

Estructuras de repetición

Dr. José Lázaró Martínez Rodríguez

Estructuras de Repetición

Iterar o ciclar o repetir una tarea: es hacer algo y luego regresar y hacerlo una y otra vez hasta terminar la tarea.

- La computadora puede ejecutar varias veces una *línea* o un *grupo de líneas de código*. A ese grupo se le llama *cuerpo del ciclo*

Estructuras de Repetición: aplicaciones

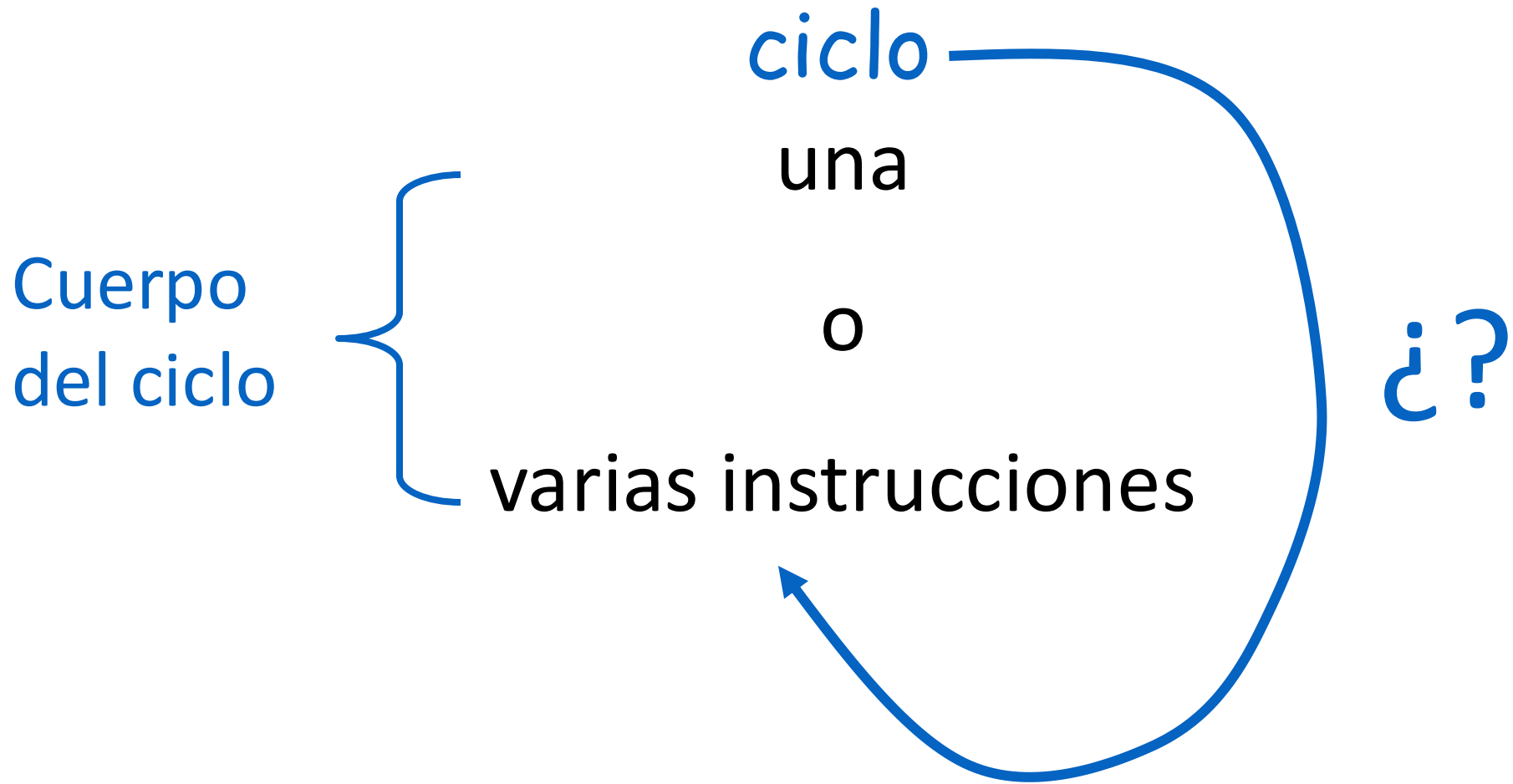
- La *introducción de muchos datos*, uno tras otro, para efectuar diversos cálculos.

