



universidad
cenfotec_
La U de la informática

Fundamentos de Programación

Estructuras de construcción.
Diseño y confección de una estructura iterativa.

Objetivos

- Aprender el uso de las estructuras iterativas.
- Entender dos de los tipos de estructuras iterativas.
- Identificar los procesos que se ejecutan dentro de una estructura iterativa.
- Comprender la solución de un ejercicio que utilice estructuras iterativas para su resolución.

Estructura iterativa

Una **estructura iterativa** es una *estructura de control* que permite ejecutar una instrucción o un bloque de instrucciones múltiples veces.

Estas estructuras pueden ser llamadas también *bucles* o *ciclos*.

Al igual que en una estructura de selección, las acciones se ejecutarán de acuerdo a una condición dada.

Tipos de estructuras iterativas

Incondicional con contador.

- La condición de estas estructuras depende de una variable **contador** de tipo **numérica**.
- Antes de iniciar la ejecución del ciclo es posible determinar **cuántas** veces se ejecutarán las instrucciones.
- En cada ejecución del ciclo se incrementa (o decrementa) el valor de la variable **contador**.

Condicional con bandera.

- La condición depende de un valor o una variable de tipo **booleana** llamada **bandera**.
- **No** se conoce de antemano cuántas veces se ejecutará el ciclo debido a que en cada ejecución se evalúa la condición.
- En cada ejecución del ciclo se debe **modificar** uno de los valores que eventualmente harán que la condición ya no se cumpla.

Acumuladores y contadores

Durante la ejecución de las estructuras iterativas se presenta la necesidad de almacenar valores que se obtendrán durante las ejecuciones del ciclo. Esto se logra mediante variables **acumuladoras y contadoras**.

(27+20+27+23) 97/4 24.25

- **Variable acumuladora:** Es una variable que **acumula** los valores obtenidos en las iteraciones. Una vez se concluye con la ejecución del ciclo, esta variable contiene el resultado total de los valores obtenidos en las repeticiones.
 - **Ejemplos:** La suma total del salario de 5 empleados, la suma del porcentaje de las tareas realizadas por un estudiante.
- **Variable contadora:** Esta variable tiene la responsabilidad de **contar** la cantidad de iteraciones realizadas por la estructura iterativa. Tiene un funcionamiento similar a una variable acumuladora, ya que por cada ejecución acumula un nuevo valor.
 - **Ejemplos:** Panel que muestra la posición del cliente que está siendo atendido en un banco, aplicación para contar la cantidad de personas dentro de un local.

