



# Introducción a la Programación

*Bioinformática 24-25*

Grado en Biomedicina

[idelhgar@uax.es](mailto:idelhgar@uax.es)

# Lógica de programación

- Ordenadores, móviles y otros dispositivos electrónicos funcionan realizando una serie de operaciones en un orden determinado.
- A la lista de órdenes se llama **programa**
- Para darle esas órdenes se emplea un **lenguaje de programación**
- A la lista de operaciones que resuelven algún problema paso a paso se le llama **algoritmo** -> **conjunto de tareas ordenadas**

## 1. Primer algoritmo:

1. Salir de la clase

## 2. Segundo algoritmo

1. Preparar un vaso de leche



# Diagramas de flujo

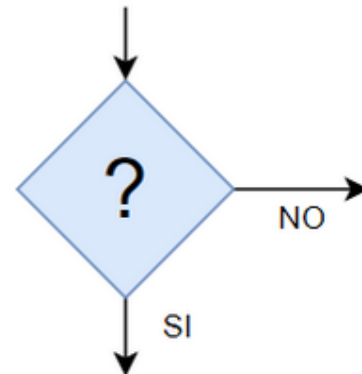
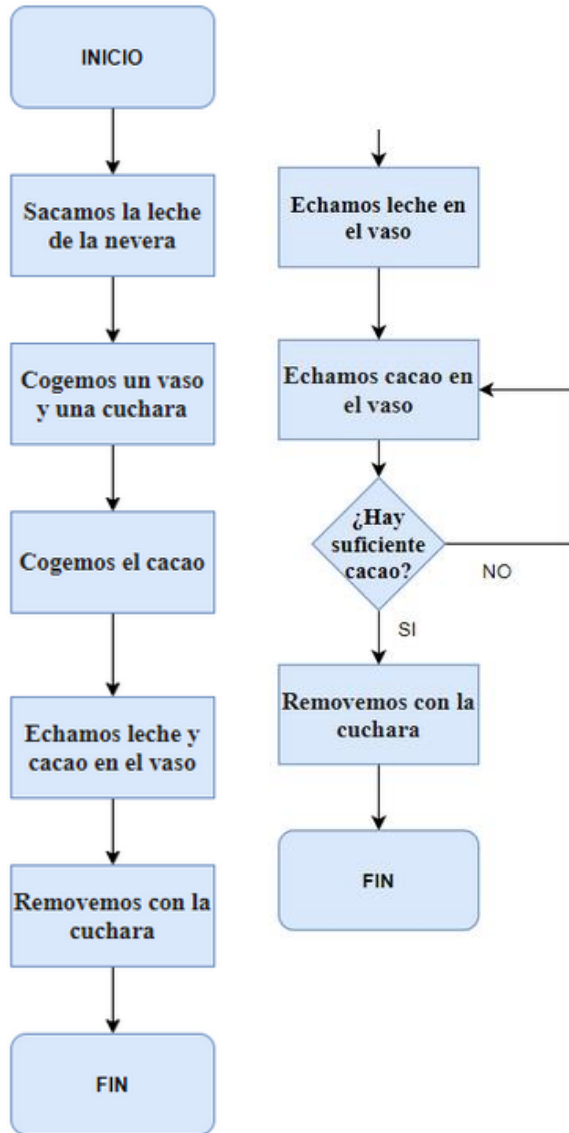


Los algoritmos pueden representarse de varias formas, con frases, gráficos etc..

Una de las maneras más habituales es con **diagramas de flujo**.

Generalmente tenemos que tomar decisiones (bucles if)

