

# 5. Iterativos

## Fundamentos de Informática

Especialidad de Electrónica – 2013-2014

Ismael Etxeberria Agiriano



Escuela Universitaria  
de Ingeniería  
Vitoria-Gasteiz

Ingeniaritzako  
Unibertsitate Eskola  
Vitoria-Gasteiz



Universidad  
del País Vasco

Euskal Herriko  
Unibertsitatea

# Índice

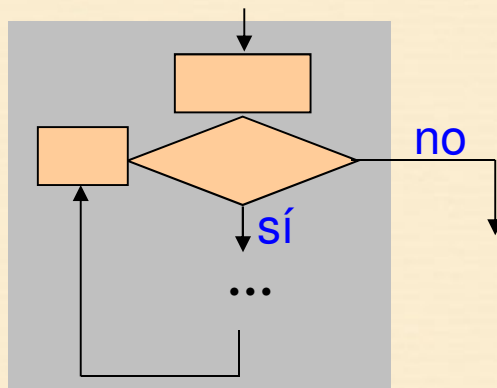
## 5. Iterativos

1. Análisis de algoritmos iterativos
2. Ej09: while
3. Ej10: do-while
4. Ej11: for
5. Resumen

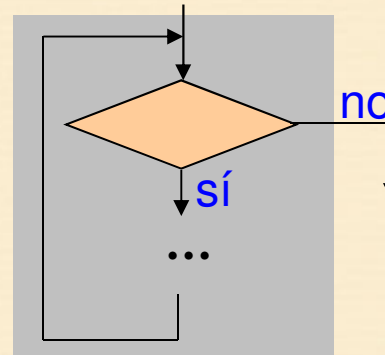
# 1. Análisis: condición de parada

- ¿Puedo determinar de antemano **cuántas veces** se ha de ejecutar el cuerpo (“...”) ?
  - Sí: estructura **for**
- ¿Quiero que se ejecute **ceros o más veces** ?
  - Sí: estructura **while**
- ¿Quiero que se ejecute **al menos una vez** ?
  - Sí: estructura **do-while**

