



# Teoría 3

## Resolución de problemas

### Resolución de Problemas y Algoritmos

Ingeniería en Computación (TU y TFA)

Profesorado en Ciencias de la Computación (TU y TFA)



# Teoría 3

## Resolución de problemas



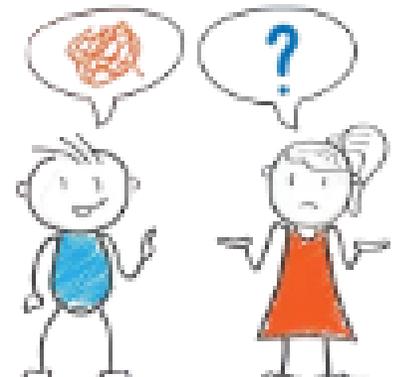
- ✓ Categorías de problemas: no computacionales y computacionales.
- ✓ **¿Cómo resolverlos?** Etapas en la resolución de problemas.
- ✓ **Etapa 1: Análisis del Problema.**
  - Entender el enunciado y comprender el problema.
  - Representación e interpretación de datos (entrada/proceso/salida).
- ✓ **Etapa 2: Diseño y desarrollo del algoritmo para resolver el problema.**
  - Descomposición de un problema en tareas.
  - Algoritmo que da solución a un problema: pseudocódigo y diagrama de flujo
- ✓ **Etapa 3: Codificación en un lenguaje de programación (programa).**
- ✓ **Etapa 4: Verificar/revisar la solución**

## Problema

es una situación que se nos presenta y que mediante la aplicación de un **algoritmo** pretendemos resolver.

## Algoritmo

es una sucesión **finita y ordenada** de acciones o pasos **precisos** que permiten resolver un problema.



# Categorías de Problemas

---



## **Problema no computacional**

son problemas que se resuelven mediante algoritmos que involucran acciones físicas o mentales sin requerir el uso de un computador. La “ejecución” de la solución recae en un procesador humano que utiliza herramientas o elementos del entorno para llevar a cabo la tarea.

## **Problema computacional**

Son problemas que se resuelven mediante algoritmos que luego serán ejecutados por computadoras o dispositivos digitales. Estos ejecutan una serie de acciones primitivas definidas en un lenguaje de programación, son aquellos que para resolverlos involucran, generalmente, operaciones aritméticas, lógicas y relacionales.

# Categorías de Problemas

---



## **Problema no computacional**

Por ejemplo: cebar mate, cambiar la llanta de un auto o buscar una palabra en el diccionario.

## **Problema computacional**

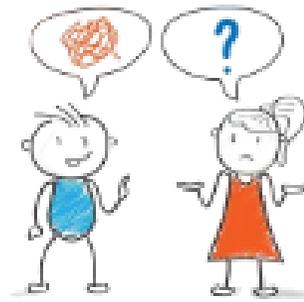
Por ejemplo: obtener la suma de dos números, encontrar el promedio de un conjunto de notas, encontrar el menor número en una secuencia, determinar si un número es par o impar, etc.

## Problemas NO computacionales

- Hacer un licuado de frutillas.
- Encontrar un camino para llegar a la universidad.
- Grabar y subir un video en tiktok.
- Organizar una fiesta de cumpleaños.

## Problemas computacionales

- Ordenar alfabéticamente una secuencia de letras.
- Mostrar el nombre del estudiante con menos faltas a clase y mayor nota en el primer parcial en una planilla.
- Calcular el promedio de todas las notas del año de un estudiante.
- Calcular la edad de una persona sabiendo el año de nacimiento.

