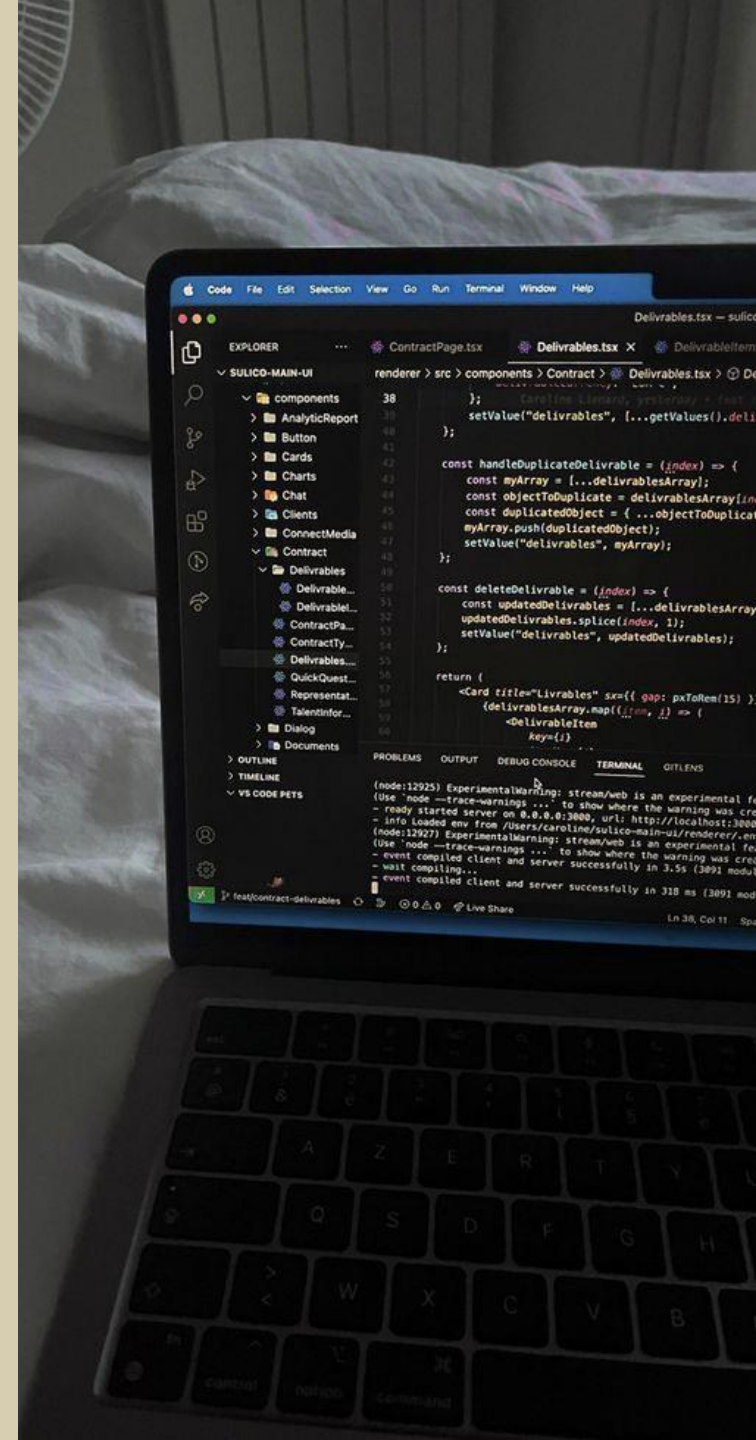


# !difficode

M1L1: ¿Qué es la programación  
y por qué es importante la  
lógica?

# 1. Definición de programación

- La programación es el proceso de escribir instrucciones o **código** para que una computadora ejecute tareas específicas. Estas instrucciones, conocidas como **algoritmos**, deben estar bien estructuradas y seguir una secuencia lógica para que el resultado sea el esperado.



# 1. Definición de programación

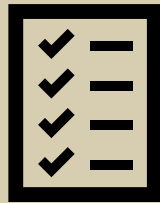
## Definición básica



# 1. Definición de programación

## Ejemplo básico

- Si le pides a un amigo que te ayude a preparar un sándwich, le das una lista de pasos a seguir (por ejemplo, tomar pan, agregar queso, poner jamón, etc.). Este sería el equivalente a un algoritmo. Cada instrucción debe ser clara para que se entienda bien, lo cual es muy similar a la forma en que programamos.