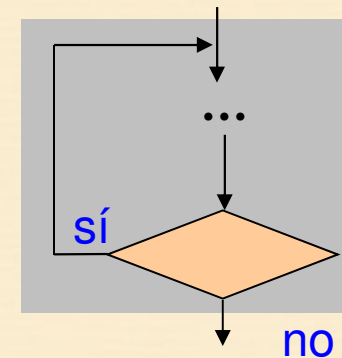
for



while



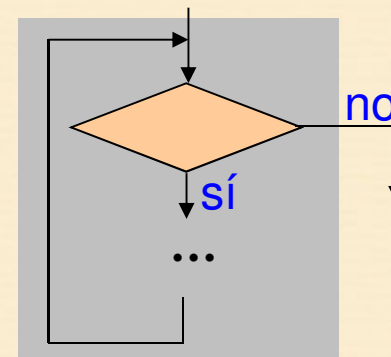
do-while



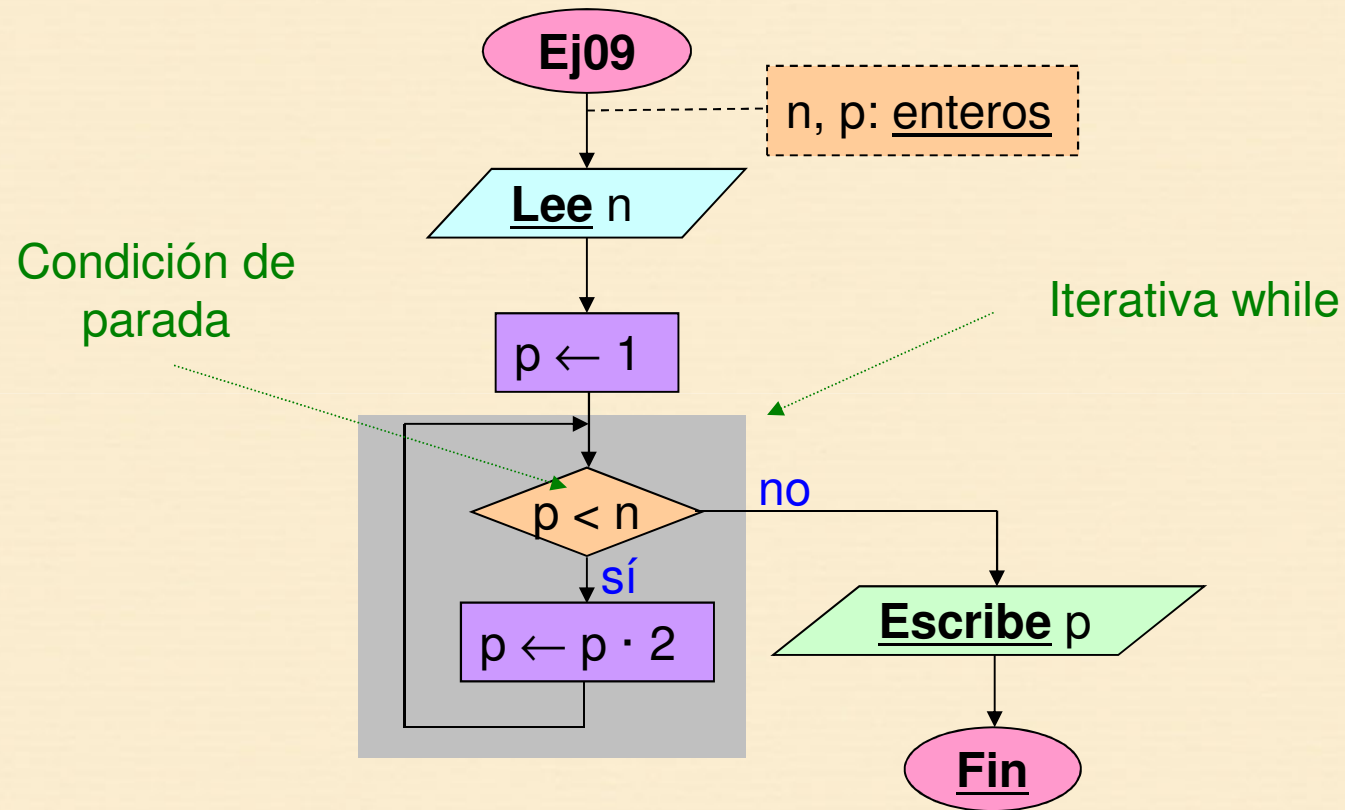
## 2. Ejemplo 09

- **Título:**
  - Iterativa while (mientras)
- **Nombre**
  - Ej09
- **Descripción**
  - Calcular la primera potencia de 2 mayor o igual que un número natural dado
- **Observaciones**
  - Cero o más veces: **while**
  - **Productorio**

while



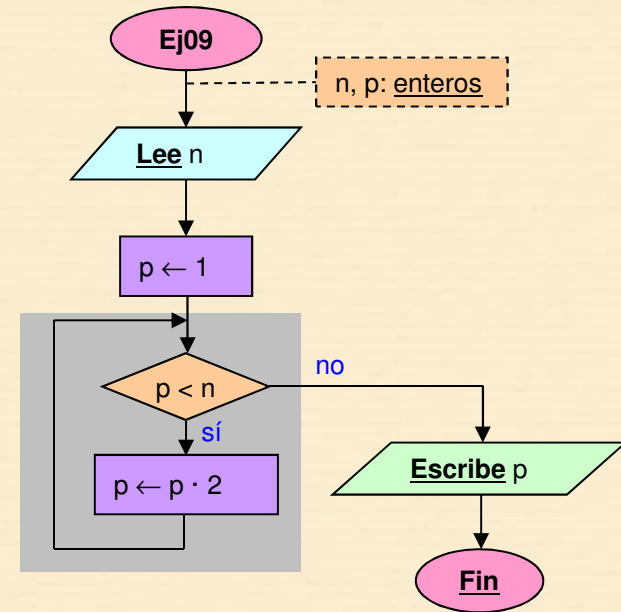
# Ej09: Diagrama de Flujo



# Ej09: Codificación C

```
/* Ej09 */
#include <stdio.h>
void main (void)
{
    int n, p;
    printf ("Introduce un número: ");
    scanf ("%d", &n);
    p = 1;
    while (p < n)
        p *= 2;
    printf ("Potencia: %d\n", p);
}
```