$x = x + 1$ $x++$

$x = x + 5$

$x+=5$

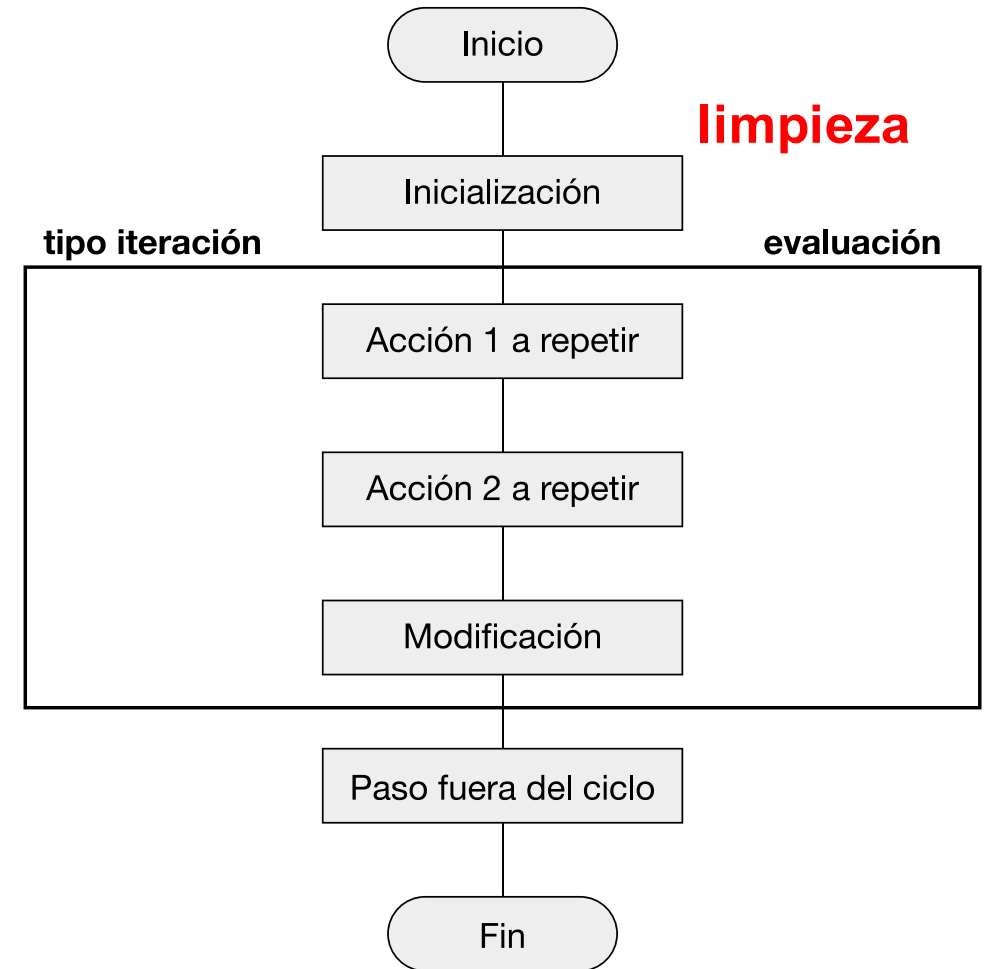
$x = x - 10$

$x-=10$

Representación en diagrama

Las estructuras iterativas están compuestas por los siguientes pasos:

1. Inicialización.
2. Evaluación.
3. Cuerpo del ciclo.
4. Modificación.

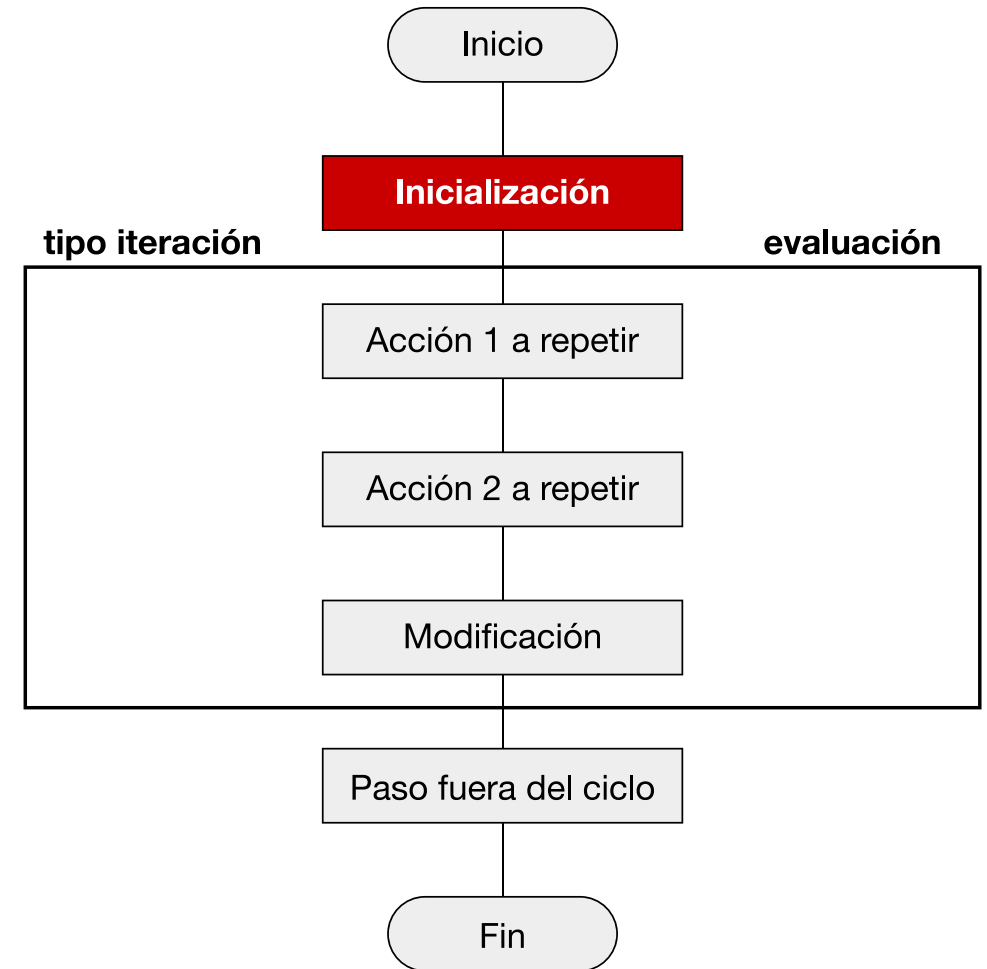


Representación en diagrama

1. Inicialización.

Paso donde se inicializan las variables que serán utilizadas para la evaluación del ciclo y dentro de las acciones que serán ejecutadas.

Aquí se inicializan las variables **contadoras** y **acumuladoras**.