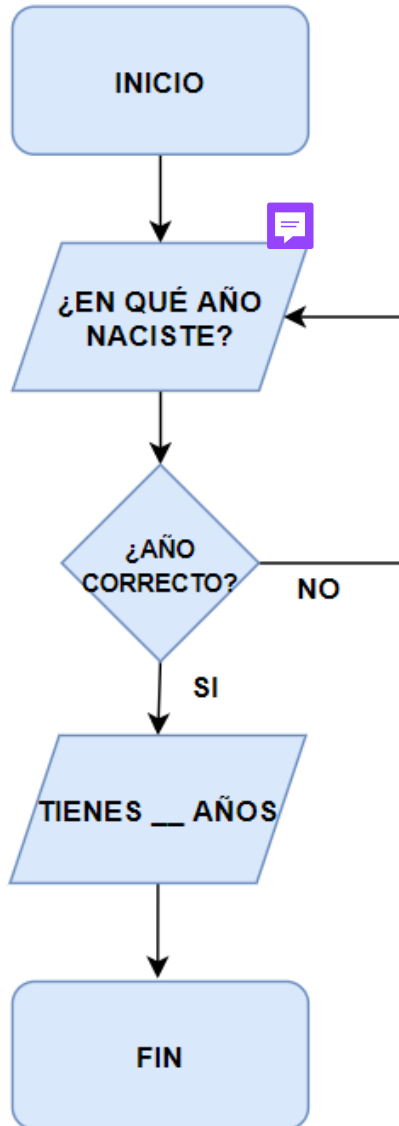
Hay que tener cuidado de no cometer errores pensando muy bien lo que se hace en cada uno de los bloques o pasos



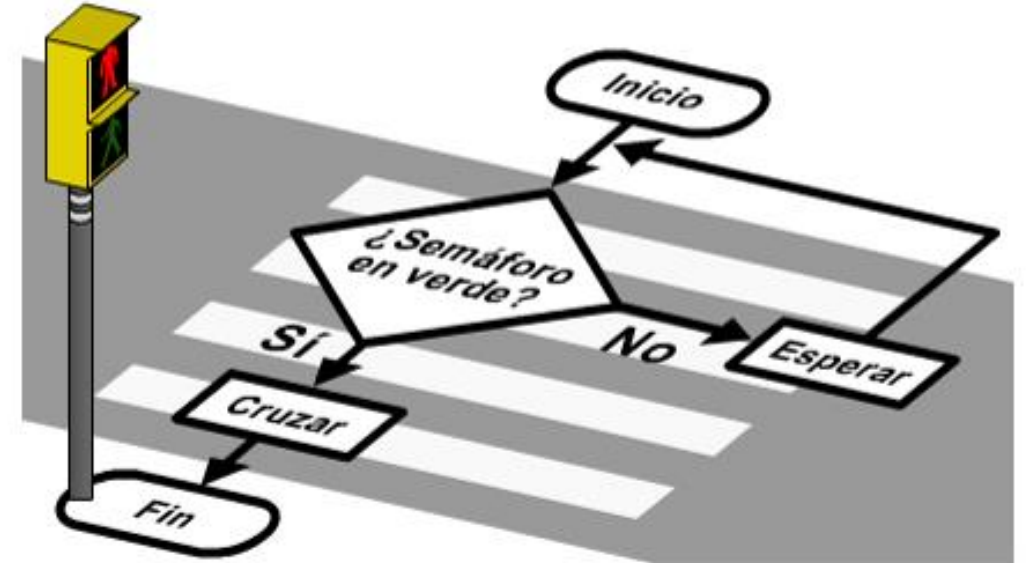
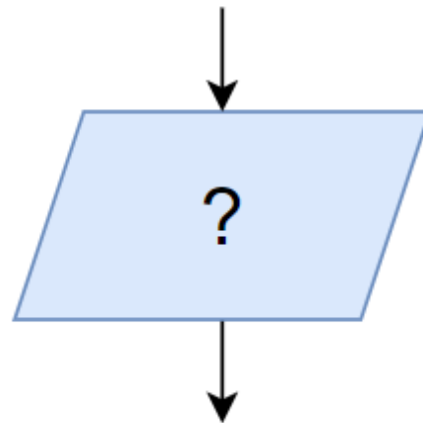
Símbolo	Nombre	Función
	Inicio / Final	Representa el inicio y el final de un proceso
	Linea de Flujo	Indica el orden de la ejecución de las operaciones. La flecha indica la siguiente instrucción.
	Entrada / Salida	Representa la lectura de datos en la entrada y la impresión de datos en la salida
	Proceso	Representa cualquier tipo de operación
	Decisión	Nos permite analizar una situación, con base en los valores verdadero y falso

Tabla de símbolos básicos utilizados en los diagramas de flujo

# Bloque de entrada y salida



En algunos algoritmos pedimos datos o sacamos datos por la pantalla, para ello utilizamos el bloque **Entrada / Salida**



Algoritmo que seguimos para cruzar un paso de cebra con semáforo.