Ejem. obtener el promedio de calificaciones de un alumno por grupo.

- La *clasificación periódica* de una gran colección de datos.

Ejem. la clasificación de cheques procesados por sucursal bancaria (por número de cuenta del cliente, por cada día de la semana).

Estructuras de Repetición



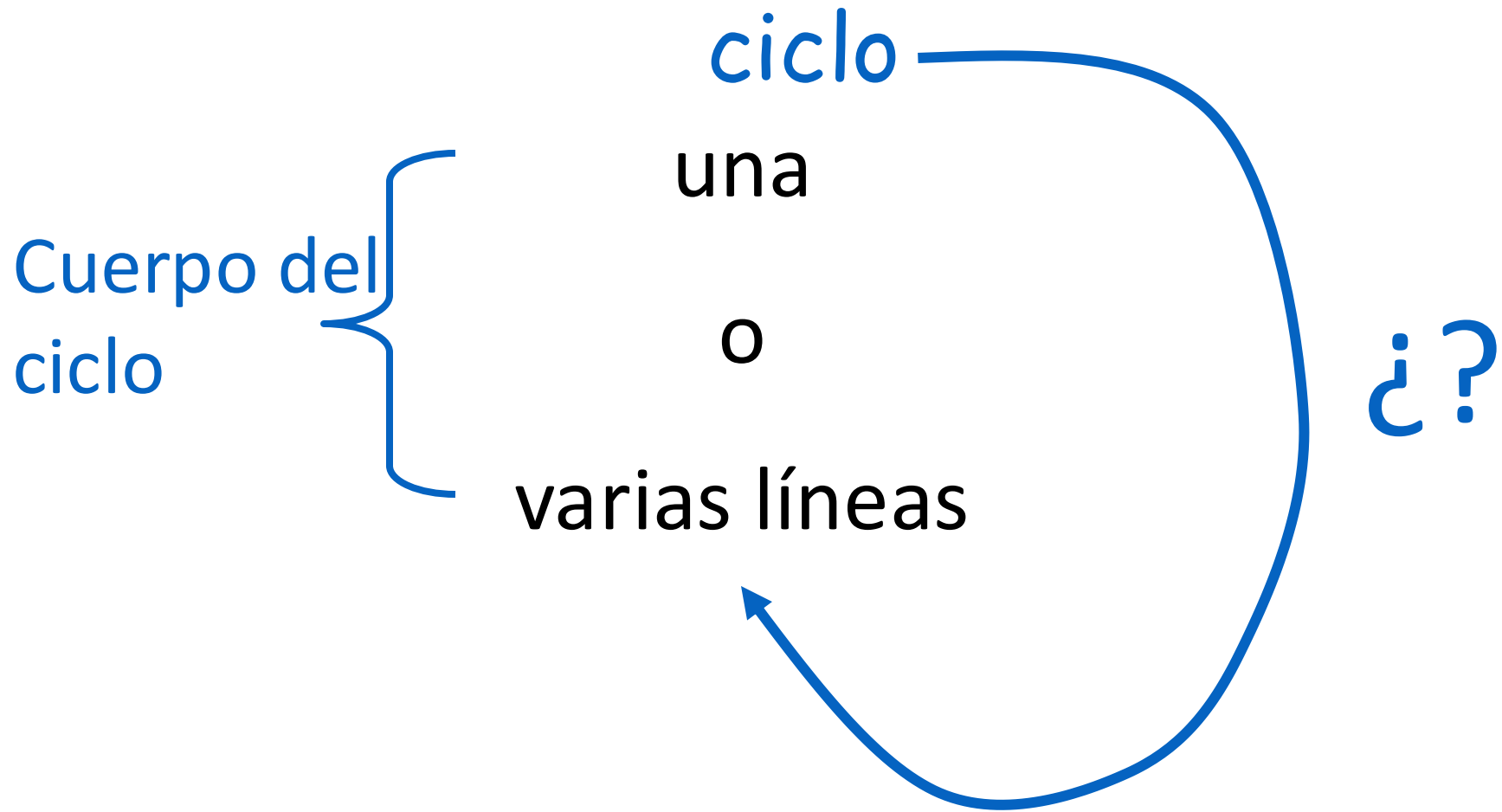
Estructuras de Repetición: aplicaciones

- La **búsqueda** de un dato en una gran colección de ellos.

