

# C1 - Introducción a la Programación y al Diseño de Software

---



Microsoft  
.NET

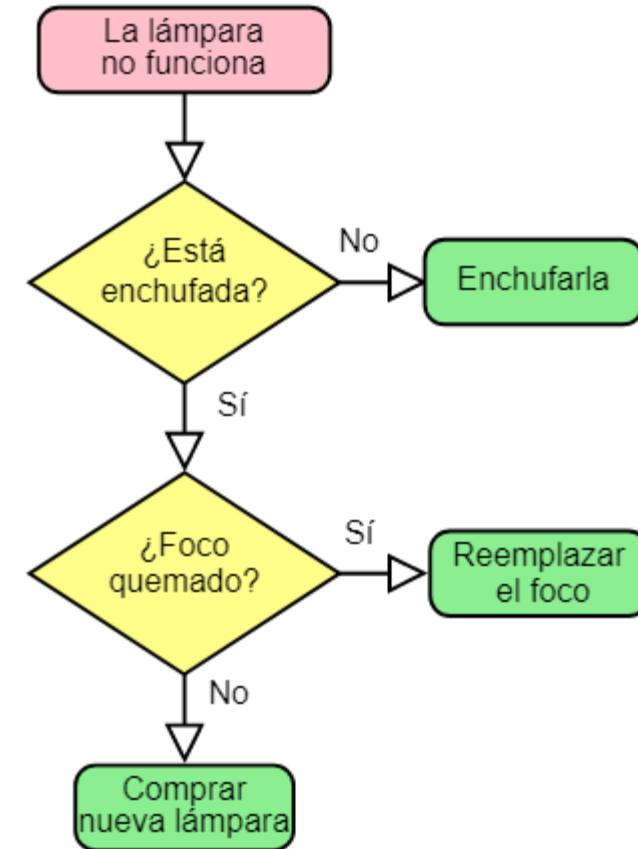
## 2 – Intro. al Desarrollo de Software

### 2.1 - Algoritmia y diagramas de flujo

## 2.1.1 Algoritmos

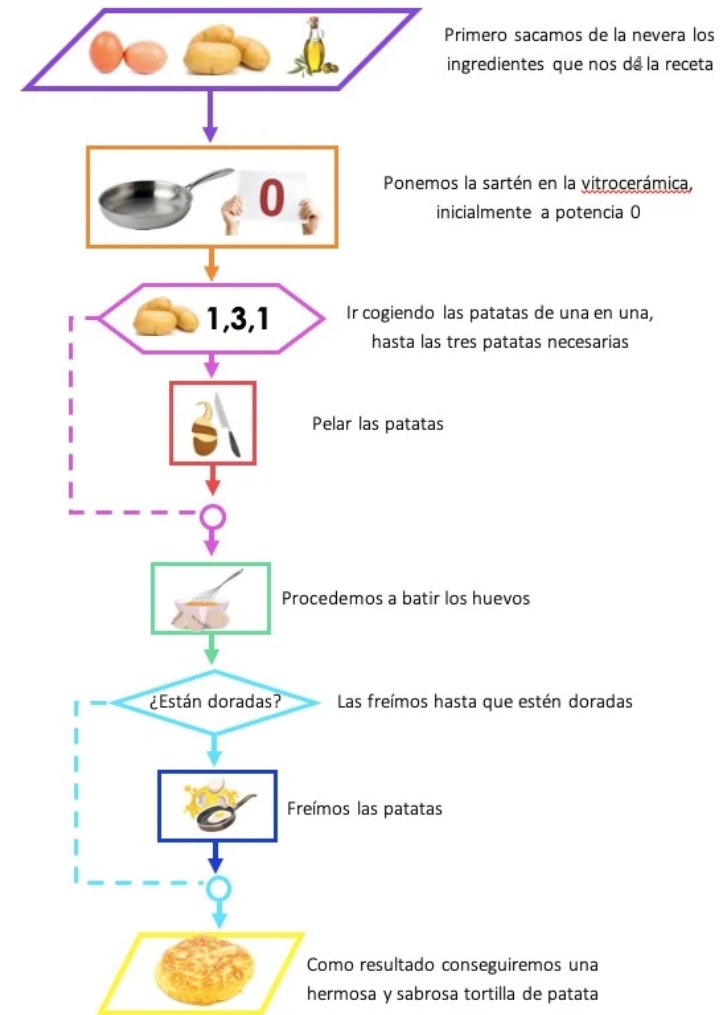
# Algoritmia

- Área de estudio cuyo objeto de estudio son los **algoritmos**.
- En computación **un algoritmo** es la serie de pasos a realizar para poder resolver una parte o la totalidad de un problema.
- Al hablar de un **problema computacional** se hace referencia a la necesidad de alcanzar un resultado buscado, dada cierta información, sabiendo que existe una o un **conjunto de soluciones capaces de ser encontradas utilizando las capacidades de procesamiento de una computadora**.



## ¿Qué es un algoritmo?

- Es un conjunto **ordenado** y **finito** de operaciones que permite hallar la solución de un problema.
- Podemos decir que un algoritmo es una "receta", ya que si se sigue de manera correcta encuentran un resultado en un tiempo acotado.
- Es un **conjunto preescrito de instrucciones** o reglas **bien definidas, ordenadas y finitas** que permite realizar una actividad mediante pasos sucesivos que no generen dudas a quien lo ejecute.
