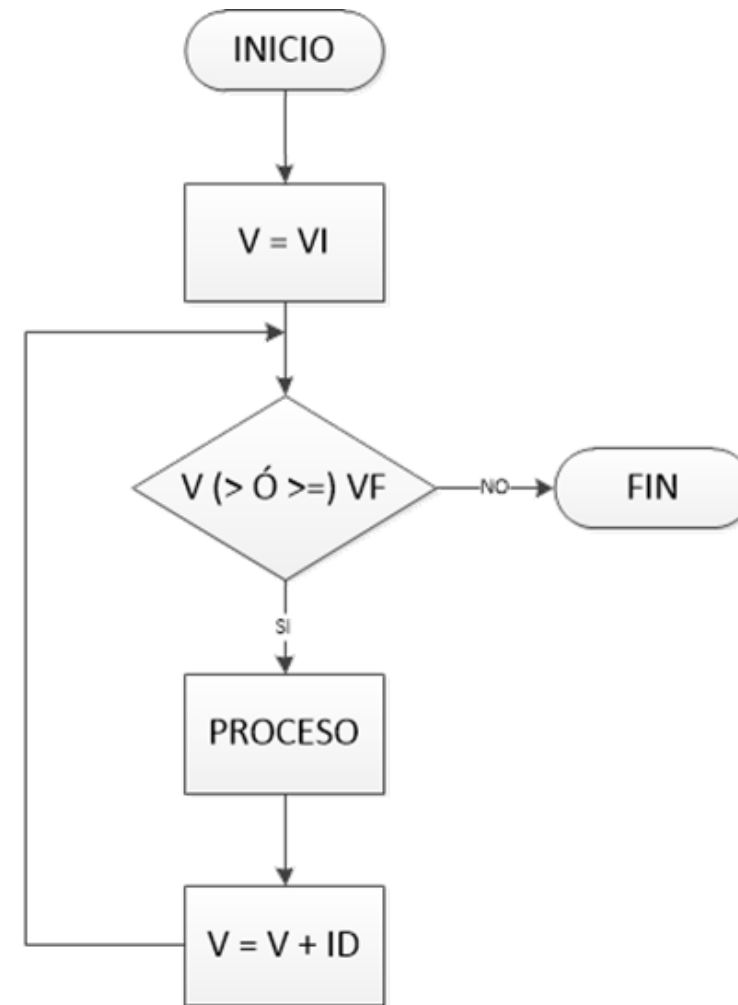
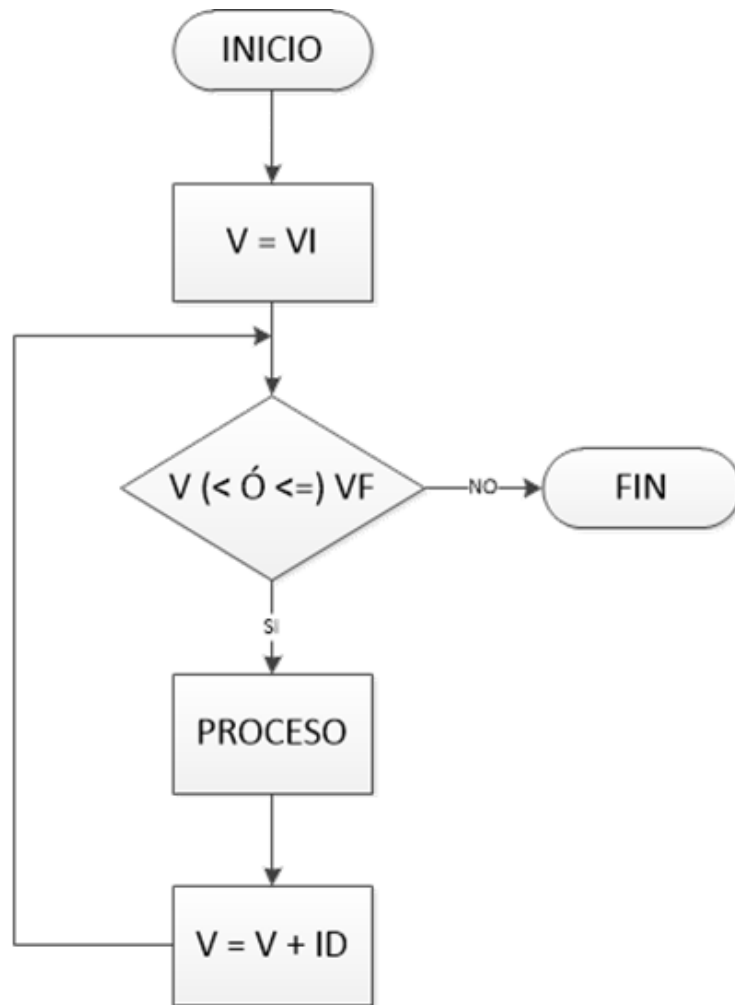
- V: es la variable de control del ciclo.
- VI: es el valor inicial.
- VF: es el valor final.
- ID: es el incremento o decremento, según sea la estructura ascendente o descendente.

