

# Fundamentos de Programación en C# 2

## Contenido

### Introducción

La estructura de repetición while

La estructura de repetición do .. while

La estructura de repetición for

La estructura de repetición for each



# Introducción

- Estructura de repetición
  - Permiten al programador especificar que una acción deberá ser repetida, dependiendo del valor de una condición
- Repetición controlada por contador, elementos
  - Variable de control
    - Usada para determinar hasta cuando el ciclo continua iterando
  - Valor inicial de la variable de control
  - Incremento o decremento
    - Describe como se modifica la variable de control durante cada iteración
  - Condición
    - Comprueba si la variable de control alcanzo el valor final



# La estructura de repetición `while` - 1

- Se comprueba la condición en la cabecera del ciclo
  - Si se cumple, se ejecuta con el cuerpo del ciclo
  - Si no se cumple se sigue con la siguiente instrucción después del ciclo



# La estructura de repetición while - 2

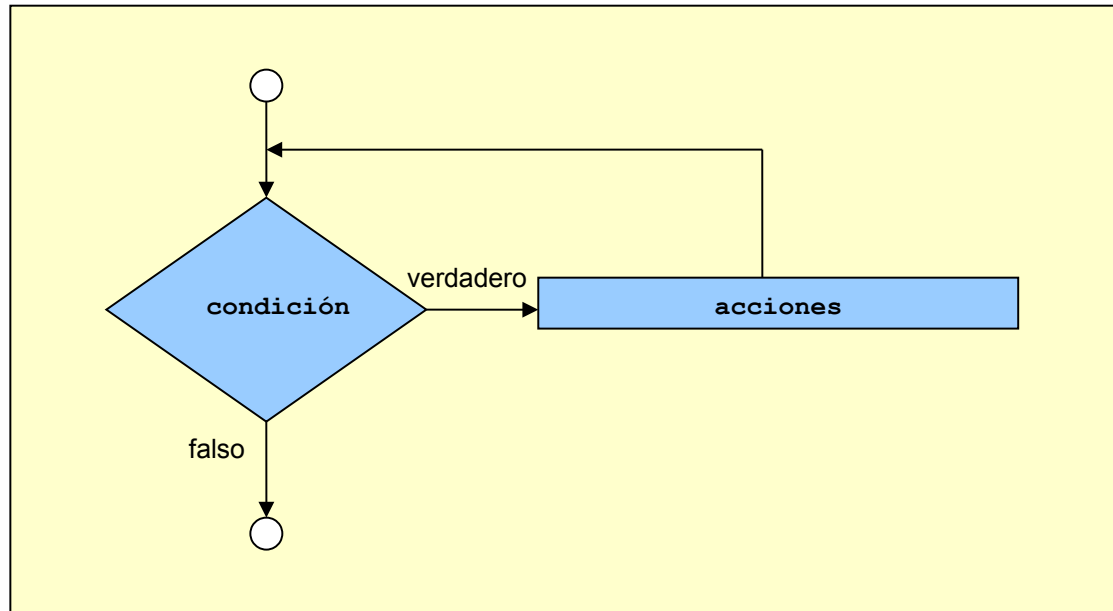


Fig. 1 Diagrama de Flujo de la estructura de repetición while



```
int contador=0;
while(contador <= 10)
{
    Console.Write("{0} ", contador);
    contador = contador + 1;
}

Console.WriteLine();

Console.WriteLine("Contador tiene el valor de {0}", contador);
```

```
0 1 2 3 4 5 6 7 8 9 10
Contador tiene el valor de 11
```

**Salida del Programa**

# La estructura de repetición `do..while` - 1

- Es similar a la forma anterior
  - Solo que ahora la expresión se evalúa al final del ciclo.
- Esto garantiza que el ciclo se ejecuta al menos una vez.