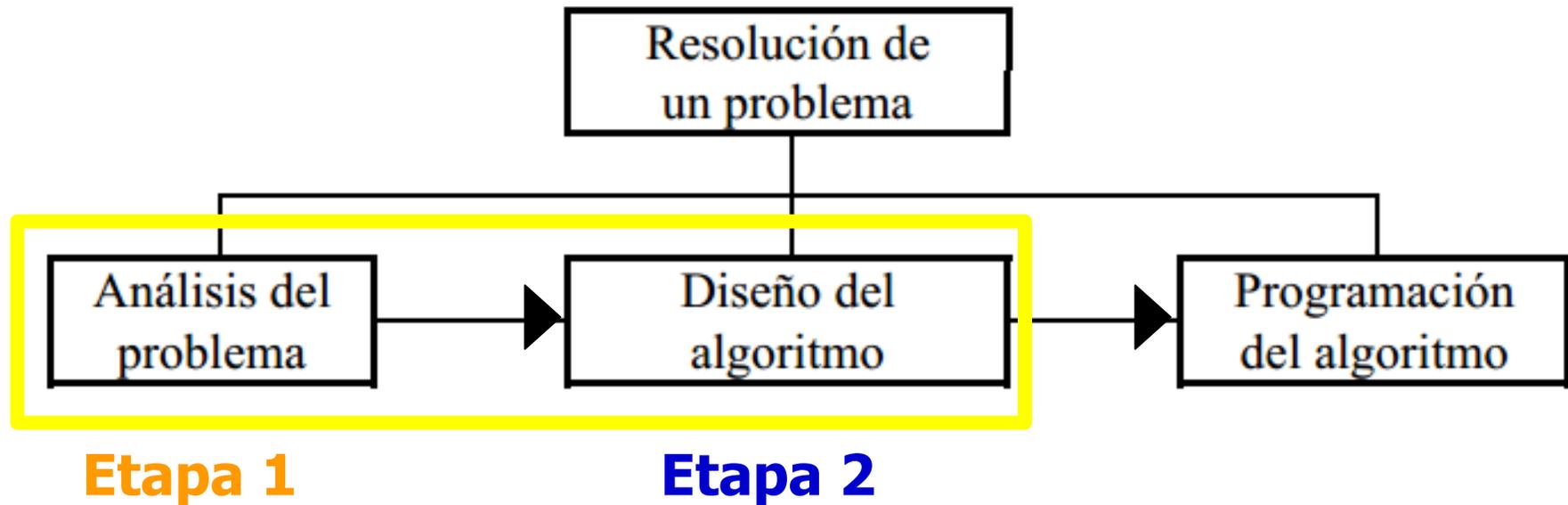


# Etapas en la Resolución de un Problema



## ¿Cómo resolver un problema?

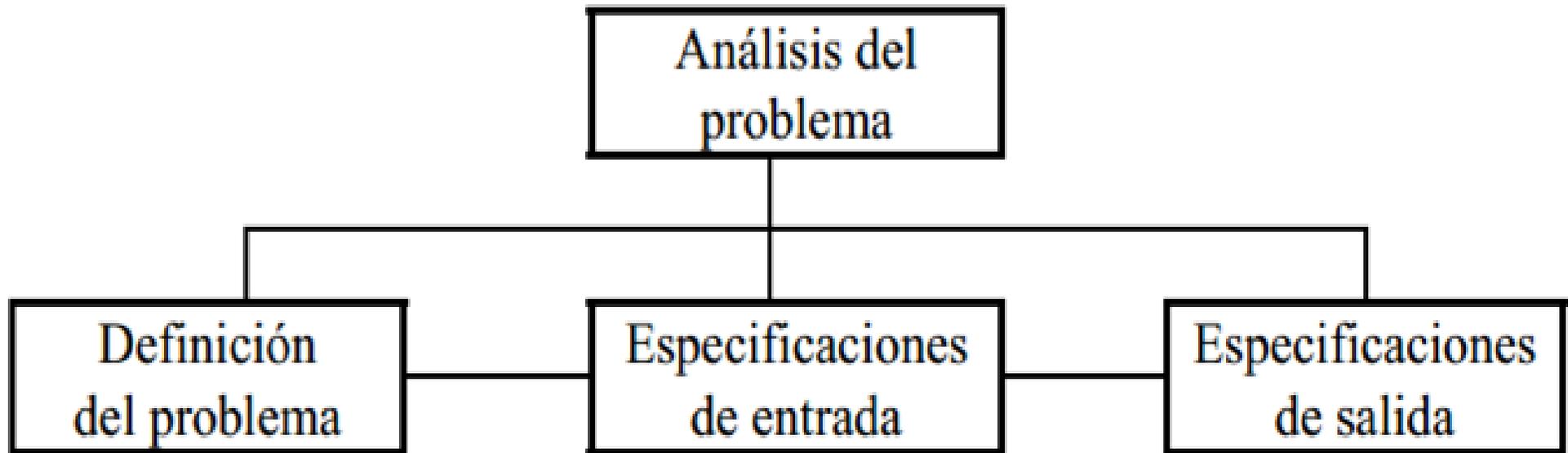
Existen diferentes métodos, todos ellos tienen en común el trabajar en etapas.



# Etapa 1: Análisis del Problema.



1. Entender el enunciado y comprender el problema.
2. Representación e interpretación de datos (entrada/proceso/salida).



# 1. Entender el enunciado y comprender el problema.

Un problema se define, generalmente, mediante un **enunciado** escrito en lenguaje natural.

**Enunciado:** En una vieja caja en el depósito de mi casa, encontré más de 200 cartas antiguas de mi abuelo. Las cartas eran de distintos tipos, tamaños y colores. La caja medía 10 pulgadas de largo, estaba muy sucia y arruinada. Necesito encontrar un algoritmo para ordenar todas las cartas.



?

Para **ayudarnos a entender correctamente el enunciado** nos podemos plantear preguntas, como por ejemplo:

1. ¿Entiendo cada palabra o siglas del enunciado?
2. ¿Qué categoría de problema es? ¿Cuál es el objetivo? ¿computacional o no computacional?
3. ¿Está completo el enunciado? ¿Hay datos que podrían estar faltando para resolver el problema?
4. ¿Hay datos en el enunciado que no son importantes para resolver el problema o no están claros?
5. ¿Necesito “googlear” o tener otros conocimientos para resolver el problema planteado?