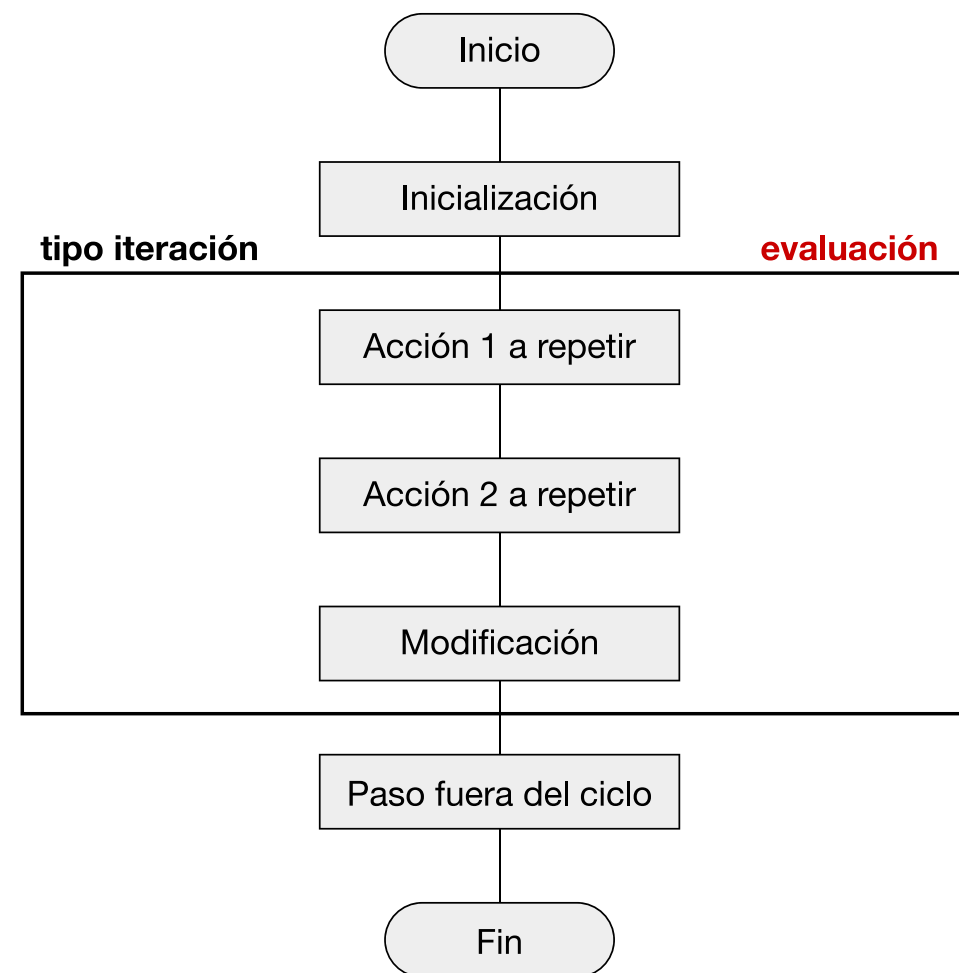


Representación en diagrama

2. Evaluación.

Paso en donde mediante una condición se determina si el ciclo se continúa ejecutando o se detiene.

Si la **evaluación** de la condición da como resultado *verdadero*, las acciones definidas dentro del ciclo se vuelven a ejecutar, si el resultado es *falso*, se ignoran las acciones y se continúa con el flujo fuera del ciclo.

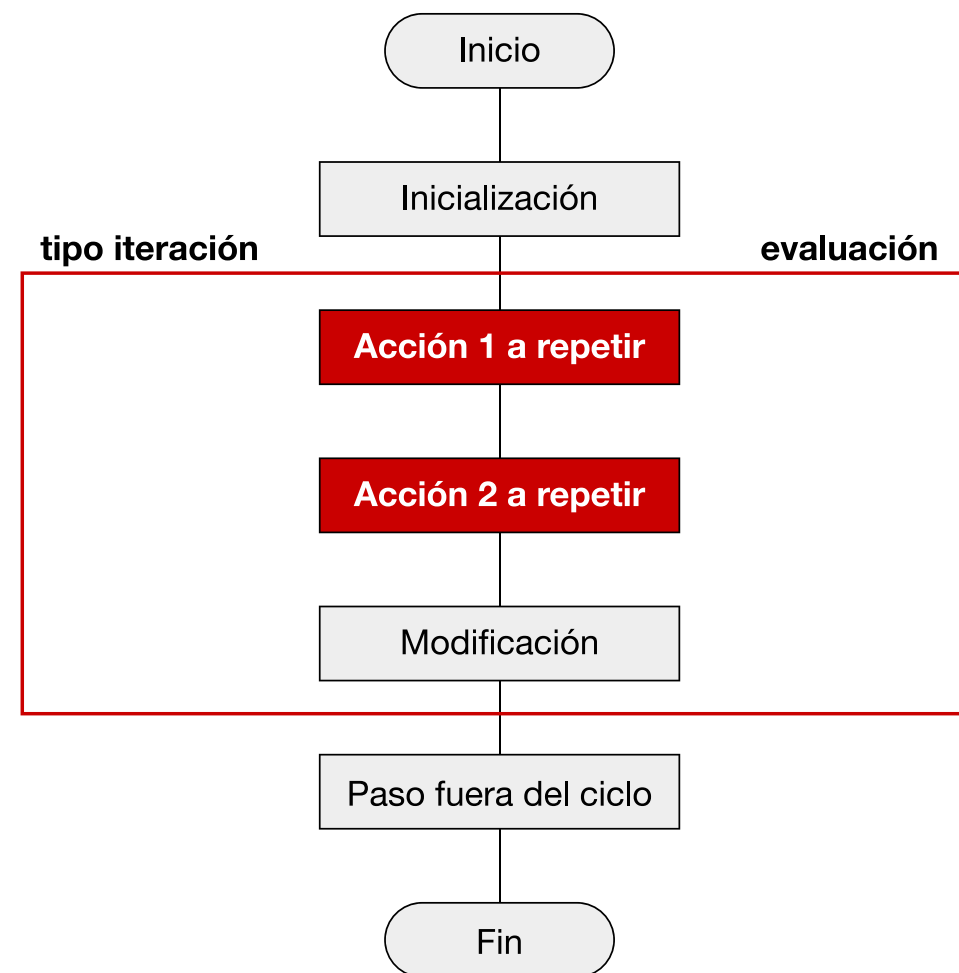


Representación en diagrama

3. Cuerpo del ciclo.

En este paso se ejecutan las **instrucciones** que se desean repetir. Puede incluir cualquier acción, como imprimir un mensaje en pantalla, pedir datos por teclado o modificar el valor de una variable.