# La estructura de repetición do.. while - 2

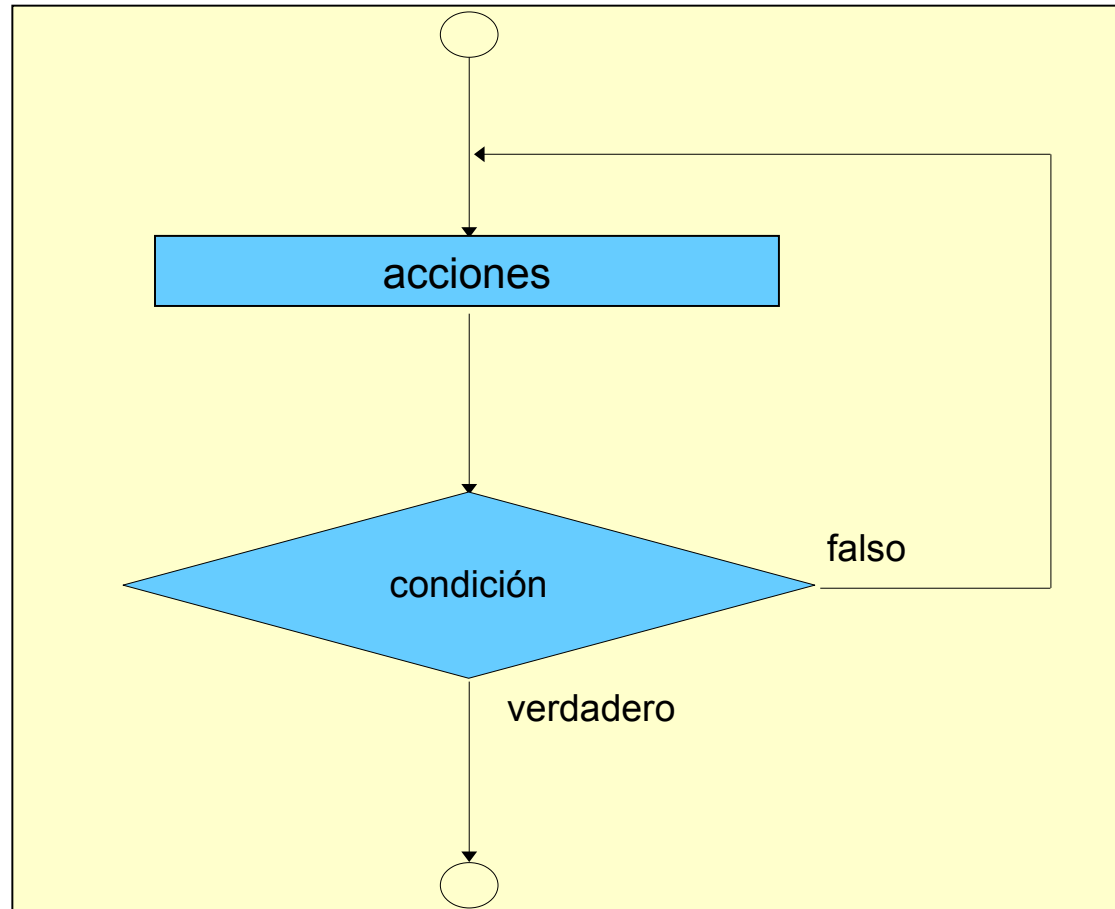


Fig. 2 Diagrama de Flujo de la estructura de repetición do .. while



```
int contador=0;
do
{
    Console.Write("{0} ", contador);
    contador = contador + 1;
} while (contador <= 10);

Console.WriteLine();
Console.WriteLine("Contador tiene el valor de {0}", contador);
```

```
0 1 2 3 4 5 6 7 8 9 10
Contador tiene el valor de 11
```

**Salida del Programa**



# La estructura de repetición **for** – 1

- Estructura de repetición **for**
  - Sintaxis: **for** (Expresion1, Expresion2, Expresion3)
    - Expresion1 = nombre de la variable de control
      - Puede contener varias variables
    - Expresion2 = condición de continuación del ciclo
    - Expresion3 = incremento / decremento
      - Si Expresion1 tiene varias variables, Expresion3 deberá tener varias variables en concordancia
      - ++contador y contador++ son equivalentes
  - Alcance de la variable
    - Expresion1 solo puede ser usada en el cuerpo del ciclo **for**
    - Cuando el ciclo termina, la variable expira



