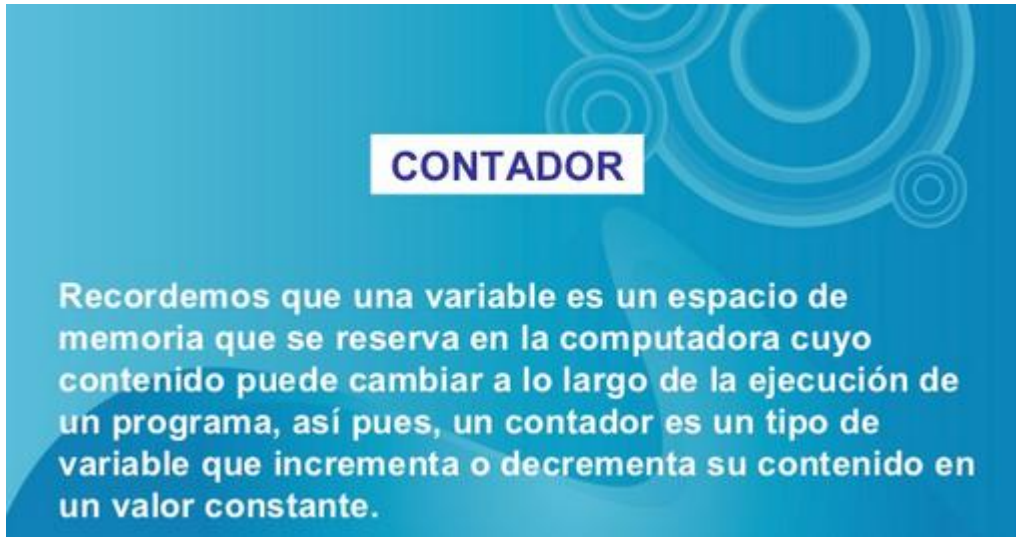


VARIABLES

CONTADORES Y ACUMULADORES

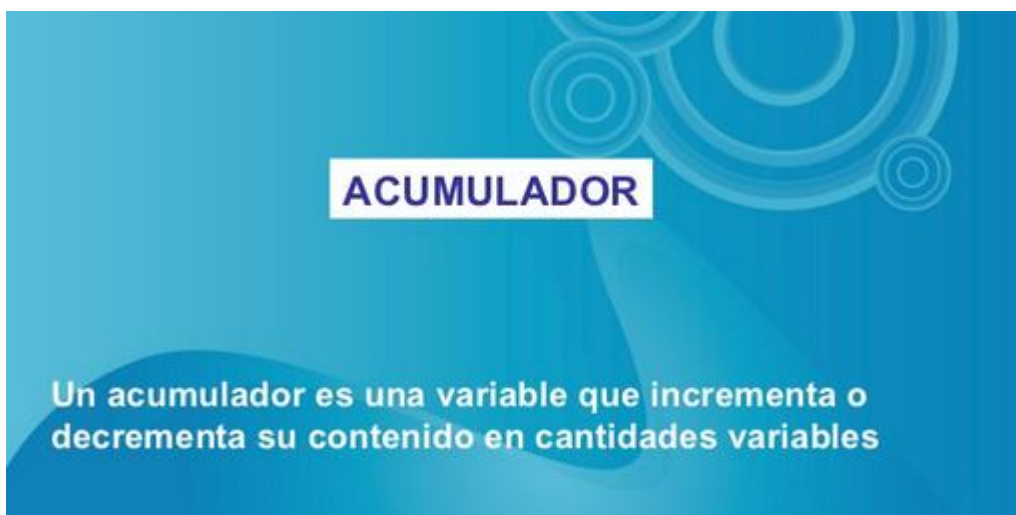
➤ **Contadores:** Se utilizan para llevar el control del número de ocasiones en que se realiza una operación o se cumple una condición. Con los incrementos generalmente de uno en uno.



CONTADOR

Recordemos que una variable es un espacio de memoria que se reserva en la computadora cuyo contenido puede cambiar a lo largo de la ejecución de un programa, así pues, un contador es un tipo de variable que incrementa o decrementa su contenido en un valor constante.

➤ **Acumuladores:** Forma que toma una variable y que sirve para llevar la suma acumulativa de una serie de valores que se van leyendo o calculando progresivamente.



ACUMULADOR

Un acumulador es una variable que incrementa o decrementa su contenido en cantidades variables

ESTRUCTURA

Para poder trabajar con estas estructuras es muy importante comprender el concepto de:

CONTADOR	Contador = contador + constante
ACUMULADOR	acumulador = acumulador + variable

https://www.youtube.com/watch?v=hIPQFnyt_RM

<https://www.youtube.com/watch?v=ScyFm9UP4FY>

ESTRUCTURAS CICLICLAS DEFINICION

Se llaman problemas repetitivos o cíclicos a aquellos en cuya solución es necesario utilizar un mismo conjunto de acciones que se puedan ejecutar una cantidad específica de veces. Esta cantidad puede ser fija (previamente determinada por el programador) o puede ser variable (estar en función de algún dato dentro del programa).