Ejem. encontrar el precio actual de un artículo o el estado de una cuenta de depósito.

- **Formulas científicas** que sólo se pueden calcular por aproximaciones sucesivas.

¿Cómo controlar el ciclo?



Mecanismos de iteración

a) Incondicional

1. **Para** todos los valores de ciclo.

b) Condicionales

2. **Mientras** se valida una condición.
3. **Repetir- hasta** que se satisfaga una condición.

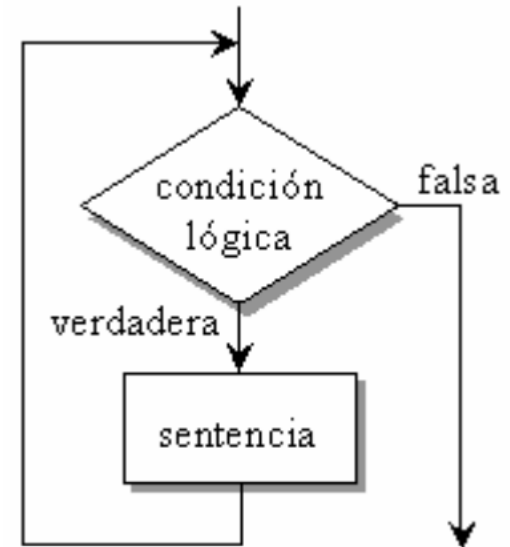
Ciclo Mientras

- Se ejecuta un grupo de instrucciones **mientras** una condición sea **verdadera**.
 - Si la expresión es **falsa** no se ejecuta el grupo de instrucciones.
- Se evalúa la expresión lógica y, si es **verdadera**, se ejecuta el grupo de instrucciones especificado antes de evaluar la condición de nuevo.

Ciclo Mientras

- Es un bucle o sentencia repetitiva con una condición al principio. Se ejecuta una sentencia mientras sea cierta una condición. La sentencia puede que no se ejecute ni una sola vez.

```
[inicializacion;]  
while (expresionLogica) {  
    sentencias;  
    [iteracion;]  
}
```



Ejemplo

Realice un algoritmo (*expresado en pseudocódigo*) que permita sumar los 10 primeros enteros positivos mayores que cero.