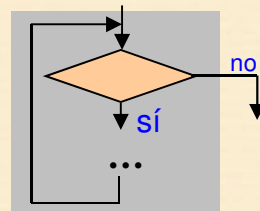
Iterativa while



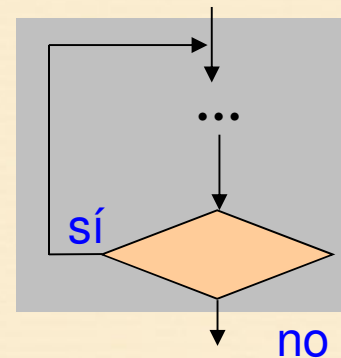
### 3. Ejemplo 10

- **Título:**
  - Iterativa do-while (haz-mientras)
- **Nombre**
  - Ej10
- **Descripción**
  - Contar una serie de caracteres introducidos del teclado hasta encontrar un punto (incluido).
- **Observaciones**
  - Una o más veces: **do-while**
  - **Contar**

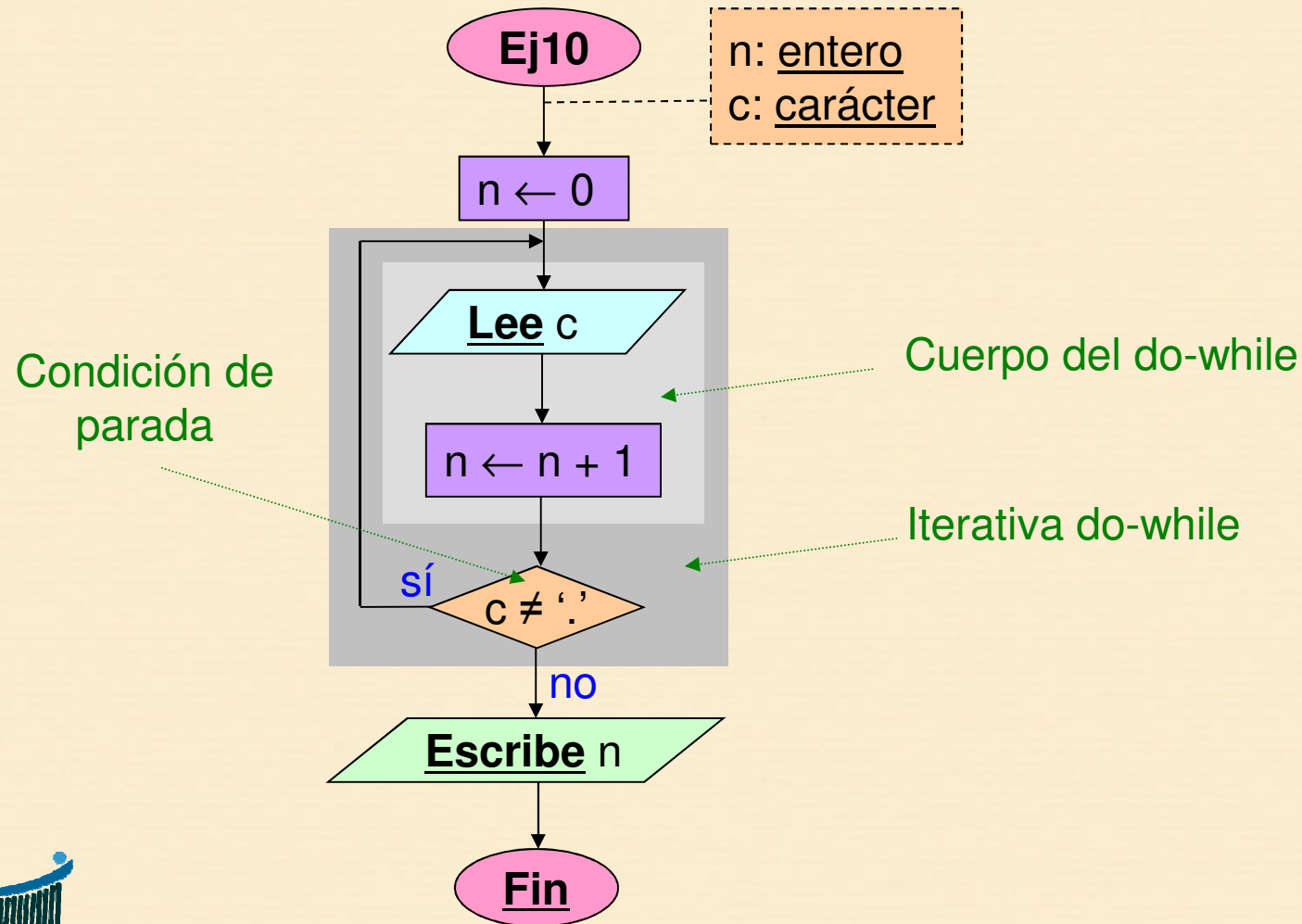
while



do-while