OTRA DEFINICION

Ciclos de los algoritmo
Los Ciclos.

Un algoritmo se comporta de una manera secuencial, cada instrucción se lee desde inicio hasta fin. Esto nos da pie a pensar que para lograr repetir algo, será necesario copiarlo muchas veces. En el mundo de los algoritmos encontramos la forma de repetir las veces que deseamos una instrucción o un bloque de ellas, a este elemento lo denominamos ciclo. En la naturaleza y en el mundo real la mayoría de cosas son cíclicas, comer, dormir, respirar, trabajar, estudiar, descansar.... son cosas que se repiten con diferente periodicidad.

IMAGEN CON UN EJEMPLO.



VIDEOS

<https://www.youtube.com/watch?v=mLPkF6T7Hik>

<https://www.youtube.com/watch?v=UjsU4ILNJY0>

CICLO PARA

Ciclo para también conocido en programación como ciclo **for** es un bucle que ejecuta una serie de sentencias un determinado número de veces controlado por una variable de control.

En este bucle se inicializa en una sola sentencia la variable de control el valor final y el incremento o decremento de la variable.

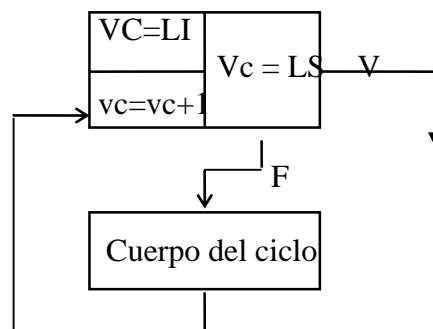
Prácticamente funciona igual que el bucle mientras y es cuestión de gustos utilizar un bucle u otro, en lo personal me gusta más emplear en mis algoritmos el ciclo mientras porque siento que tengo más control sobre el bucle ya que este se incrementa en cuerpo del bucle.

Pero cuando conozco exactamente el número de veces que se repetirá un ciclo entonces es ahí donde aplico el bucle, considero a este ciclo como manejar un carro automático donde únicamente le pisas el acelerador y te olvida de todo así de fácil

Son aquellos en que el número de iteraciones se conoce antes de ejecutarse el ciclo. La forma de esta estructura es la siguiente:

ESTRUCTURA

```
Hacer para V.C = L.I a L.S
    Accion1
    Accion2
    .
    .
    .
    AccionN
Fin-para
```



Donde:

V.C Variable de control del ciclo
L.I Limite inferior
L.S Limite superior

En este ciclo la variable de control toma el valor inicial del ciclo y el ciclo se repite hasta que la variable de control llegue al limite superior.

<https://pseudocodigoejemplos.com/ciclo-para-en-pseint/>

<https://www.youtube.com/watch?v=utpKHCAWUcA>

https://www.youtube.com/watch?v=zNbvwpl_XwE

Ciclo para en pseint

<https://www.youtube.com/watch?v=ddgdWGNMj0w>

CICLO MIENTRAS

El ciclo mientras también conocido como ciclo While en los lenguajes de programación, es una estructura que evalúa una condición y devuelve un valor que puede ser verdadero o falso cuando la condición es verdadera se ejecutan una serie de sentencias dentro del cuerpo del bucle.

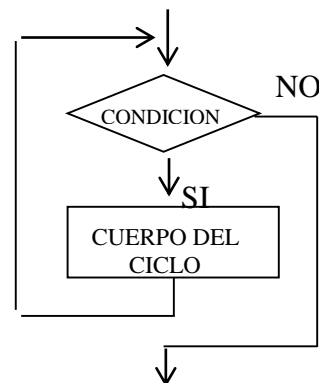
El flujo de los datos retorna y evalúa nuevamente la condición si la condición se sigue cumpliendo se siguen ejecutando las sentencias del cuerpo del bucle hasta que la condición ya no se cumpla.

Existe una variable que controla el **ciclo mientras** y se tiene que inicializar previamente antes del bucle esta variable normalmente se incrementa o decremento dependiendo la condición.

Hacer-Mientras: Esta es una estructura que repetirá un proceso durante “N” veces, donde “N” puede ser fijo o variable. Para esto, la instrucción se vale de una condición que es la que debe cumplirse para que se siga ejecutando. Cuando la condición ya no se cumple, entonces ya no se ejecuta el proceso. La forma de esta estructura es la siguiente:

ESTRUCTURA

```
Hacer mientras <condición>
    Accion1
    Accion2
    .
    .
    AccionN
Fin-mientras
```



Ejemplo:

<https://www.youtube.com/watch?v=Na7aLu07JOY>