Estas instrucciones se van a ejecutar mientras la evaluación de la condición siga siendo *verdadera*.



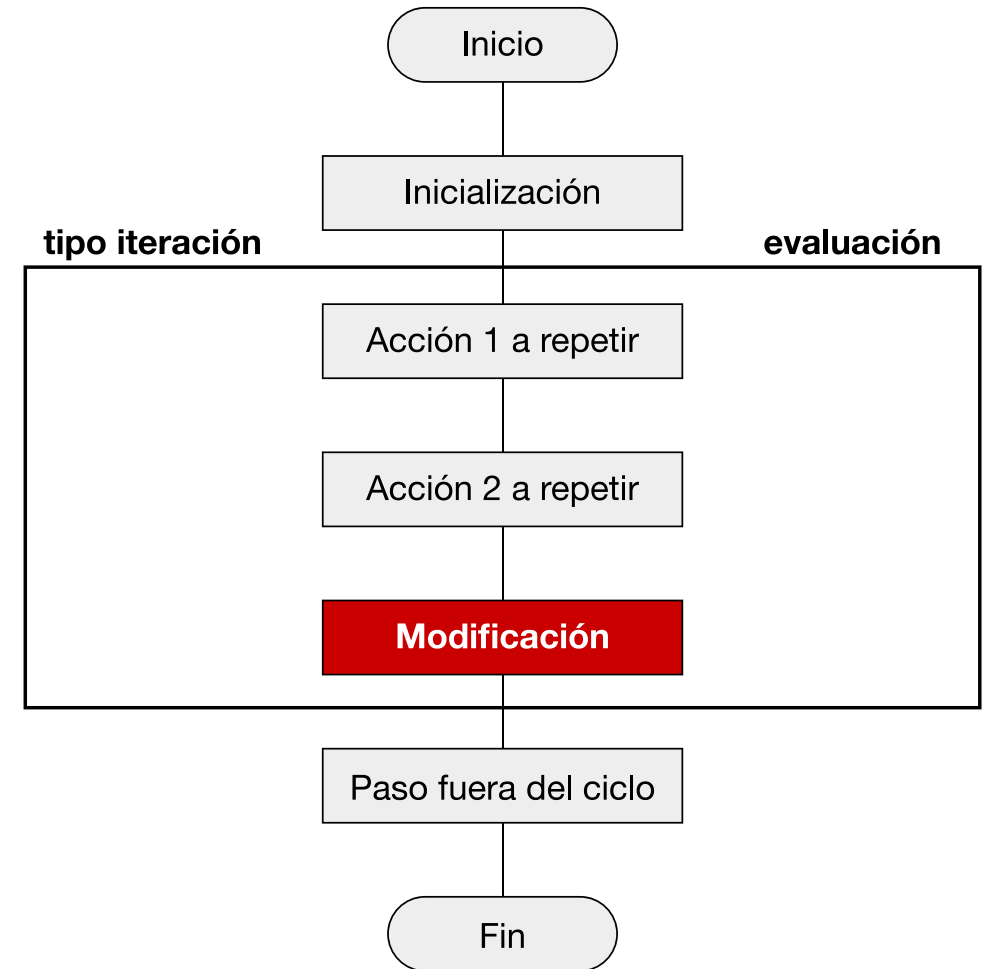
Representación en diagrama

4. Modificación.

La **modificación** es el paso responsable de cambiar la variable de la evaluación, para que el ciclo cumpla su condición de cierre y se finalice su ejecución.

La modificación siempre se ejecuta después de ejecutar el cuerpo del ciclo y antes de evaluar la condición.