# Ej10: Diagrama de Flujo





# Ej10: Codificación C

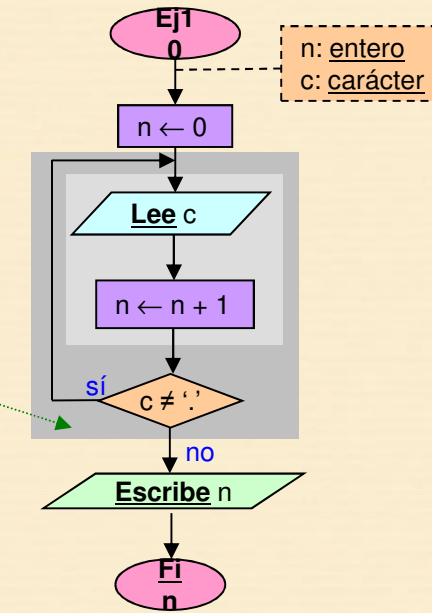
```

/* Ej10 */
#include <stdio.h>
void main (void)
{
    int n;
    char c;
    n = 0;
    printf ("Introduce caracteres: ");
    do {
        scanf ("%c", &c);
        n++;
    } while (c != '.');
    printf ("Caracteres: %d\n", n);
}

```

Iterativa do-while

Pedimos una sola vez



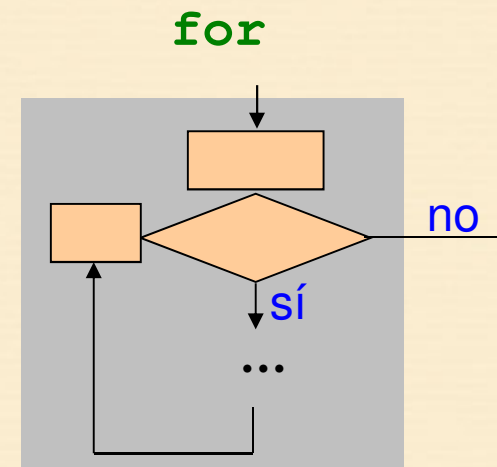
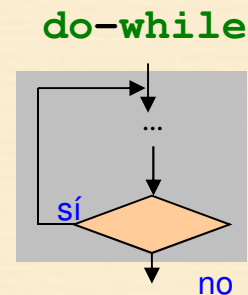
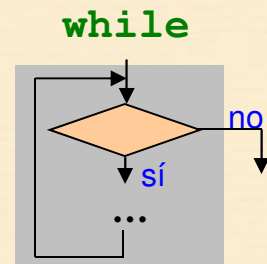
## 4. Ejemplo 11