# La estructura de repetición **for** - 2

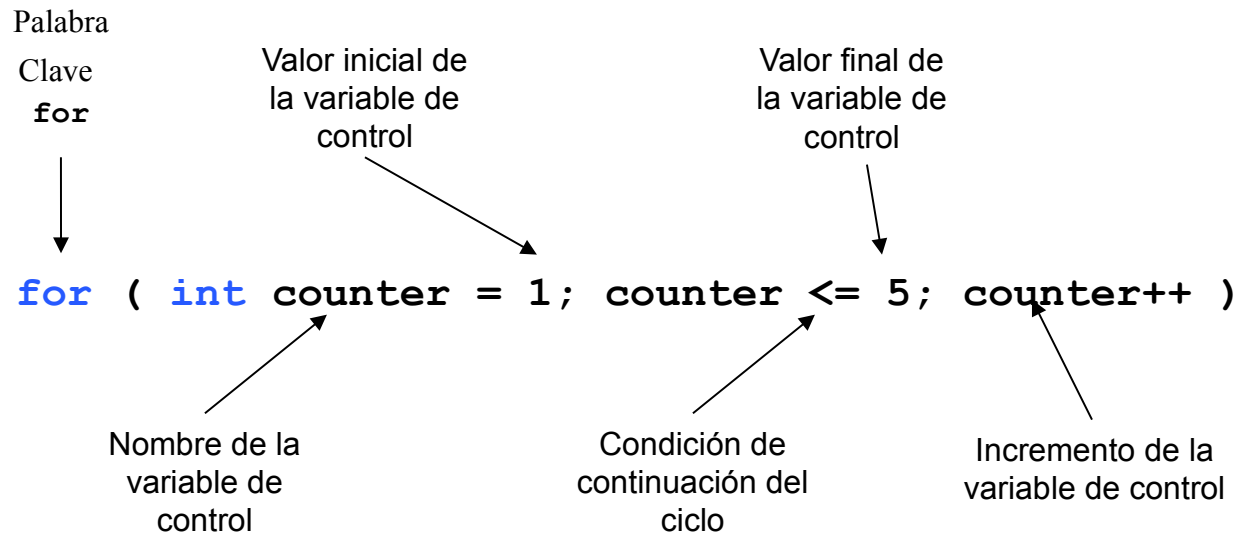
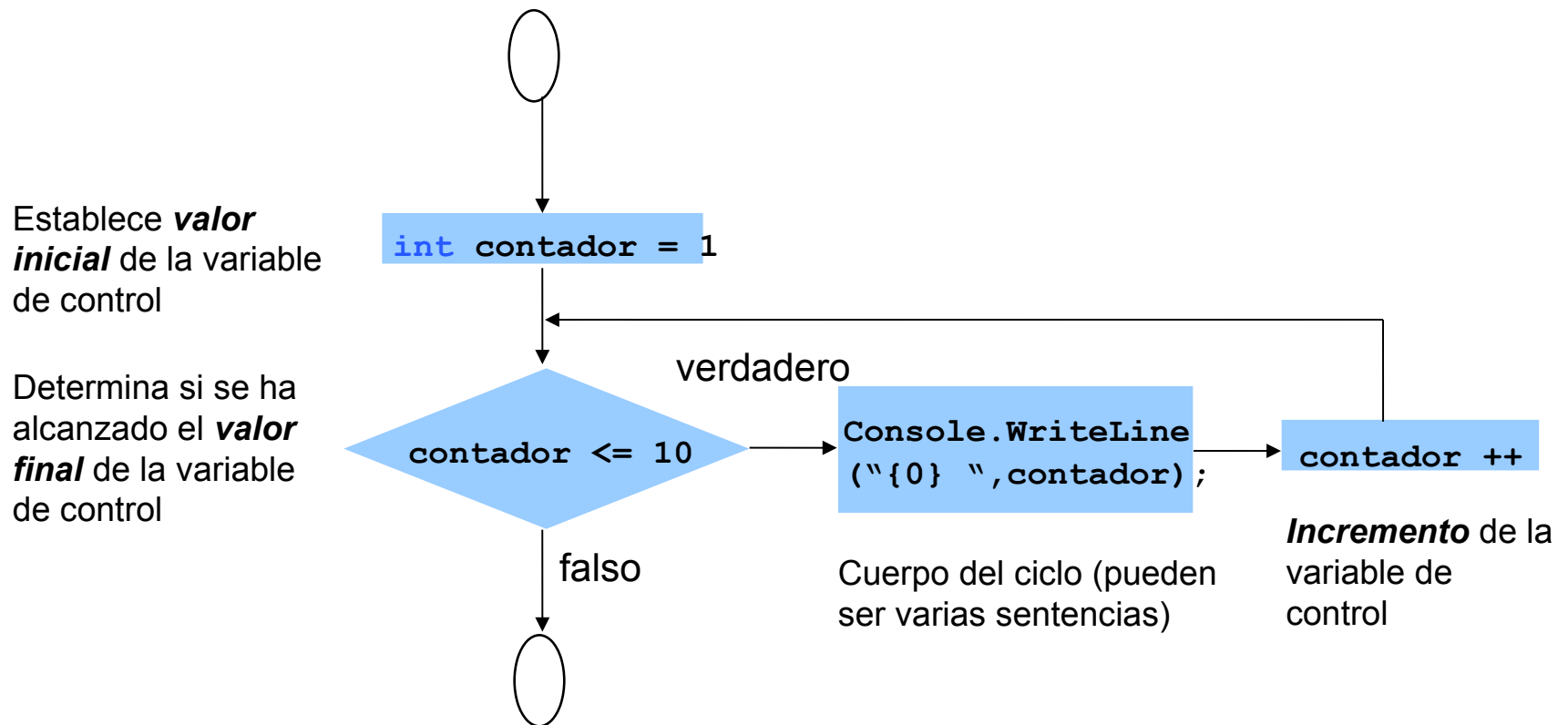


Fig. 3 Componentes de la cabecera de un **for** típico



# La estructura de repetición **for** - 3



**Fig. 4** Diagrama de flujo para una estructura de repetición **for** típica





```
for(int contador=0 ; contador <= 10 ; contador++)  
{  
    Console.Write("{0} ", contador);  
}
```

0 1 2 3 4 5 6 7 8 9 10

**Salida del Programa**

## La estructura de repetición for each

- La estructura de repetición es usada para iterar a través de los valores en las estructuras de datos tales como arreglos
- No tienen contador
- Se usa una variable para representar el valor de cada elemento





```
int[] arregloGrados = {77, 68, 86, 73 ,  
                        98, 87, 89, 81 , 70, 90, 86, 81 };  
  
int gradoMinimo = 100;  
  
foreach ( int grado in arregloGrados )  
{  
    if (grado < gradoMinimo)  
        gradoMinimo = grado;  
}  
  
Console.WriteLine( "El grado mínimo es: " + gradoMinimo );
```

El grado mínimo es: 68

**Salida del Programa**