**Enunciado:** En una vieja caja en el depósito de mi casa, encontré más de 200 cartas antiguas de mi abuelo. Las cartas eran de distintos tipos, tamaños y colores. La caja medía 10 pulgadas de largo, estaba muy sucia y arruinada. Necesito encontrar un algoritmo para ordenar todas las cartas.

1. ¿Entiendo cada palabra o siglas del enunciado?
2. ¿Qué categoría de problema es? ¿Cuál es el objetivo? ¿computacional o no computacional?
3. ¿Está completo el enunciado? ¿Hay datos que podrían estar faltando para resolver el problema?
4. ¿Hay datos en el enunciado que no son importantes para resolver el problema o no están claros?
5. ¿Necesito “googlear” o tener otros conocimientos para resolver el problema planteado?

Ejemplos de problemas:

1. Calcular la temperatura promedio durante la primera semana de agosto.
2. Calcular el porcentaje de H<sub>2</sub>O de una persona según su peso y edad.
3. Cocinar la comida favorita del cumpleaños para todos los invitados.
4. Informar el total de visualizaciones acumuladas en todos los videos del tiktokker más famoso de Argentina.
5. Dibujar una hoja en el pasillo de la universidad.

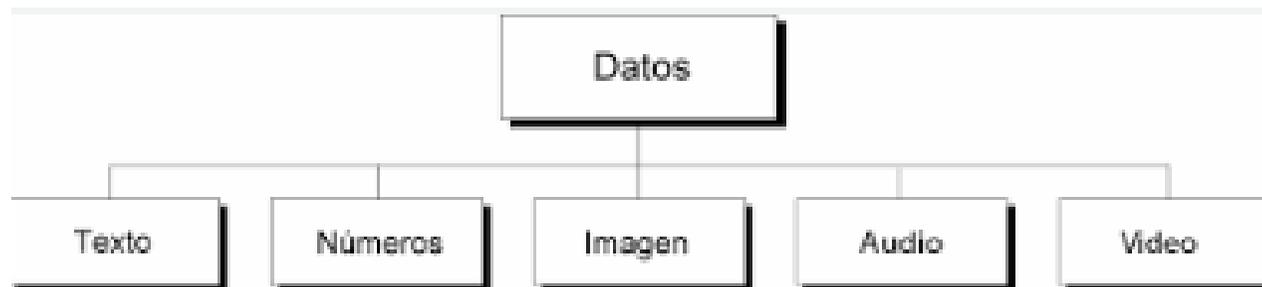
**Actividad 1:** Elegir 2 problemas y completar una tabla para cada uno:

<b>Problema número:</b>	
<b>¿Qué categoría de problema es?</b>	
<b>¿Qué palabras del enunciado no entiendo o no se su significado?</b>	
<b>¿Tiene todos los datos necesarios? Si su respuesta en No, ¿cuáles faltan?</b>	
<b>¿Sabe a qué tiene que llegar, un número, una palabra, una lista?</b>	

## 2. Representación e interpretación de datos.

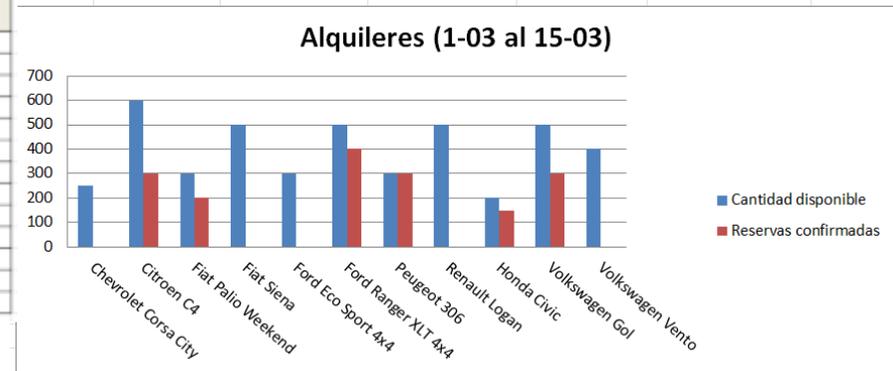
Saber **representar e interpretar datos** es muy importante para resolver problemas.

Es fundamental desarrollar la capacidad para analizar diferentes **representaciones de datos** y proponer procesos capaces de obtener información relevante para la toma de decisiones.

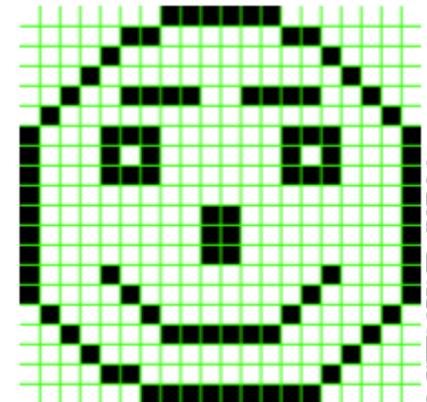


Un problema puede ser de distintos tipos como así también sus **datos** pueden tomar distintas representaciones.