- Dados un **estado inicial** y una entrada, siguiendo los pasos sucesivos se llega a un **estado final** y se obtiene una **solución**.



# Ejemplo: Algoritmo para cambiar una bombilla

- Retirar la bombilla fundida
  - Colocar una escalera debajo de la bombilla
  - Subir la escalera
  - Desenroscar la bombilla en el sentido contrario de las agujas del reloj
  - Bajar la escalera
- Poner una bombilla nueva
  - Coger la bombilla nueva
  - Subir la escalera
  - Enroscar la bombilla en el sentido de las agujas del reloj
  - Bajar la escalera
  - Retirar la escalera



## Métodos algorítmicos

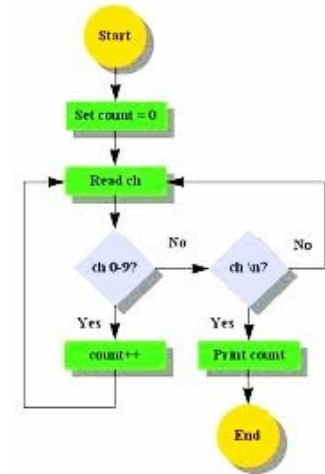
- Un **algoritmo** es **unívoco**: si se ejecuta varias veces el mismo algoritmo con un mismo conjunto de datos de entrada, siempre se obtiene la misma solución.
- Además, el resultado debe generarse en un tiempo finito.
- Los métodos que utilizan algoritmos se utilizan se denominan **métodos algorítmicos**, en oposición a los que implican algún juicio o interpretación, que se denominan **heurísticos**.
- Los métodos algorítmicos se pueden implementar fácilmente en computadoras; los métodos heurísticos no tanto, además de que no aseguran obtener siempre la misma salida dadas las mismas entradas.