- **Título:**
  - Iterativa for (para)
- **Nombre**
  - Ej11
- **Descripción**
  - Escribir las sumas parciales de los n primeros términos de una progresión aritmética con  $a_0 = 1$  y  $d=1$
- **Observaciones**
  - Número n de veces: **for**
  - **Sumatorio**

$$\left\{ f = \sum_{i=1}^n i \right\}$$

Salida para n = 8

i	f
–	–
1:	1
2:	3
3:	6
4:	10
5:	15
6:	21
7:	28
8:	36

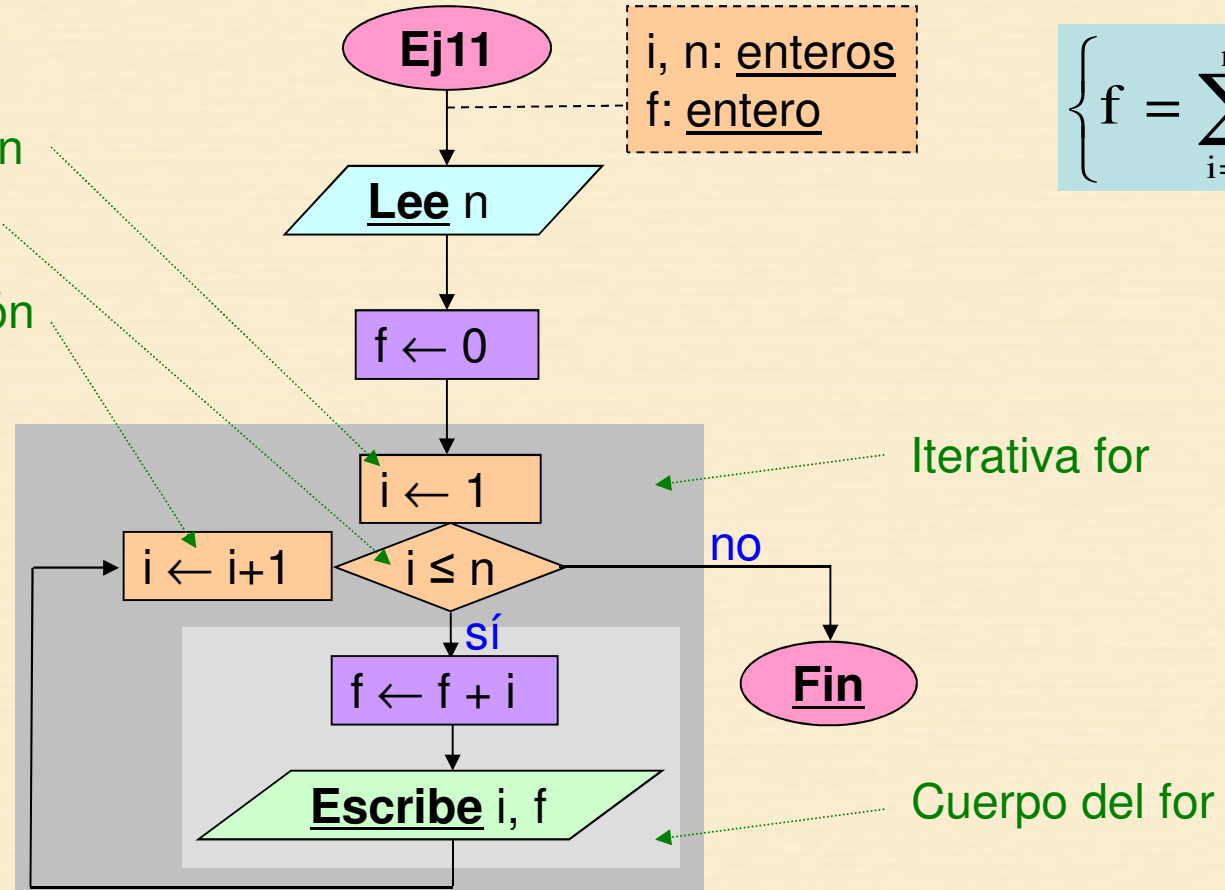


# Ej11: Diagrama de Flujo

Inicialización  
Condición  
Actualización

*i, n: enteros*  
*f: entero*

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



# Ej11: Codificación C

```

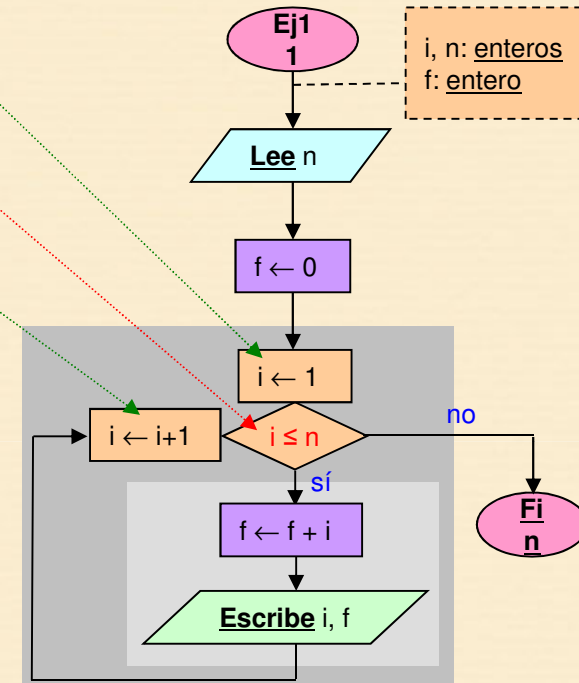
/* Ej11 */
#include <stdio.h>
void main (void)
{
    int i, n;
    int f;