	A	B	C	D	E	F	G	H	I	J	K
1		Total de Clases:	22								
2											
3		Estudiante	Asistencias	Inasistencias	% de Asistencia	Nota 1	Nota 2	Nota 3	Presentó certificado?	Promedio	tiene acceso a una recuperación por trabajo?
4		Estudiante 1	15	7	68,18%	6,00	5,00	7,00	si	6,00	Si tiene acceso
5		Estudiante 2	20	2	90,91%	7,00	8,00	7,00	si	7,33	Si tiene acceso
6		Estudiante 3	18	4	81,82%	6,00	0,00	7,00	no	4,33	No tiene acceso
7		Estudiante 4	20	2	90,91%	7,00	8,00	7,00	no	7,33	No tiene acceso
8		Estudiante 5	20	2	90,91%	8,00	8,00	7,00	no	7,67	No tiene acceso
9		Estudiante 6	20	2	90,91%	8,00	8,00	5,00	no	7,00	No tiene acceso
10		Estudiante 7	20	2	90,91%	8,00	8,00	6,00	si	7,33	Si tiene acceso
11		Estudiante 8	20	2	90,91%	9,00	0,00	4,00	si	4,33	Si tiene acceso
12		Estudiante 9	20	2	90,91%	9,00	9,00	7,00	no	8,33	No tiene acceso
13		Estudiante 10	10	12	45,45%	9,00	9,00	8,00	no	8,67	No tiene acceso
14		Estudiante 11	6	16	27,27%	5,00	10,00	5,00	no	6,67	No tiene acceso
15		Estudiante 12	9	13	40,91%	7,00	0,00	9,00	no	5,33	No tiene acceso
16		Estudiante 13	18	4	81,82%	4,00	10,00	8,00	si	7,33	Si tiene acceso



Licuada  
duraznos  
leche  
azúcar



# Entrada

Datos de entrada con los que se dispone

Licadora  
duraznos  
leche  
azúcar



# Proceso

Proceso o tratamiento a realizarse con los datos.

licuar  
y  
servir

2

# Salida

Datos de salida es la información solicitada.