# Notaciones para describir un algoritmo

- Método informal: Lenguaje natural
  - Ventajas: comprensible e intuitivo
  - Inconveniente: impreciso
- Método formal: Pseudocódigo
  - Lenguaje natural limitado y sin ambigüedad
- Diagrama: Diagrama de flujo u organigrama
  - Utiliza un conjunto de símbolos para representar cada estructura de control y mediante líneas de flujo se indica el orden en que se realiza el flujo lógico del algoritmo
  - Los pasos sucesivos se escriben en cajas con distintas formas según la estructura de control que representen

```

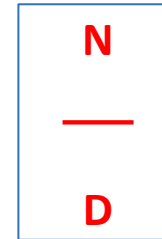
Inicio
  aprobados ← 0
  reprobados ← 0
  resultado ← 0
  estudiantes ← 1
  Mientras estudiantes ≤ 10
    Leer resultado
    Si resultado == 1 entonces
      aprobados ← aprobados + 1
    SiNo
      reprobados ← reprobados + 1
    FinSi
    estudiantes ← estudiantes + 1
  FinMientras
  Mostrar aprobados
  Mostrar reprobados
  Si aprobados > 8 entonces
    Mostrar "Aumentar la colegiatura"
  FinSi
Fin
  
```



## Ejemplo : Calcular el cociente de una división entera

- Sin usar la operación de división (seria hacer trampa...)
- En lenguaje natural

1. Leer el numerador N y el denominador D
2. Inicializar el cociente a 0 ( $C \leftarrow 0$ )
3. Si  $N < D$  ir al paso 7
4. Restar a N el valor de D ( $N \leftarrow N - D$ )
5. Incrementar en uno C ( $C \leftarrow C + 1$ )
6. Si  $N \geq D$  volver al paso 4
7. Mostrar el valor de C