V, contador del ciclo normalmente es representado por las letras “i, j, k”

El ciclo se ejecuta mientras V es menor, menor igual, mayor o mayor igual al valor de VF, el valor de V se incrementa o disminuye en cada iteración. Cuando V supera el valor de VF el ciclo se detiene.

PARA V desde VI hasta VF
Cuerpo / Operaciones
FIN PARA

Ciclo Para – (FOR)



5. Ciclo mientras – (WHILE)

Ciclo mientras – (WHILE)

Mientras, también llamado WHILE, es la estructura adecuada para utilizar en *un ciclo cuando no sabemos el número de veces que este se ha de repetir.*

Dicho número depende de las proposiciones dentro del ciclo.

En el ciclo mientras (WHILE) se distinguen dos partes:

- Ciclo: Conjunto de instrucciones que se ejecutarán repetidamente.
- Condición de terminación: La evaluación de esta condición permite decidir cuando finalizará la ejecución del ciclo. La condición se evalúa al inicio del mismo.

Ciclo mientras – (WHILE)

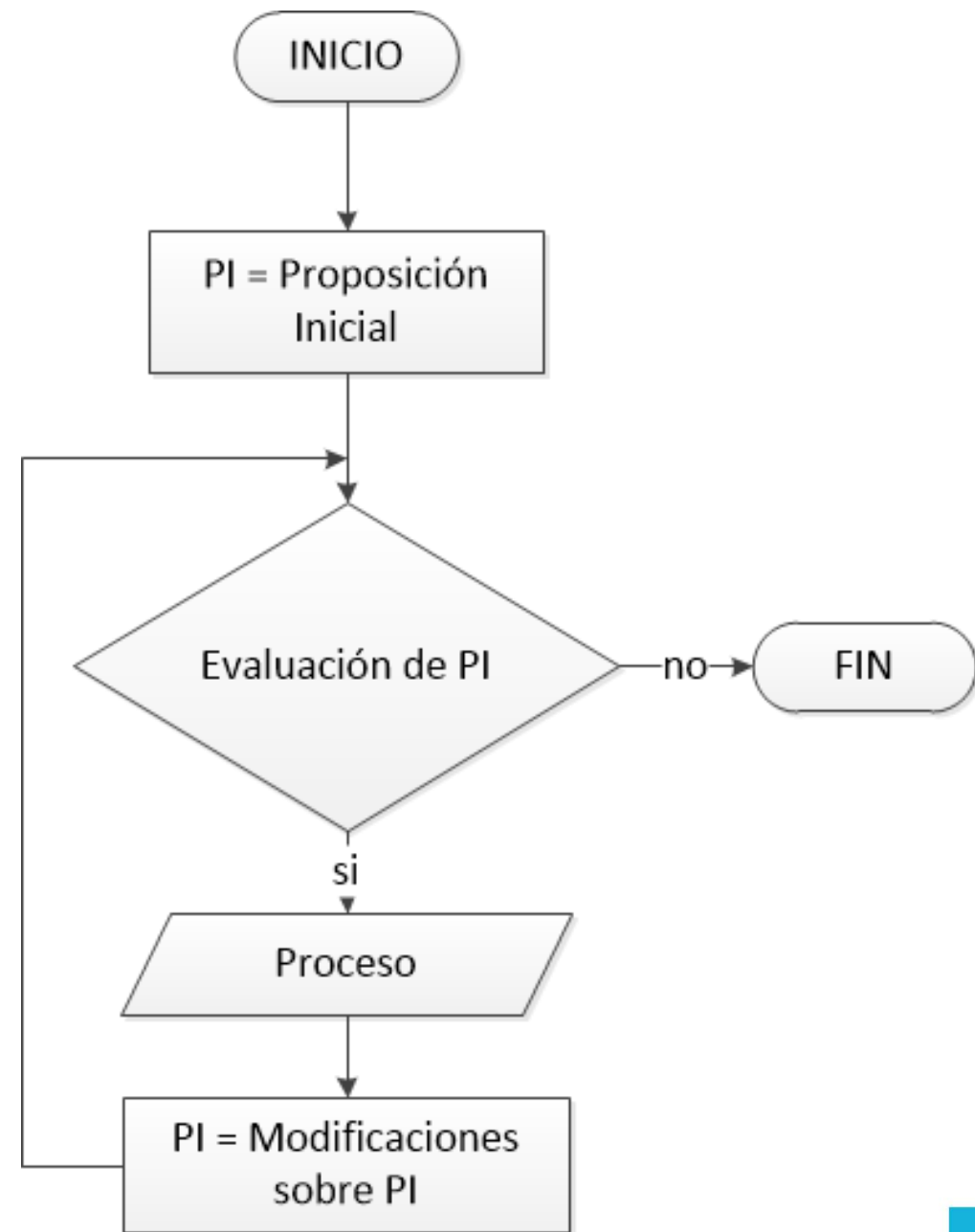
Para los ciclos Mientras (WHILE) se definen las siguientes variables:

PI: La proposición inicial, debe tener un valor verdadero inicialmente, si el valor de PI es falso, entonces el ciclo no se ejecutará.

Debe existir también un enunciado dentro del ciclo que afecte la condición, para evitar que el ciclo se ejecute indefinidamente.

MIENTRAS PI
Cuerpo / Operaciones
PI = Validaciones
FIN MIENTRAS

Ciclo Mientras (While)



6. Ciclos anidados

(Estructuras de repetición anidadas)

Ciclos anidados – Estructuras de repetición anidadas

Es posible insertar un ciclo dentro de otro si todas las instrucciones del ciclo interno están contenidas en el externo.

En el esquema podemos observar que j varía más rápidamente que i ; es decir, el ciclo interno varía más rápido que el ciclo externo. Es posible combinar estructuras como Para y Mientras, Mientras y Hasta Que, Para y Para, etcétera.

Se pueden anidar tantas estructuras como sea necesario, pero debemos recordar que se debe incluir un ciclo dentro del otro y no solaparlos.

ciclo externo – para $i = 1$ hasta 10

ciclo interno – para $j = 1$ hasta 5

```
i = 1, j = 1, j = 2, j = 3, j = 4, j = 5  
i = 2, j = 1, j = 2, j = 3, j = 4, j = 5  
i = 3, j = 1, j = 2, j = 3, j = 4, j = 5  
i = 4, j = 1, j = 2, j = 3, j = 4, j = 5  
i = 5, j = 1, j = 2, j = 3, j = 4, j = 5  
i = 6, j = 1, j = 2, j = 3, j = 4, j = 5  
i = 7, j = 1, j = 2, j = 3, j = 4, j = 5  
i = 8, j = 1, j = 2, j = 3, j = 4, j = 5  
i = 9, j = 1, j = 2, j = 3, j = 4, j = 5  
i = 10, j = 1, j = 2, j = 3, j = 4, j = 5
```

5

10



Esta licencia permite a otros distribuir, remezclar, retocar, y crear a partir de esta obra de manera no comercial y, a pesar que sus nuevas obras deben siempre mencionar a la IU Digital y mantenerse sin fines comerciales, no están obligados a licenciar obras derivadas bajo las mismas condiciones.



¡Gracias!

¿Preguntas?

IUDigital
de Antioquia
INSTITUCIÓN UNIVERSITARIA DIGITAL DE ANTIOQUIA