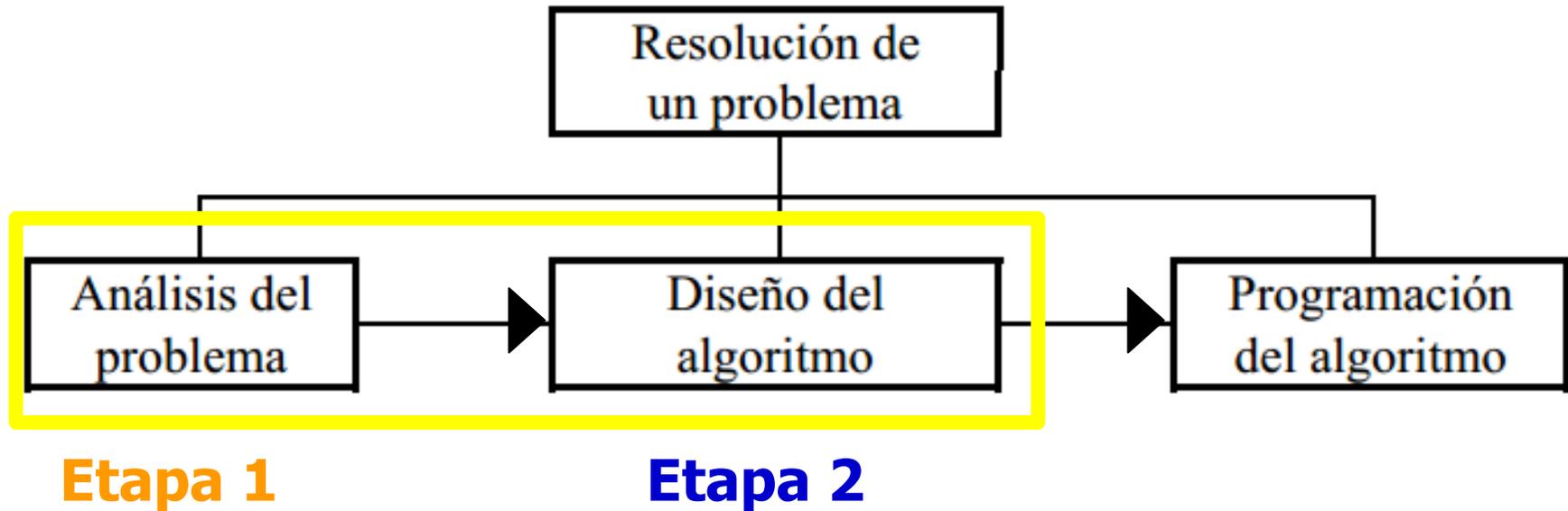
1

3

**Actividad 2:** Completar los valores para 1, 2 y 3.

1-	2-	3-
----	----	----

# Etapas en la Resolución de un Problema

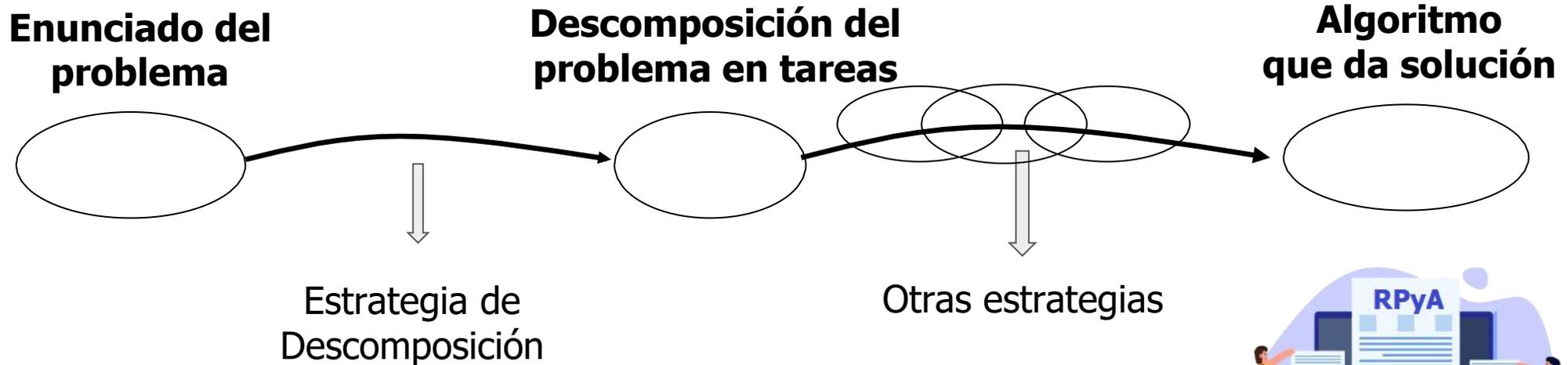


- 1. Entender el enunciado y comprender el problema.**
- 2. Representación e interpretación de datos (entrada/proceso/salida).**

# Etapa 2: Diseño y desarrollo del algoritmo



1. Descomposición de un problema en tareas.
2. Algoritmo que da solución a un problema: pseudocódigo y diagrama de flujo.



# 1. Descomposición de un problema en tareas.

**Estrategia de descomponer** un problema es dividirlo en partes más pequeñas y manejables que sean más simples de resolver.



$$\neg (P \rightarrow \neg(Q \wedge R)) \vee (P \rightarrow \neg S) \wedge (((P \wedge (\neg Q)) \wedge R) \vee S)$$

**Enunciado:** Determinar el presupuesto total para organizar una fiesta temática de "Rock argentino". El algoritmo debe calcular el costo total sumando catering, decoración y música. Tener en cuenta que el costo de catering es por persona, decoración y música es un valor fijo.

## **Versión 1 (Estrategia de descomponer el problema)**

**T1.** Definir los datos para catering por persona, decoración y música.

T2. Ingresar los datos para catering por persona, decoración y música.

T3. Sumar los costos de catering, decoración y música.

T4. Informar el resultado calculado en T3.

**Enumerar las tareas  
y escribir en infinitivo**

**¿Faltan datos?  
¿Son importantes?**

**Enunciado:** Determinar el presupuesto total para organizar una fiesta temática de "Rock argentino" para 100 personas. El algoritmo debe calcular el costo total sumando catering, decoración y música. Tener en cuenta que el costo de catering es por persona, decoración y música es un valor fijo.

### **Versión 1 (Estrategia de descomponer el problema)**

**T1.** Definir los datos para catering por persona, decoración y música.

T2. Ingresar los datos para catering por persona, decoración y música.

T3. Calcular el costo de catering para 100 personas.

T4. Sumar los costos de catering, decoración y música.

T5. Informar el resultado calculado en T4.

**Enunciado:** Determinar el presupuesto total para organizar una fiesta temática de "Rock argentino". El algoritmo debe calcular el costo de catering, decoración y música. Tener en cuenta que el costo de decoración y música es un valor fijo.

**¿Faltan datos?**

**¿Son importantes?**

**LOS PIDO (Definir e ingresar)**

### **Versión 1 (Estrategia de descomponer el problema)**

**T1.** Definir los datos para costo de catering por persona, decoración, música y **cantidad de personas.**

T2. Ingresar los datos para costo de catering por persona, decoración, música y **cantidad de personas.**

T3. Calcular el costo de catering para **la cantidad de personas ingresadas.**

T4. Sumar los costos de catering, decoración y música.

T5. Informar el resultado calculado en T4.

**Actividad 3:** Descomponer el problema planteado en el siguiente enunciado.

**Enunciado:** Según el plan de estudio de la carrera se deben cursar 3 materias: Álgebra, Cálculo y Resolución de problemas y algoritmos. Quiero saber cuántas horas totales en todo el cuatrimestre asistiré a clase, teniendo en cuenta las horas por semana que se cursa cada materia y la cantidad de semanas del cuatrimestre.