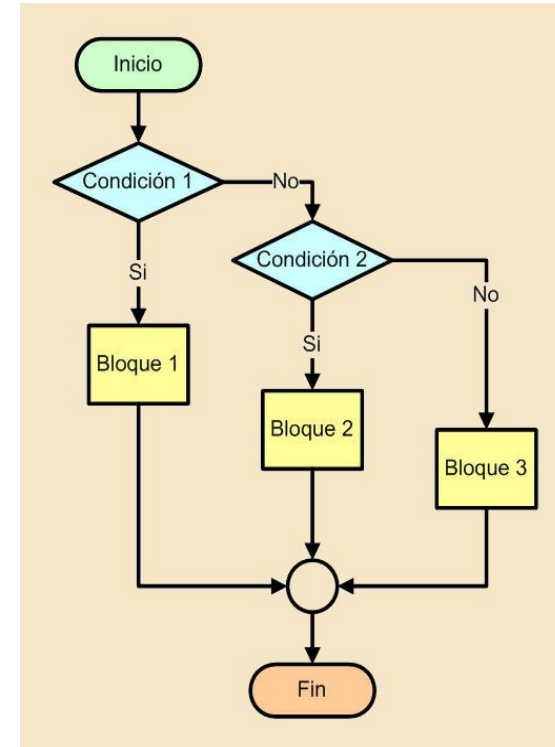




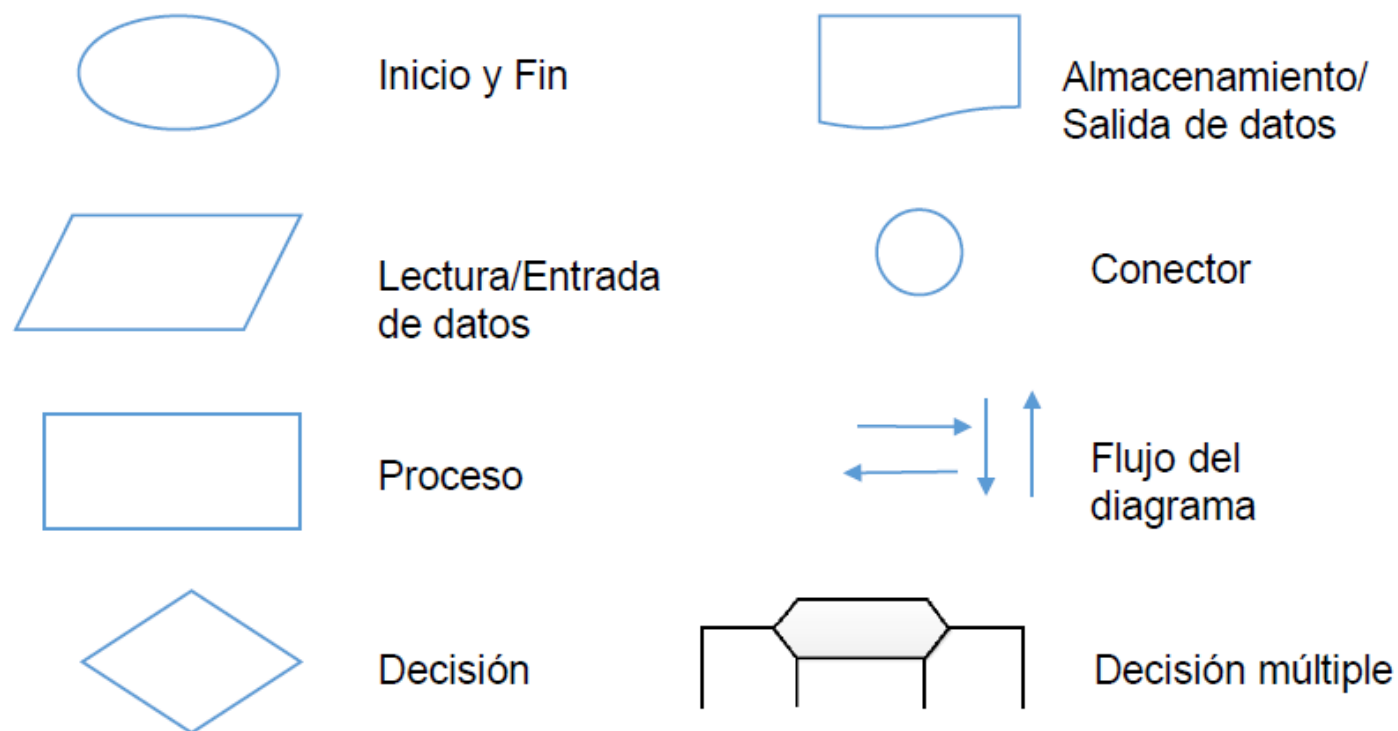
## 2.1.2 Diagramas de flujo

## Diagrama de flujo

- Existen distintas formas gráficas de representar un algoritmo, el diagrama de flujo fue una de las primeras empleadas.
- Un **diagrama de flujo** utiliza símbolos con significados bien definidos que representan los pasos del algoritmo, y representan el flujo de ejecución mediante flechas que conectan los puntos de inicio y de término.



# Formas utilizadas en los diagramas de flujo



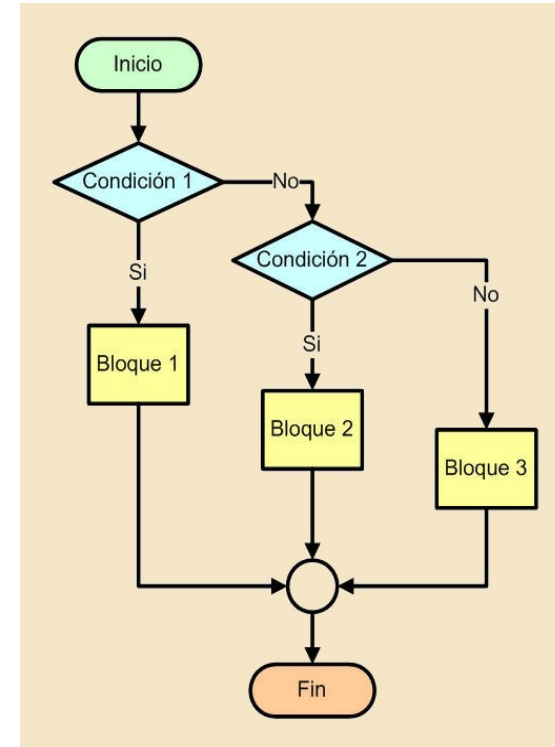
## Símbolos utilizados en los diagramas de flujo