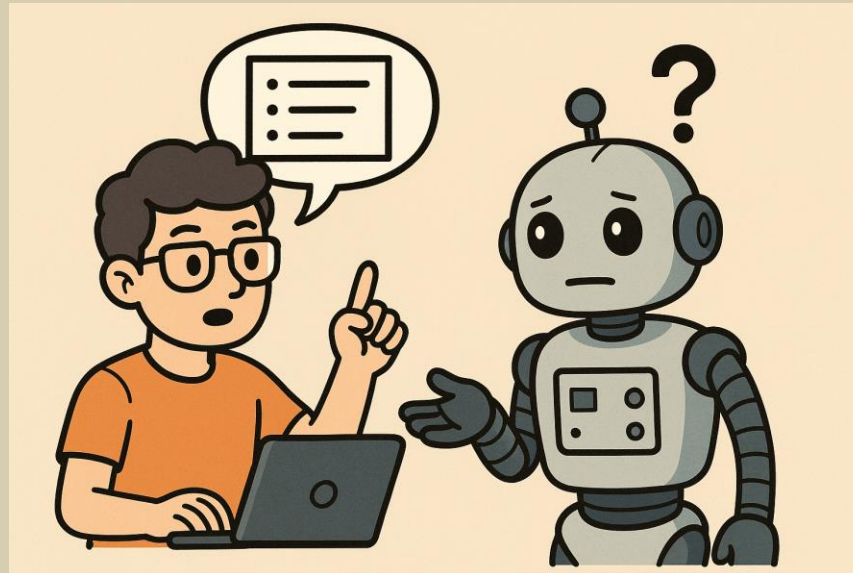
=

**Algoritmo**

# 1. Definición de programación

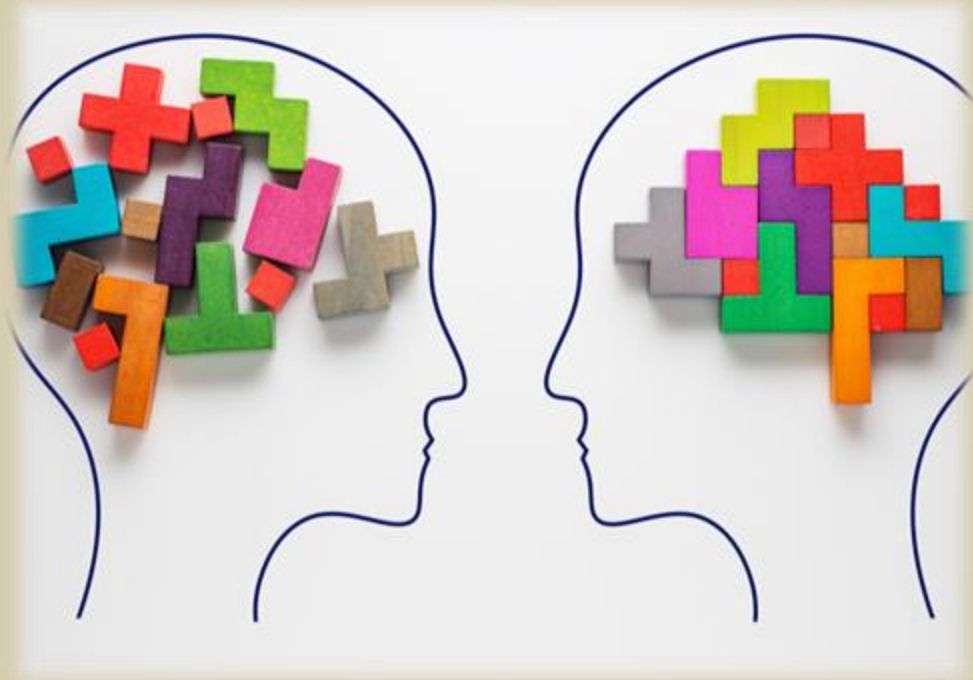
## Analogía

- Programar es como dar instrucciones detalladas a un robot para realizar una tarea específica. Si las instrucciones no son claras, el robot (o la computadora) no hará lo que esperamos.



## 2. La importancia de la lógica en la programación

La **lógica** es la capacidad de razonar de manera ordenada y coherente. En programación, es la base para tomar decisiones correctas en cada paso del algoritmo.



## 2. La importancia de la lógica en la programación

### ¿Por qué la lógica es crucial?

- Sin lógica, los algoritmos no tienen sentido y no lograrían cumplir su objetivo. La programación no se trata solo de escribir líneas de código, sino de pensar cómo resolver problemas de manera eficiente y clara. La lógica permite descomponer un problema grande en partes más pequeñas y manejables.