### **Versión 1 (Estrategia de descomponer el problema)**

**T1.** Definir los datos para ...

T2. Ingresar los datos para ...

T3. Sumar la cantidad ...

T4. Multiplicar ....

T5. Informar el resultado calculado en T4.

# ¿Qué es un Algoritmo?

Un algoritmo es una **secuencia finita y ordenada de acciones primitivas** que pueden ser ejecutadas por un **procesador** y que lleva a la solución de un problema planteado en un **enunciado**, valiéndose de un **conjunto de recursos necesarios** para su ejecución.

**Descomposición  
del problema**

Versión 1

**Algoritmo  
que da solución**

Versión  
final



**Problema:** es una situación a resolver.

**Enunciado:** descripción detallada de un problema a resolver.

**Procesador:** Toda entidad capaz de comprender las acciones primitivas de un algoritmo y ejecutar el trabajo indicado por el mismo.

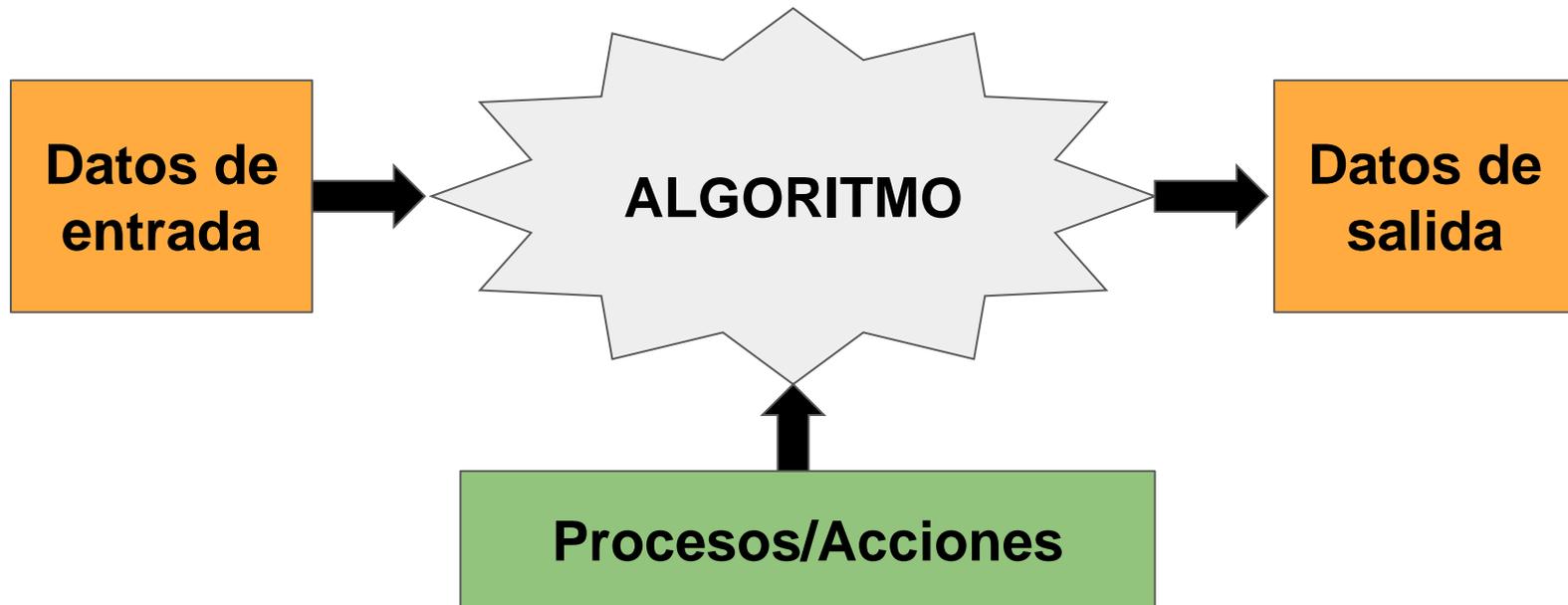
**Ambiente:** El conjunto de todos los recursos necesarios para la ejecución de un algoritmo.

**Algoritmo:** conjunto de **acciones (pasos)** que debe realizar el procesador para resolver el problema planteado.

# ¿Qué se considera ACCIÓN?

Es un **evento** que modifica al **ambiente** establecido.

Una acción es **primitiva** si dicha acción pueda ser comprendida por el procesador sin mayor información adicional.



# Características de un algoritmo

- 1. Finito:** un algoritmo debe terminar después de ejecutar un número finito de pasos.
- 2. Preciso:** cada paso en un algoritmo debe estar definido con precisión, esto es, la acción a seguir no debe ser ambigua, sino rigurosamente especificada. Considerando el ambiente en el que se trabaja, cada **acción primitiva** debe significar, una sola tarea, bien determinada y sin lugar a dudas.
- 3. Efectivo:** Un algoritmo debe llevar a la solución del problema.
  - **Entrada:** Todo algoritmo tiene **un comienzo**. Se considera como entrada el conjunto de **datos requerido** para resolver un problema dado.
  - **Salida:** Todo algoritmo tiene **un fin**. La salida es un resultado que se obtiene al aplicar el algoritmo al conjunto de datos de entrada.