+	Sumar
-	Menos
*	Multiplicación
/	División
=	Equivalencia
>	Mayor que
<	Menor que
≥	Mayor o igual que
≤	Menor o igual que
<> o !=	Diferente de
← o →	Asignación

«La definición de datos se da por supuesta, principalmente para variables sencillas»

## Reglas para su construcción

1. Todo diagrama de flujo debe tener un inicio y un fin
2. Las líneas utilizadas para indicar la dirección del flujo del diagrama deben ser rectas: verticales u horizontales
3. Todas las líneas utilizadas para indicar la dirección del flujo del diagrama deben estar conectadas a alguna forma
4. El diagrama de flujo debe construirse de arriba hacia abajo y de izquierda a derecha
5. La notación utilizada en el diagrama de flujo debe ser independiente de cualquier lenguaje de programación
6. Si la construcción del diagrama de flujo requiere más de una hoja, se deben dividir el diagrama

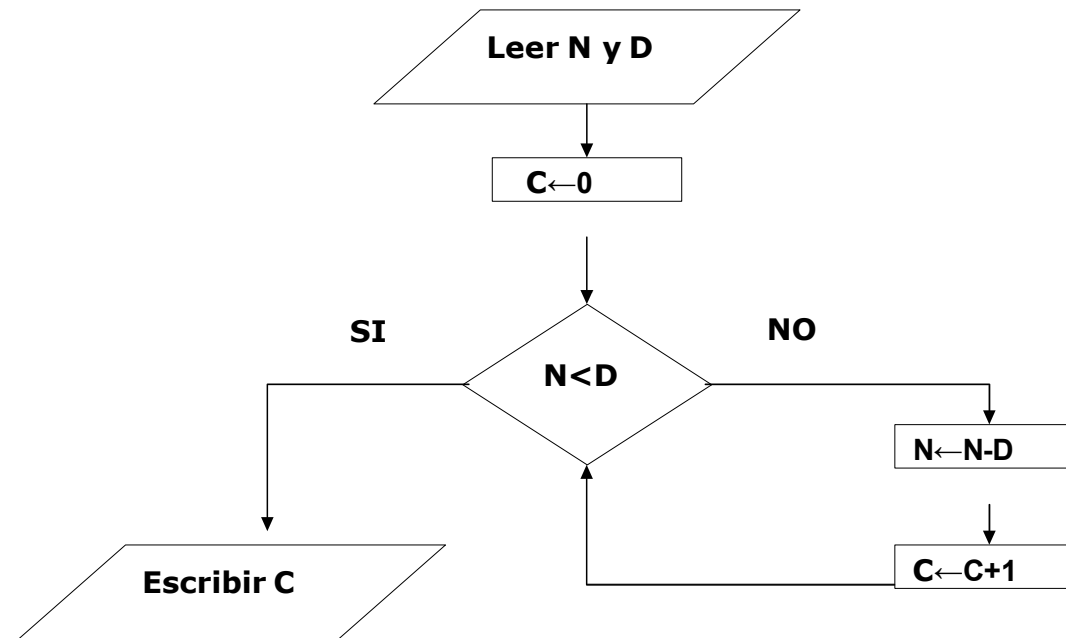


## Ejercicio E01 :

- Diseñar un diagrama de flujo de un algoritmo que realice una división entera
- (basarse sobre el algoritmo en lenguaje natural ya desarraollado)

*Solución en la diapo siguiente:  
NO MIRAR antes de haber  
desarrollado vuestra propia  
solución*

Solución :