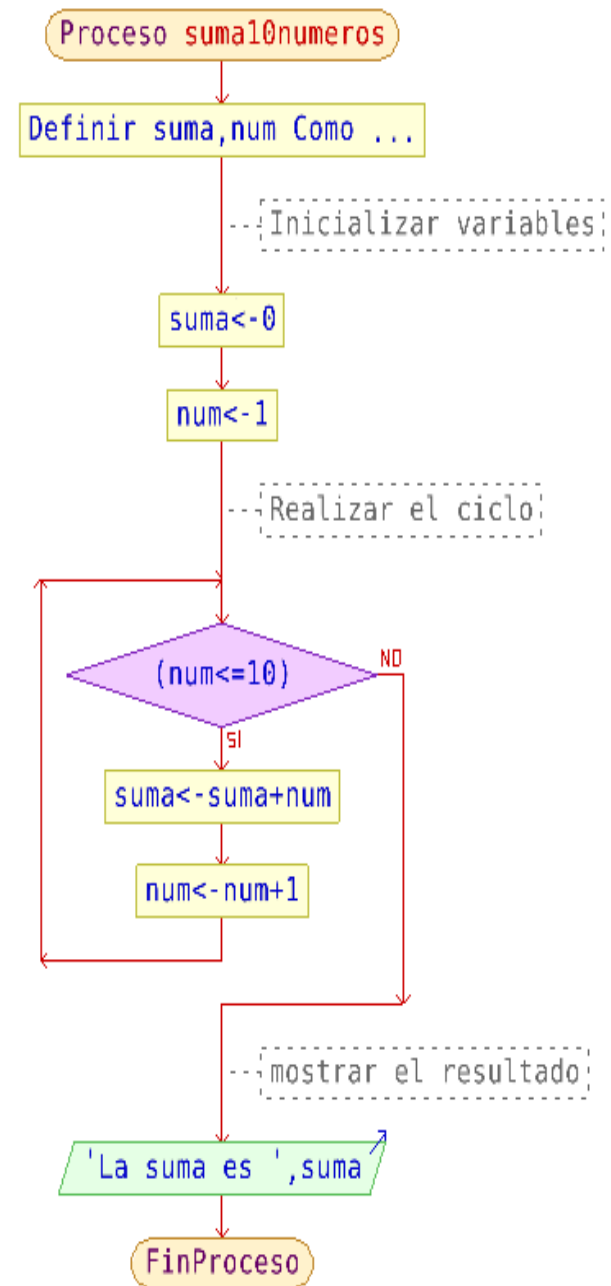
$$\sum_{i=1}^n i$$

Usar la estructura de control Mientras

Pseudocódigo

- 1. Identificación de variables
 - Definir suma, num como ENTERO
- 2. Suma de los primeros 10 números enteros
 - suma \leftarrow 0
 - num \leftarrow 1
 - Mientras (num \leq 10) hacer
 - suma \leftarrow suma + num
 - num \leftarrow num + 1
 - Fin Mientras
- 3. Salida
 - Escribir “La suma es: “, suma

Diagrama de flujo:



Código

Algún problema?

```
public clas Program
{
    public static void Main()
    {
        int suma=2;
        int n=10;
        i=0; //variable de control

        while(i<n){
            suma+=i; //acumular suma
            ++i; //incremento
        }

        System.out.println("La suma es "suma);
    }
}
```

```
using System;

public class Prog
{
    public static void Main()
    {
        int suma=0;
        int n=10;
        int i=0; //variable de control

        while(i<=n){
            suma+=i; //acumular suma
            ++i; //incremento
        }

        Console.WriteLine(suma);
    }
}
```

Ejercicios: Mientras

- 1. Pedir 10 números al usuario y mostrar solo los que son positivos.
- 2. Pedir las temperaturas máximas de los 7 días de la semana y mostrar la temperatura más alta.

Ejemplo

Realice un algoritmo (*expresado en pseudocódigo*) que permita multiplicar los 5 primeros enteros positivos pares mayores que cero.

$$\prod_{i=1}^n (i - 1)$$

Usar la estructura de control Mientras

Ejercicios:

3. Realizar el diseño de un algoritmo que a través de un menú permita realizar las operaciones de sumar, restar, multiplicar, dividir y salir. Las operaciones constarán solamente de dos operandos (leer desde teclado). Se mostrará de nuevo el menú hasta que el usuario elija salir. Por ejemplo, la ejecución final mostrará los mensajes siguientes:

```
1. Sumar
2. Restar
3. Multiplicar
4. Dividir
5. Salir
Seleccione la operación deseada: 3
Ha seleccionado la operación Multiplicar
Dato 1: 2.5
Dato 2: 10
Resultado = 25.0
Presione Enter para continuar
```


Ciclo Para

- Se usa **cuando se conoce de antemano**, el número de veces que debe repetirse una instrucción o conjunto de ellas.
- Es un ciclo incondicional, que abarca todos los valores de una progresión, empieza con el primer valor y termina con el último de ellos.
- Los valores de la progresión deben ser asignados a una variable denominada variable de control.

Pseudocódigo

Ciclo incondicional:

Para *Variable_de_control* \leftarrow *valor_inicial* **hasta** *valor_final* **Con**
 Paso *incremento* **hacer**
 Acción(es)
Fin Para



Cuerpo del ciclo **Para**

The diagram consists of a blue rectangular box. Inside the box, the text 'Cuerpo del ciclo Para' is written. A blue arrow originates from the top-right corner of the box and points diagonally upwards and to the left, ending at the word 'hacer' in the pseudocode block above.

Ciclo Para

- Es un bucle o sentencia repetitiva que
 - i) ejecuta la sentencia de inicio
 - ii) verifica la expresión booleana de término.
 - a. si es cierta, ejecuta la sentencia entre llaves y la sentencia de iteración para volver a verificar la expresión booleana de término
 - b. si es falsa, sale del bucle.