    printf ("Introduce n: ");
    scanf ("%d", &n);
    f = 0;
    for (i=1; i <= n; i++) {
        f += i;
        printf ("%2d: %d\n", i, f);
    }
}

```

Inicialización

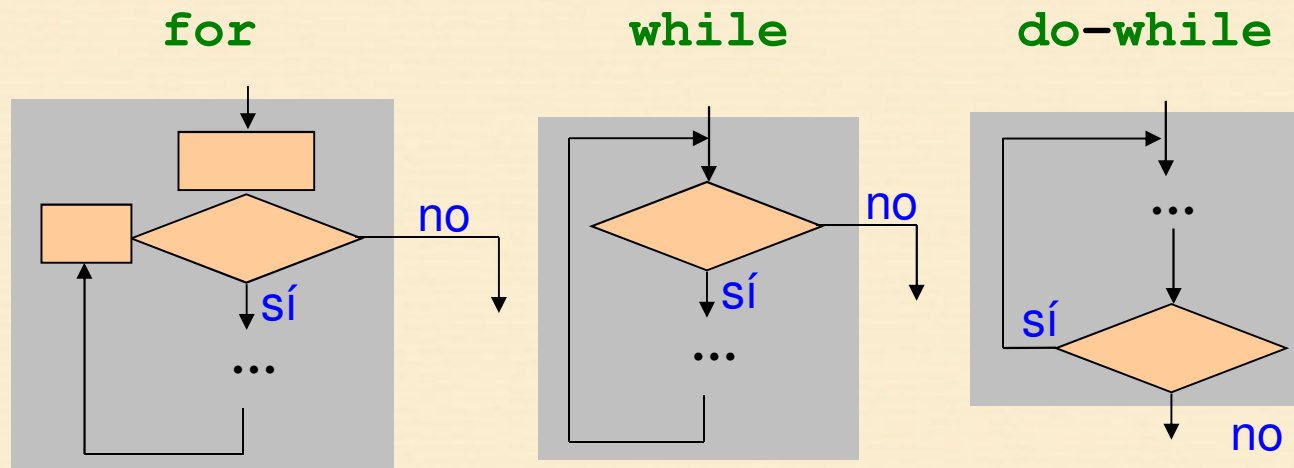
Condición

Actualización



## 5. Resumen

- Ejemplos básicos
  - Inicializar al **elemento neutro** y realizar la operación iterativa
- Patrones de operaciones
  - **Sumar**. Elemento neutro: 0
  - **Contar**. Elemento neutro: 0
  - **Producto**. Elemento neutro: 1
  - **Concatenación**. Elemento neutro: cadena vacía





Escuela Universitaria de Ingeniería Vitoria-Gasteiz    Ingeniaritzako Unibertsitate Eskola Vitoria-Gasteiz

eman ta zabal zazu



Universidad del País Vasco    Euskal Herriko Unibertsitatea