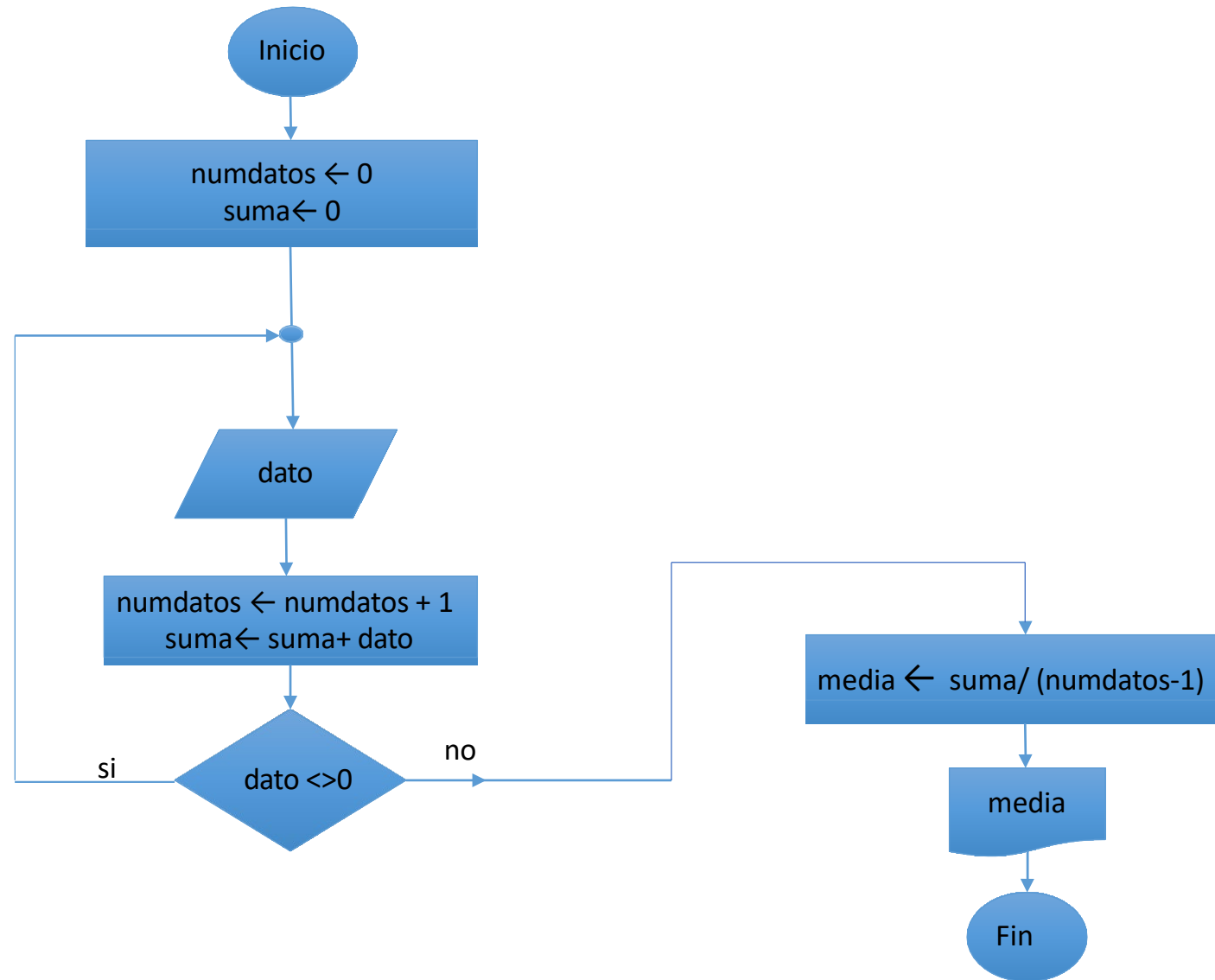
## Ejercicio E02:

- Calcular la media de una serie de números positivos, suponiendo que los datos se leen uno a uno.
- Un valor de cero como entrada indicará que se ha alcanzado el final de la serie de números positivos.
- Pasos para la resolución:
  - Analizar el problema
  - Dibujar el diagrama EN PAPEL (más fácil de retocar)

*Solución en la diapo siguiente:  
NO MIRAR antes de haber  
desarrollado vuestra propia  
solución*



Solución:



18