## 2. La importancia de la lógica en la programación

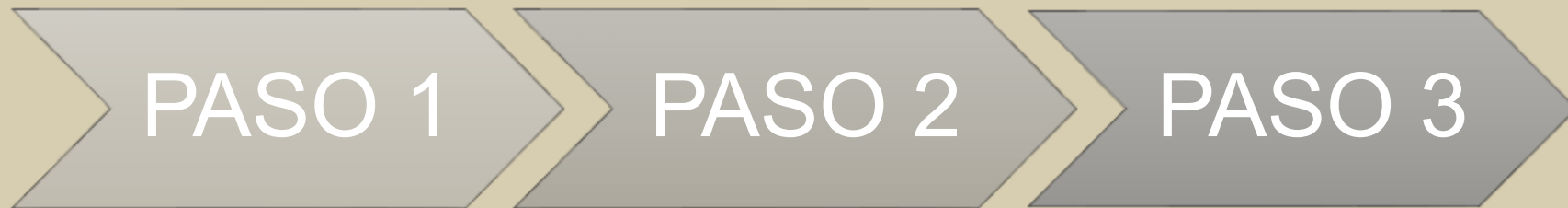
- **La toma de decisiones lógica** es fundamental. Por ejemplo, si queremos que un programa verifique si una persona es mayor de edad, necesitamos usar una declaración lógica como "Si la edad es mayor o igual a 18, entonces..." (Esto es una **estructura condicional**).



### 3. Relación entre Lógica y Algoritmos

#### Algoritmo

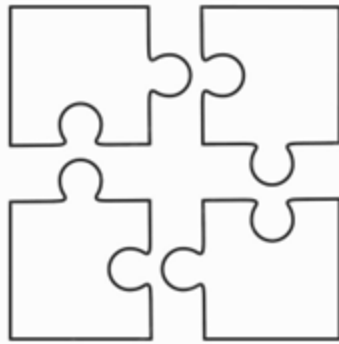
Es una serie de pasos ordenados para resolver un problema específico.



# 3. Relación entre Lógica y Algoritmos

## Lógica

Es lo que nos permite construir estos pasos de manera coherente y efectiva.



# 3. Relación entre Lógica y Algoritmos

## **El algoritmo como conjunto de pasos lógicos:**

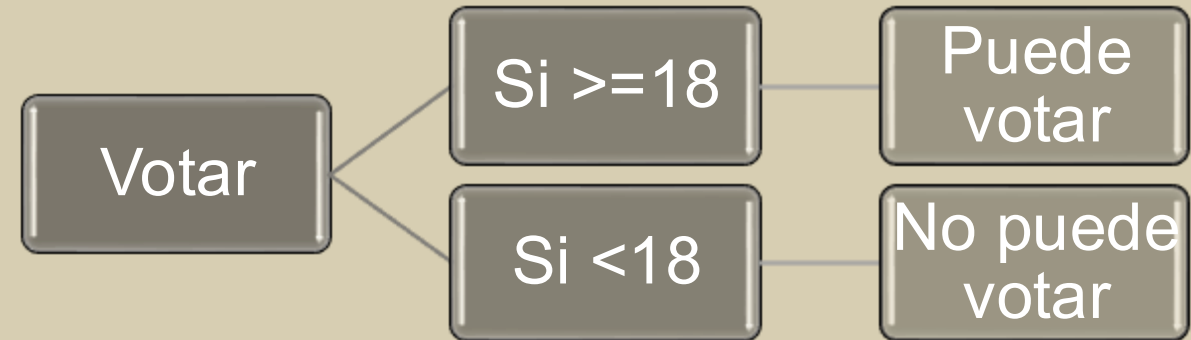
La creación de un algoritmo involucra diseñar la secuencia correcta de pasos para resolver un problema. Para lograr esto, necesitamos pensar de manera lógica, asegurándonos de que cada paso sea claro y esté bien fundamentado.

### 3. Relación entre Lógica y Algoritmos

#### Ejemplo de algoritmo lógico:

Imagina que queremos un algoritmo que determine si una persona es apta para votar. Los pasos lógicos serían:

- Verificar si la persona tiene 18 años o más.
- Si la persona cumple con la edad mínima, entonces está apta para votar.
- Si no cumple con la edad mínima, entonces no está apta para votar.



## 4. Pensamiento Algorítmico

El **pensamiento algorítmico** es la habilidad de descomponer problemas en pasos lógicos y ordenados. Es como tener una caja de herramientas que nos permite resolver cualquier tipo de problema.



# 4. Pensamiento Algorítmico

- **Descomposición**

Un problema grande puede parecer abrumador, pero al descomponerlo en partes más pequeñas y manejables, lo hace mucho más fácil de abordar. El pensamiento algorítmico nos enseña cómo dividir los problemas en pasos pequeños y claros, y luego solucionarlos de manera eficiente.