Al implementarse en un lenguaje de programación, la modificación se puede encontrar al inicio o al finalizar la estructura, esto depende del tipo de iteración y de la estructura que defina el programador.



El proceso de análisis y diseño

Al construir **estructuras iterativas**, de la misma forma que se evalúa un algoritmo secuencial, es importante analizar el problema expuesto y reconocer las partes que lo componen. Para realizarlo se deben seguir estos pasos:

1. Comprender el problema y lo que se desea crear para resolverlo.
2. Identificar los datos de entrada y salida del programa.
3. Identificar las acciones que se requieren ejecutar múltiples veces, las cuáles se incorporarán dentro de la estructura iterativa.
4. Definir la evaluación y la modificación de la estructura, lo que le permite definir cuál será la condición de cierre del ciclo.
5. Traducir el algoritmo a expresiones computacionales.

El proceso de análisis y diseño

1. Comprender el problema y lo que se quiere averiguar.
2. Identificar las entradas y salidas.
3. Identificar las acciones que se repiten dentro del proceso de solución.
4. Definir la evaluación y la modificación.
5. Traducir el algoritmo a expresiones computacionales.

Problema

Se requiere calcular el promedio de una cantidad de números enteros definida por el usuario por teclado.

Planteamiento

contador

Cuando un usuario digite que desea calcular el promedio de 10 números, el algoritmo debe solicitar 10 números enteros al usuario por teclado y obtener el promedio total de estos. Si el usuario desea calcular el promedio de 6 números, solo se deben solicitar 6 números.

Debido a que la cantidad de números varía, solucionar este problema utilizando una variable para cada número se vuelve imposible.

```
public class Ejemplo01 {
```

Run | Debug

```
public static void main(String[] args) {
```

```
    int acumulador = 0, limite, numero;
```

```
    double promedio;
```

```
    limite = Utils.leerEntero("Digite la cantidad de datos: ");
```

```
    for (int i = 0; i < limite; i++) {
```

```
        numero = Utils.leerEntero(String.format("Digite el valor #%02d: ", (i+1)));
```

```
        acumulador = acumulador + numero;
```

```
    }
```

```
    promedio = acumulador / limite;
```

```
    System.out.printf("El promedio es: %.2f\n", promedio);
```

```
}
```

97

4

(double) entero

double

entero

----- > double

double

entero

----- > entero

entero

El proceso de análisis y diseño

1. Comprender el problema y lo que se quiere averiguar.
2. Identificar las entradas y salidas.
3. Identificar las acciones que se repiten dentro del proceso de solución.
4. Definir la evaluación y la modificación.
5. Traducir el algoritmo a expresiones computacionales.

Variables

- ***cantidadNumeros***: variable de entrada que almacenará la cantidad de números de los cuales el usuario desea calcular el promedio.
- ***sumaTotal***: es una variable intermedia que almacenará la suma de todos los números que digite el usuario para calcular el promedio. Esta es una variable **acumuladora**, ya que su función es acumular los datos obtenidos en cada iteración.
- ***promedio***: variable de salida que guardará el resultado de la suma de todos los números digitados por el usuario (*sumaTotal*) entre la cantidad de números digitados (*cantidadNumeros*).

$$\text{promedio} = \text{sumaTotal} / \text{cantidadNumeros}$$

El proceso de análisis y diseño

1. Comprender el problema y lo que se quiere averiguar.
2. Identificar las entradas y salidas.
3. Identificar las acciones que se repiten dentro del proceso de solución.
4. Definir la evaluación y la modificación.
5. Traducir el algoritmo a expresiones computacionales.

Tabla de variables de entrada, intermedia y salida

Variables del programa			
Descripción	Notación		Ejemplo
	Nombre	Tipo	
Entrada			
Cantidad de números que se deben solicitar al usuario.	cantidadNumeros	entero	10
Número que ingresa el usuario por teclado.	numero	real	25
Intermedia			
Suma de todos los números ingresados por el usuario.	sumaTotal	real	145
Salida			
Promedio de todos los números ingresados.	promedio	real	14.5

El proceso de análisis y diseño

1. Comprender el problema y lo que se quiere averiguar.
2. Identificar las entradas y salidas.
3. Identificar las acciones que se repiten dentro del proceso de solución.
4. Definir la evaluación y la modificación.
5. Traducir el algoritmo a expresiones computacionales.