**Tiene que tener**

**Problema:** realizar un cálculo aritmético.

**Enunciado:** Escribir un algoritmo para calcular y mostrar el promedio de 3 números cualesquiera.

**Procesador:** la computadora.

**Ambiente:** los 3 números y el promedio. **¿Resuelve el enunciado?**

**Algoritmo:** Inicio  
Definir num1, num2, num3, prom de tipo reales  
Escribir "Ingrese el primer número:"  
Leer num1  
Leer num2  
Escribir "Ingrese el segundo número:"  
 $promedio = num1 + num2 + num3 / 3$   
Escribir "Ingrese el tercer número:"  
Leer num3  
Escribir "El promedio es:", prom

Fin

**Llega a la salida esperada  
esperada**



# ¿Cómo saber si un algoritmo resuelve el enunciado? Es decir, llega a la salida esperada.

La **ejecución de un algoritmo** es el proceso de seguir paso a paso todas las instrucciones definidas en el mismo, transformando los datos de entrada en los resultados deseados. Es decir, es la puesta en práctica del algoritmo para resolver un problema de forma sistemática y ordenada.

Al hacerlo, podemos encontrar y corregir errores, como usar datos incorrectos o seguir un orden equivocado.

En resumen, la **ejecución de un algoritmo** permite detectar y corregir errores, lo que es fundamental para garantizar la precisión en cualquier tarea o proceso automatizado.

**Enunciado:** Escribir un algoritmo para calcular y mostrar el promedio de 3 números cualesquiera.

**Algoritmo:**

- Inicio
- Definir num1, num2, num3, prom de tipo reales
- Escribir "Ingrese el primer número:"
- Leer num1
- Leer num2
- Escribir "Ingrese el segundo número:"
- $promedio = num1 + num2 + num3 / 3$
- Escribir "Ingrese el tercer número:"
- Leer num3
- Escribir "El promedio es:", prom
- Fin

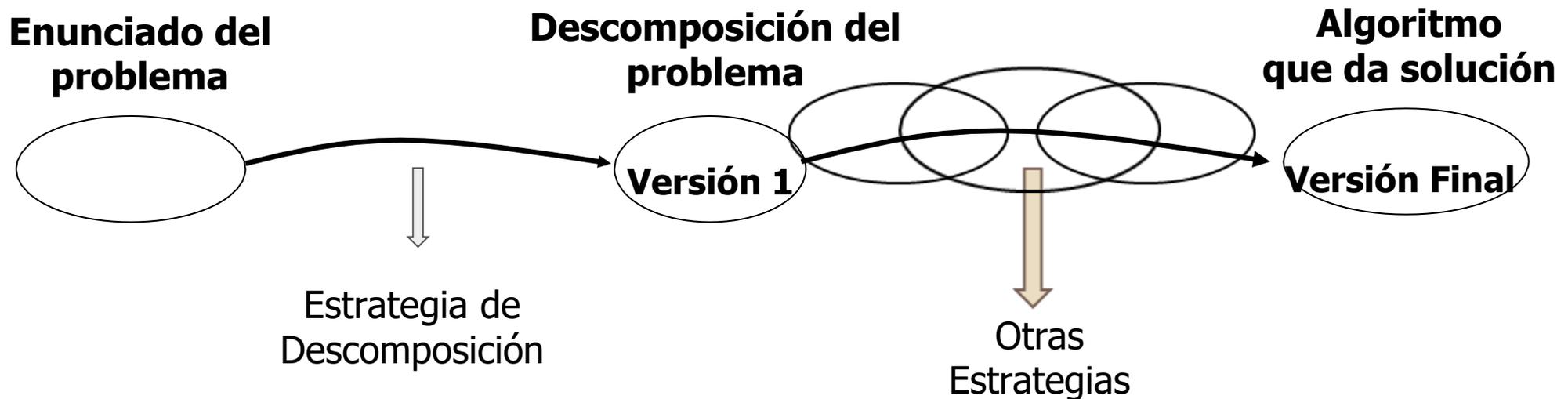
**Actividad 4:** Ejecutar el algoritmo e identificar los 5 errores que impiden que se resuelva correctamente el problema planteado. Por ejemplo:

**Error 1: el orden de las sentencias (acciones)** Leer num2 y Escribir "Ingrese el segundo número:". Debería estar primero el cartel con lo que se solicita ingresar.

Para definir un algoritmo en la materia:

**Versión 1:** aplicar estrategia de descomposición del problema en tareas más simples.

**Versión Final:** algoritmo que da solución al problema teniendo en cuenta las acciones primitivas y los estados (inicial y final si se han dado).