## 4. Pensamiento Algorítmico

- **Abstracción**

Es importante aprender a abstraer detalles innecesarios. Por ejemplo, al resolver un problema, no nos importa cómo la computadora va a realizar una suma, solo nos importa que la computadora haga la suma correctamente. La abstracción permite que los programadores se concentren en lo esencial.

# 4. Pensamiento Algorítmico

- **Generalización**

La lógica también implica reconocer patrones y generalizarlos. Una vez que hemos resuelto un tipo de problema, podemos aplicar la misma lógica para resolver otros problemas similares.



# 5. La Relación entre la Lógica y los Diagramas de Flujo

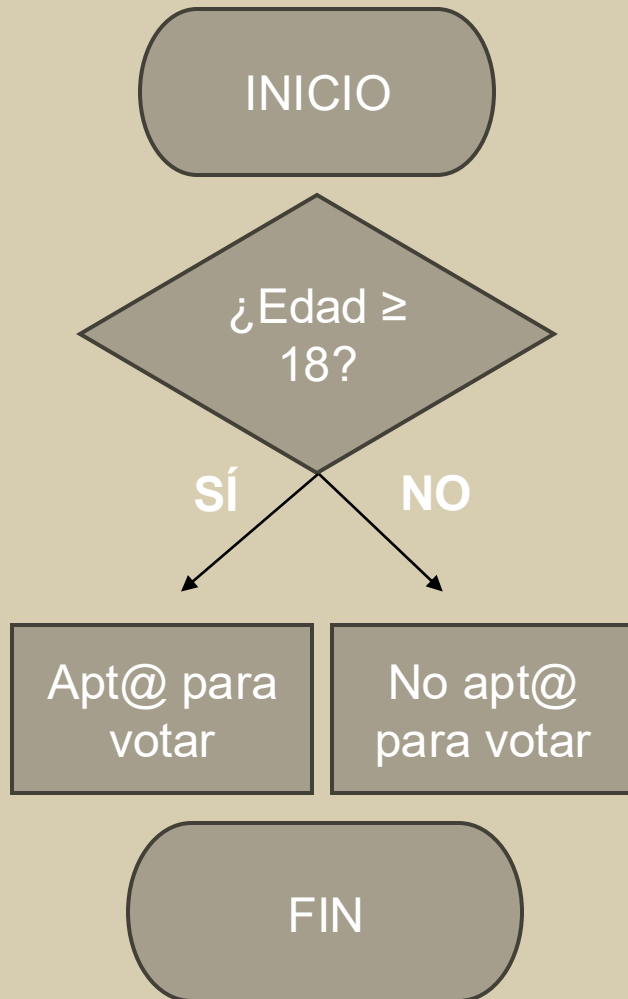
Los **diagramas de flujo** son representaciones gráficas que nos ayudan a visualizar la lógica de un algoritmo. Este tipo de diagramas se utiliza para descomponer problemas y algoritmos de manera estructurada.

# 5. La Relación entre la Lógica y los Diagramas de Flujo

## **¿Qué es un diagrama de flujo?**

Un diagrama de flujo es una representación gráfica de los pasos y decisiones involucrados en un algoritmo. Utiliza símbolos específicos para representar acciones, decisiones y la secuencia de ejecución.

## 5. La Relación entre la Lógica y los Diagramas de Flujo



### Ejemplo de diagrama de flujo:

Vamos a visualizar el algoritmo de "verificar si alguien es mayor de edad". El diagrama de flujo podría tener los siguientes pasos:

- Un símbolo de inicio.
- Un rombo (decisión) que verifica si la edad es mayor o igual a 18.
- Dos flechas saliendo del rombo: una hacia una acción que dice "Apt@ para votar" si es verdadero, y otra hacia una acción que dice "No apt@ para votar" si es falso.
- Un símbolo de fin.

# 5. La Relación entre la Lógica y los Diagramas de Flujo

## **Importancia de los diagramas de flujo:**

- Ayudan a organizar visualmente el pensamiento, permitiendo ver el camino claro de la solución antes de implementarlo en un lenguaje de programación.

## 6. Conclusión de la Lección

La programación no es solo sobre escribir código, sino sobre **pensar lógicamente** y descomponer los problemas en partes manejables. A través de algoritmos bien diseñados y con una base lógica sólida, podemos resolver cualquier desafío que se nos presente en el mundo de la programación.