¿Qué acciones se repiten? (Cuerpo del ciclo)

Se requiere solicitar al usuario un número tantas veces como se indique en la variable *cantidadNumeros*. Cada número solicitado será almacenado en una variable llamada *numero*.

Para calcular el promedio, primero se debe calcular la suma de todos los números, entonces, la variable *numero* debe ser **acumulada** en una variable intermedia llamada *sumaTotal* de la siguiente forma:

Leer numero
sumaTotal = sumaTotal + numero

El proceso de análisis y diseño

1. Comprender el problema y lo que se quiere averiguar.
2. Identificar las entradas y salidas.
3. Identificar las acciones que se repiten dentro del proceso de solución.
4. Definir la evaluación y la modificación.
5. Traducir el algoritmo a expresiones computacionales.

¿Cuántas veces se repetirá el ciclo? (evaluación)

El ciclo se repetirá mientras que la cantidad de números solicitados se menor o igual a la variable *cantidadNumeros*. La cantidad de números solicitados durante la ejecución del ciclo se almacena en una variable de tipo **contador**.

contador <= **cantidadNumeros**

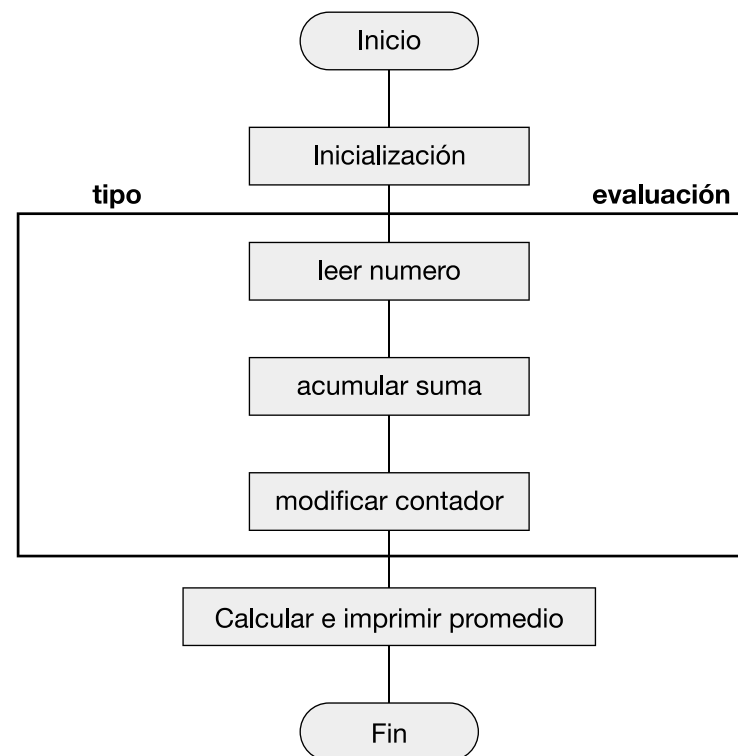
¿Cómo se modificará el contador? (modificación)

De uno en uno, **contando** la cantidad de números que el usuario ha agregado.