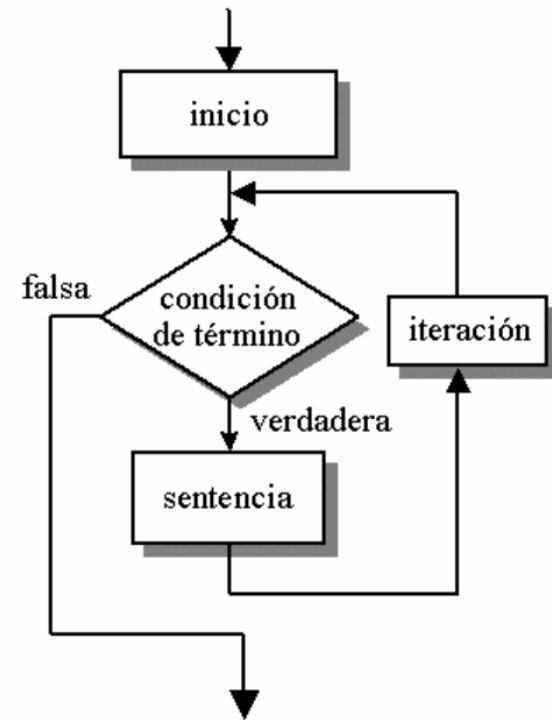
Ciclo Para

- Estructura

```
for (inicio; termino; iteracion)  
    sentencia;
```

- Usando variable de control

```
for (i = valor_inicial; i <= valor_final; i++) {  
    sentencia;  
}
```



Ciclo For

- Imprimir Hola Mundo 10 veces

```
using System;  
  
public class Program  
{  
    public static void Main()  
    {  
        for(int i=1;i<=10;i++){  
            Console.WriteLine("Hello World");  
        }  
    }  
}
```

Algún problema?

Ninguno, :)

Incremento

Valor inicial

Condición de
término

Ciclo For

- Y si quiero mis 10 de otra forma?

```
for(int i=0;i<=9;i++){  
    Console.WriteLine("Hello World");  
}
```

Algún problema?

```
for(int i=10;i>=1;i--){  
    Console.WriteLine("Hello World");  
}
```

Y ahora?

```
for(int i=1;i<=20;i+=2){  
    Console.WriteLine("Hello World");  
}
```

Y esto también
imprime 10 veces?

Ciclo For

- Pues cuánto vale mi variable de control?

```
for(int i=1;i<=10;i++){  
    Console.WriteLine("Hello World {0}",i);  
}
```

```
Hello World 1  
Hello World 2  
Hello World 3  
Hello World 4  
Hello World 5  
Hello World 6  
Hello World 7  
Hello World 8  
Hello World 9  
Hello World 10
```

Entonces puedo usar mi variable de control en otras expresiones?

claro

Ejemplo

Realice un algoritmo (*expresado en pseudocódigo*) que permita sumar los 5 primeros enteros positivos

Usar la estructura de control Para

Pseudocódigo: Suma 5 Enteros

/* Suma los 10 primeros enteros positivos*/

INICIO

1. Identificación de variables

Definir suma, num como Entero

2. Suma de los primeros 10 números enteros

suma \leftarrow 0

Para num \leftarrow 1 hasta 10 con Paso 1
hacer

 suma \leftarrow suma + num

Fin Para

Pseudocódigo: Suma 10 Enteros

3. Salida de resultado

Escribir ('La suma es: ', suma)

FIN

Ciclo For

- Posible solución

Algún problema?

```
using System;

public class Program
{
    public static void Main()
    {
        for(i=0;i<5;i++){
            int suma=0;
            suma=suma+i;
            Console.WriteLine("La suma")
        }
    }
}
```

```
using System;

public class Program
{
    public static void Main()
    {
        int suma=0;
        for(int i=0;i<=5;i++){
            suma=suma+i;
        }
        Console.WriteLine("La suma es: "+suma);
    }
}
```

Ejercicios: ciclo para

- 1. Realizar un algoritmo que realice la sumatoria de todos los números pares entre 0 y 100.
- 2. Realizar un algoritmo que de cómo resultado la sumatoria de los cuadrados de los primeros n números enteros. El valor n está dado por el usuario.
- 3. Diseñar un algoritmo para imprimir la tabla de multiplicar de un número dado por el usuario entre 1 y 10.