# 6. Conclusión de la Lección

## **Resumen de los puntos clave:**

- La programación es un proceso de dar instrucciones claras a la computadora.
- La lógica es la clave para organizar estas instrucciones y garantizar que el algoritmo funcione correctamente.
- El pensamiento algorítmico nos permite descomponer problemas grandes y complejos en pasos pequeños y manejables.
- Los diagramas de flujo nos ayudan a visualizar y planificar algoritmos de manera estructurada.

# 7. Ejercicios

## Pregunta 1

¿Qué es la programación?

a) Es la acción de dar instrucciones claras a una computadora para que realice tareas específicas.

b) Es un lenguaje que solo entienden las computadoras.

c) Es la capacidad de escribir código sin necesidad de saber lo que hace.

d) Es solo una herramienta para diseñar videojuegos.

# 7. Ejercicios

## Pregunta 1

¿Qué es la programación?

a) Es la acción de dar instrucciones claras a una computadora para que realice tareas específicas.

b) Es un lenguaje que solo entienden las computadoras.

c) Es la capacidad de escribir código sin necesidad de saber lo que hace.

d) Es solo una herramienta para diseñar videojuegos.



# 7. Ejercicios

## Pregunta 2

¿Por qué es importante la lógica en la programación?

a) Porque ayuda a organizar las instrucciones de manera que la computadora las entienda.

b) Porque permite tomar decisiones que afectan el flujo del programa.

c) Porque hace que el código se vea bonito.

d) Porque descompone los problemas grandes en pasos más manejables.

# 7. Ejercicios

## Pregunta 2

¿Por qué es importante la lógica en la programación?

a) Porque ayuda a organizar las instrucciones de manera que la computadora las entienda.

b) Porque permite tomar decisiones que afectan el flujo del programa.

c) Porque hace que el código se vea bonito.

d) Porque descompone los problemas grandes en pasos más manejables.

# 7. Ejercicios

## **Pregunta 3**

Completa la siguiente afirmación con las palabras adecuadas.

"El pensamiento algorítmico es \_\_\_\_\_. Esta habilidad es útil porque facilita encontrar una solución clara y efectiva."

# 7. Ejercicios

## **Pregunta 3**

La habilidad de organizar un problema grande en problemas más pequeños. Por ejemplo, para organizar una fiesta, los pasos podrían incluir comprar los alimentos, invitar a los amigos y decorar el lugar.

# 7. Ejercicios

## Pregunta 4

¿Qué representa un diagrama de flujo?

a) Una representación gráfica de los pasos de un algoritmo.

b) El código real que una computadora ejecuta.

c) Una lista de instrucciones desordenadas.

d) Un juego de simulación para probar algoritmos.

# 7. Ejercicios

## Pregunta 4

¿Qué representa un diagrama de flujo?

a) Una representación gráfica de los pasos de un algoritmo.

b) El código real que una computadora ejecuta.

c) Una lista de instrucciones desordenadas.

d) Un juego de simulación para probar algoritmos.

## 7. Ejercicios

### **Pregunta 5**

Completa las siguientes frases con las palabras correctas:

\_\_\_\_\_ : nos permite centrarnos solo en los aspectos esenciales de un problema, ignorando detalles innecesarios.

\_\_\_\_\_ : nos ayuda a aplicar una solución que ya conocemos a problemas similares, sin tener que empezar desde cero cada vez.

# 7. Ejercicios

## Pregunta 5

Completa las siguientes frases con las palabras correctas:

**Abstracción:** nos permite centrarnos solo en los aspectos esenciales de un problema, ignorando detalles innecesarios.

**Generalización:** nos ayuda a aplicar una solución que ya conocemos a problemas similares, sin tener que empezar desde cero cada vez.



# 7. Ejercicios

## Pregunta 6

¿Qué relación existe entre **lógica y algoritmos**?

Selecciona todas las respuestas correctas:

a) Los algoritmos dependen de la lógica para que cada paso sea claro y coherente.

b) La lógica solo se utiliza para la programación avanzada y no tiene importancia al principio.

c) La lógica permite tomar decisiones dentro de un algoritmo, como verificar si una condición es verdadera o falsa.

d) Los algoritmos no pueden funcionar sin una estructura lógica adecuada.

## 7. Ejercicios

### Pregunta 6

¿Qué relación existe entre **lógica y algoritmos**?

Selecciona todas las respuestas correctas:

a) Los algoritmos dependen de la lógica para que cada paso sea claro y coherente.

b) La lógica solo se utiliza para la programación avanzada y no tiene importancia al principio.

c) La lógica permite tomar decisiones dentro de un algoritmo, como verificar si una condición es verdadera o falsa.

d) Los algoritmos no pueden funcionar sin una estructura lógica adecuada.