contador = **contador** + 1

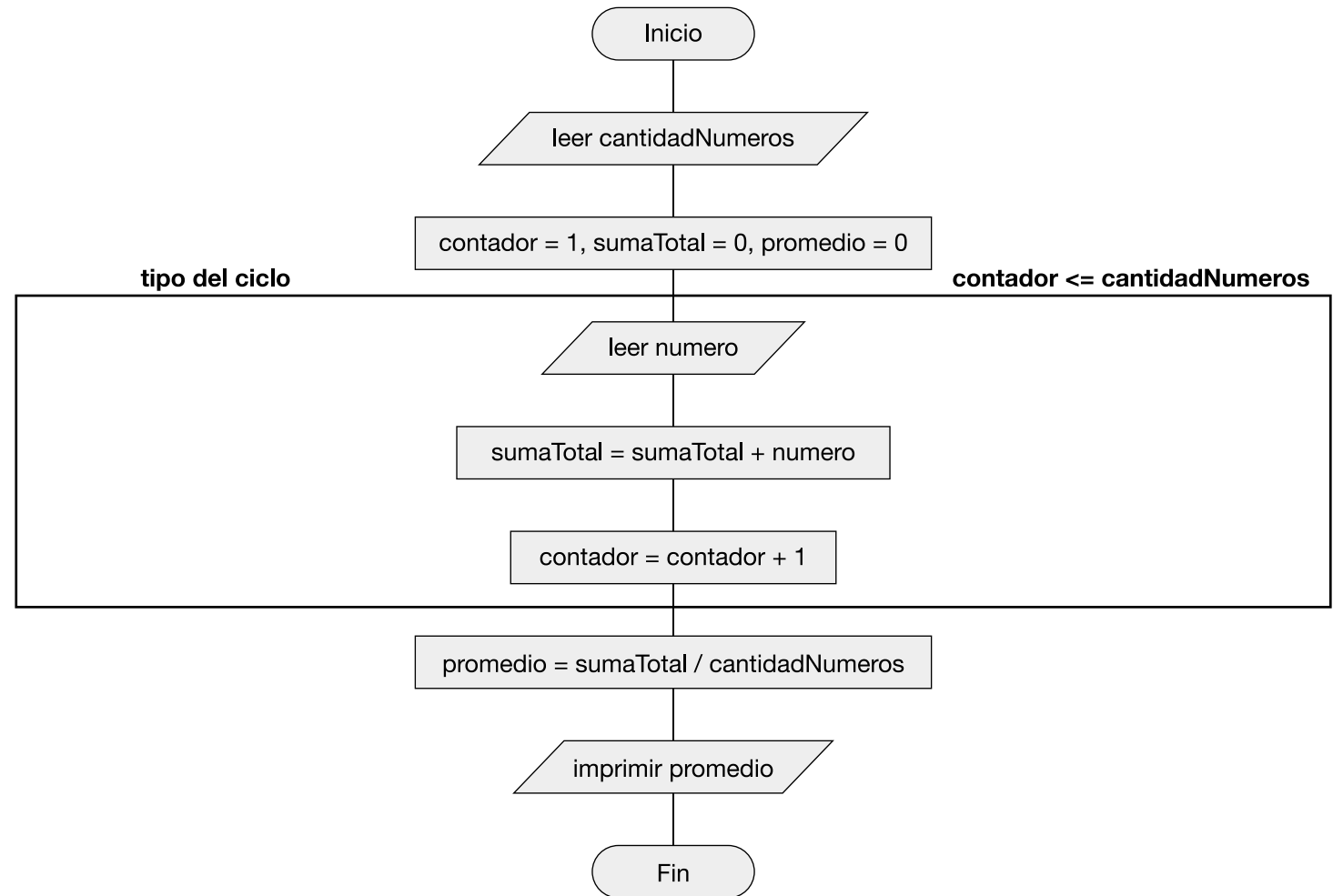
El proceso de análisis y diseño

1. Comprender el problema y lo que se quiere averiguar.
2. Identificar las entradas y salidas.
3. Identificar las acciones que se repiten dentro del proceso de solución.
4. Definir la evaluación y la modificación.
5. Traducir el algoritmo a expresiones computacionales.



El proceso de análisis y diseño

1. Comprender el problema y lo que se quiere averiguar.
2. Identificar las entradas y salidas.
3. Identificar las acciones que se repiten dentro del proceso de solución.
4. Definir la evaluación y la modificación.
5. Traducir el algoritmo a expresiones computacionales.





universidad
cenfotec_
La U de la informática