**Enunciado:** Escribir un algoritmo para calcular y mostrar el promedio de 3 números cualesquiera.

<b>Versión 1</b>	<b>Algoritmo Final</b>
T1:	<p><b>Inicio</b></p> <p>valor1, valor2, valor3, total como números reales</p> <p>Solicitar "Introduzca el primer valor:" y asignar a valor1</p> <p>Solicitar "Introduzca el segundo valor:" y asignar a valor2</p> <p>Solicitar "Introduzca el tercer valor:" y asignar a valor3</p> <p><math>total = (valor1 + valor2 + valor3) / 3</math></p> <p>Imprimir "El promedio de los valores es: ", total</p> <p><b>Fin</b></p>

**Actividad 4:** Escribir la versión 1 como una descomposición en tareas del enunciado dado. Ayudarse con el Algoritmo final.

## **Etapa 2: Diseño y desarrollo del algoritmo para resolver el problema.**

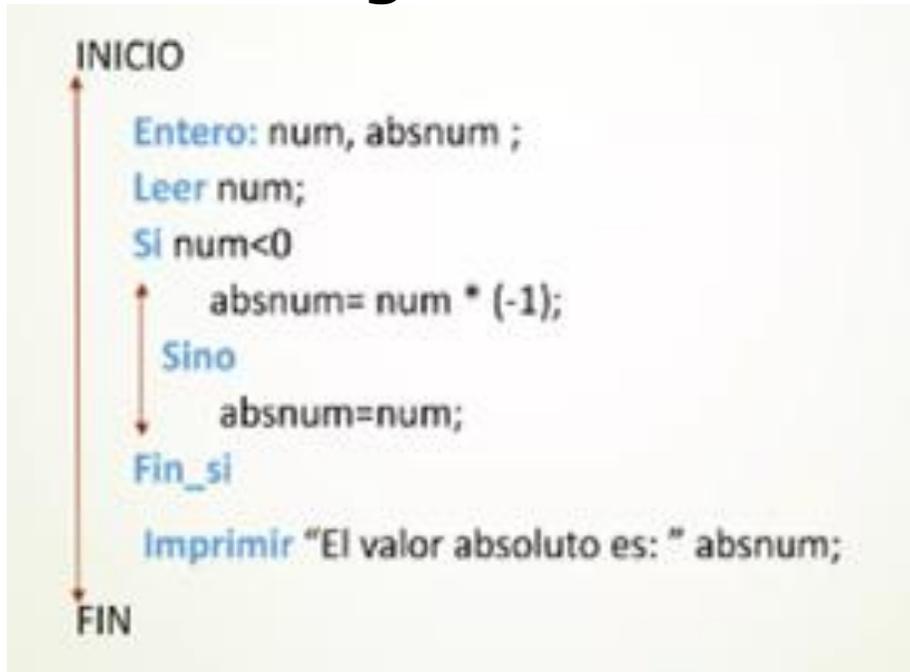
1. Descomposición de un problema en tareas.
2. Algoritmo que da solución a un problema: **pseudocódigo y diagrama de flujo.**

# Etapa 2: Pseudocódigo y diagrama de flujo

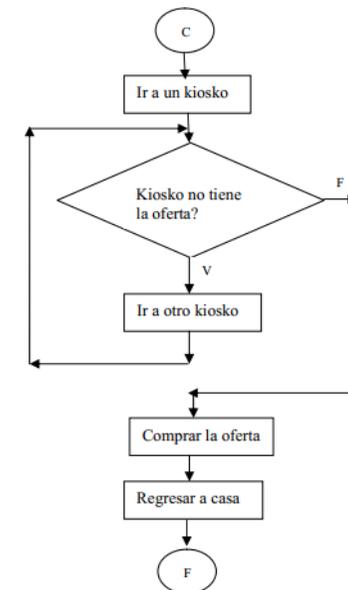


## CONTINUARÁ...en la Teoría 4

### Representación NO GRÁFICA de un algoritmo



### Representación GRÁFICA de un algoritmo



## Etapa 2: Pseudocódigo y diagrama de flujo

---



**CONTINUARÁ...en la Teoría 4**

**Pseudocódigo** es usado para la **representación no gráfica** de algoritmos de la forma más detallada posible y a su vez lo más parecida posible a lenguaje que posteriormente se utilizará para la codificación del mismo.

**Diagrama de flujo** es utilizado para **representar gráficamente** algoritmos, facilita la visualización del flujo de ejecución del mismo.

# Teoría 3

## Resolución de problemas



- ✓ Categorías de problemas: no computacionales y computacionales.
- ✓ **¿Cómo resolverlos?** Etapas en la resolución de problemas.
- ✓ **Etapa 1: Análisis del Problema.**
  - Entender el enunciado y comprender el problema.
  - Representación e interpretación de datos (entrada/proceso/salida).
- ✓ **Etapa 2: Diseño y desarrollo del algoritmo para resolver el problema.**
  - Descomposición de un problema en tareas.
  - Algoritmo que da solución a un problema: **pseudocódigo y diagrama de flujo.**



## Teoría 3

### Resolución de problemas

¡Ya podemos comenzar con el  
Práctico 3!