# Ejercicio

Realiza el diagrama de flujo correspondiente al algoritmo con los siguientes pasos:

1. Te pregunta un número entre 1 y 10
2. Comprueba si está entre 1 y 10
3. Te pregunta otro número entre 1 y 10
4. Comprueba si está entre 1 y 10
5. Te da el resultado de sumas los dos números

**<https://app.diagrams.net/#>**

**Dale a decidir más tarde para  
no tener que registrarte**

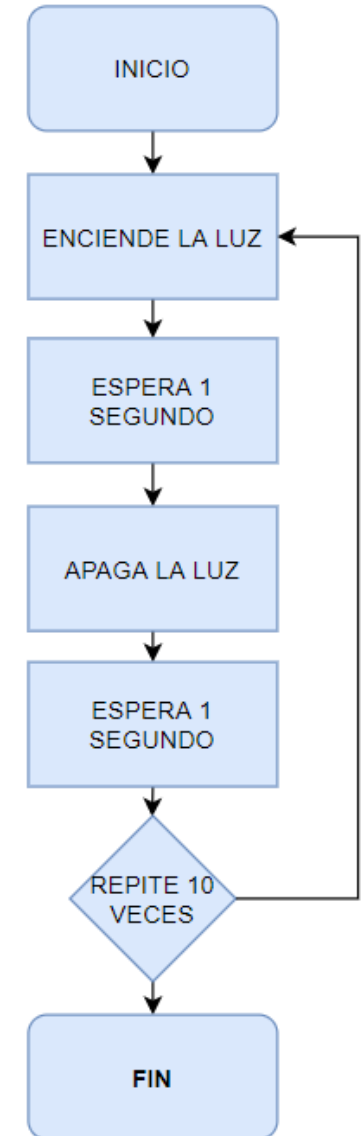
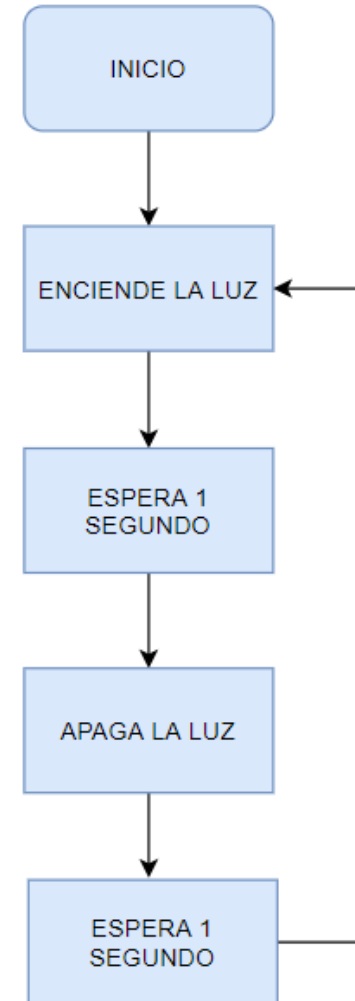
# Estructuras de repetición

A veces necesitamos repetir ciertas acciones varias veces por lo que existen **estructuras de repetición o bucles**. Dos tipos principales:

1. **Repetición para siempre (bucles while)**
  - No tiene bloque de FIN
2. **Repetición de un número determinado de veces (bucle for)**

Realiza el diagrama de flujo correspondiente al algoritmo con los siguientes pasos:

1. Mandar a un robot subir 20 escalones
2. Primero tiene que subir la pierna derecha y luego la pierna izquierda
3. Cuando llegue al último escalón, el robot tiene que detenerse



# Lenguaje de programación

El lenguaje de programación nos permite crear algoritmos en un lenguaje entendible para un ordenador.

Los equipos se comunican con **lenguaje máquina** (unos y ceros, datos binarios) -> **lenguaje de bajo nivel**

Lenguaje de programación -> **lenguaje de alto nivel** (permite ser traducido a lenguaje máquina y además ser comprensible para las personas)

**Compilador o intérprete:** traduce a código máquina las órdenes del lenguaje de programación





# Lenguaje de programación

