

# Proyecto Final: Adivina el Número

## Computador adivina el número del usuario

Integrantes: Jhonatan Macancela

Curso: Lógica de programación

Profesor: [Nombre]

Fecha: 26 de Agosto del 2025

# Objetivo del programa:

- Que el computador adivine el número que el usuario piensa.
- Aplicar la técnica de búsqueda binaria.
- Reforzar conceptos de lógica, programación y algoritmos.

# FUNCIONAMIENTO

- 1. El usuario piensa un número entre 1 y 100.
- 2. El computador hace suposiciones.
- 3. El usuario responde: “mayor”, “menor” o “igual”.
- 4. El proceso se repite hasta que el computador acierta.



# ALGORITMO PRINCIPAL

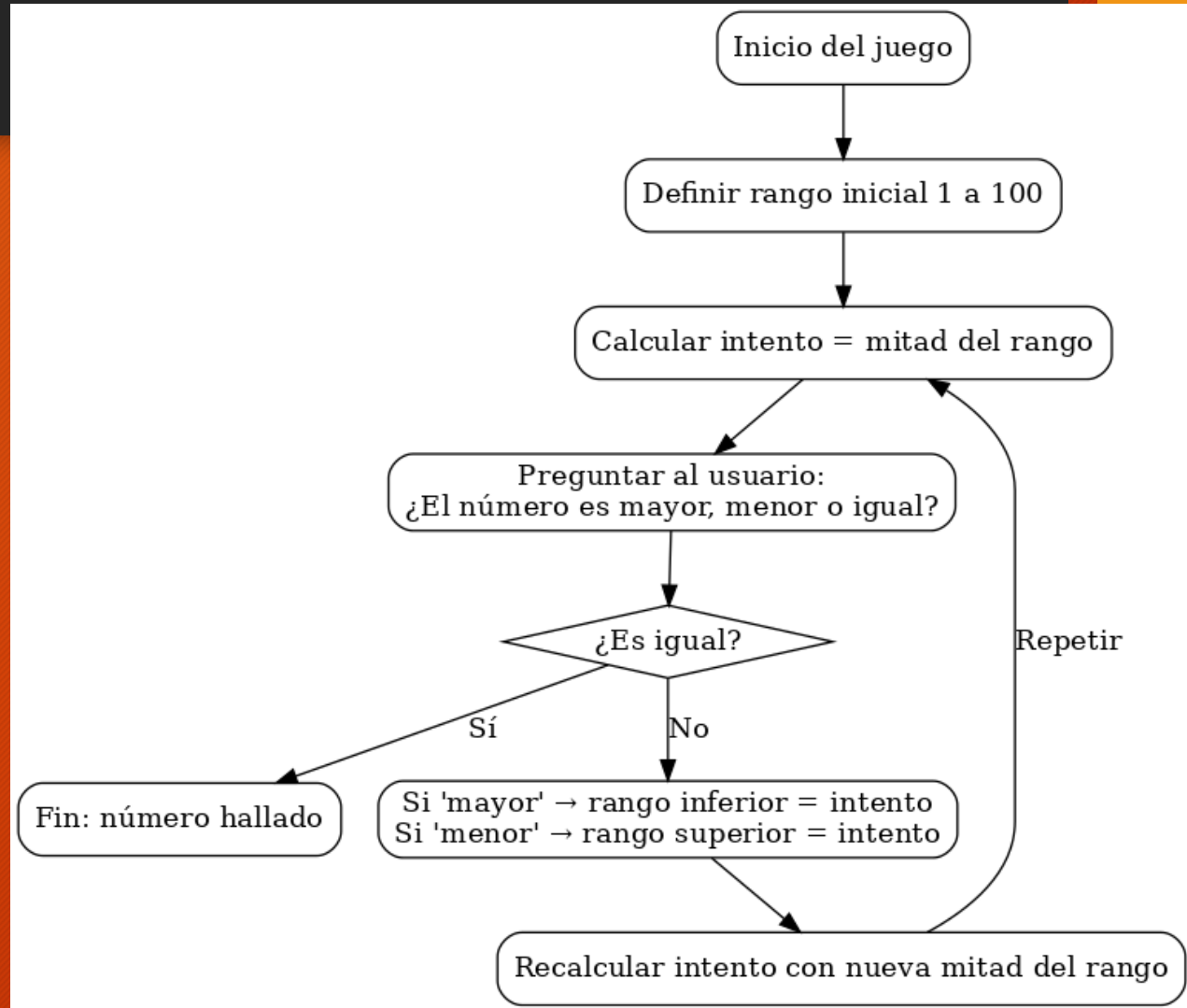
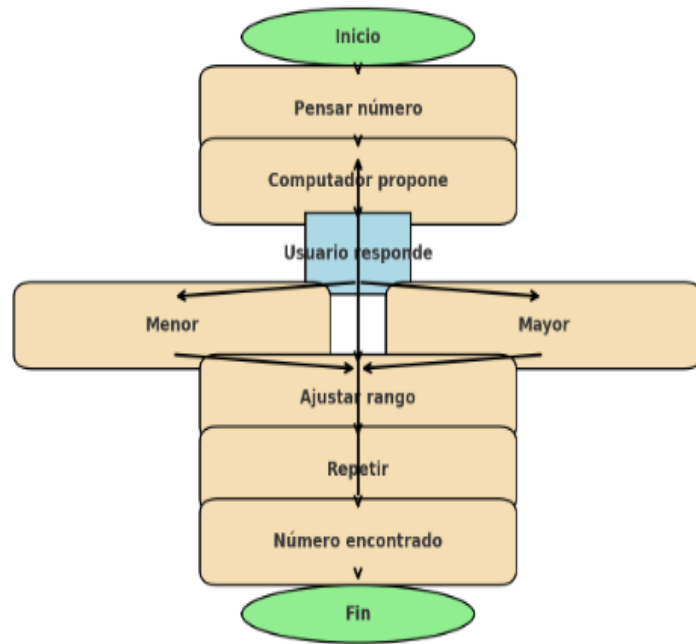
- Algoritmo: Búsqueda Binaria
- 1. Definir límite inferior y superior.
- 2. Calcular número en la mitad.
- 3. Preguntar al usuario si es mayor, menor o igual.
- 4. Ajustar los límites según la respuesta.
- 5. Repetir hasta encontrar el número.

# PSEUDOCÓDIGO

- Inicio
- $\text{limite\_inferior} \leftarrow 1$
- $\text{limite\_superior} \leftarrow 100$
- Repetir
- $\text{intento} \leftarrow (\text{limite\_inferior} + \text{limite\_superior}) / 2$
- Preguntar al usuario: ¿Es mayor, menor o igual?
- Si respuesta = “mayor” entonces
- $\text{limite\_inferior} \leftarrow \text{intento} + 1$
- Si respuesta = “menor” entonces
- $\text{limite\_superior} \leftarrow \text{intento} - 1$
- Hasta que respuesta = “igual”
- Fin

# DIAGRAMA DE FLUJO

Diagrama de Flujo: Adivina el Número





# CÓDIGO PRINCIPAL

- Ejemplo en Python:
- `limite_inferior = 1`
- `limite_superior = 100`
- `respuesta = ""`
- `while respuesta != "igual":`
- `intento = (limite_inferior + limite_superior) // 2`
- `respuesta = input(f"¿Es {intento}? (mayor/menor/igual): ").lower()`
- 
- `if respuesta == "mayor":`
- `limite_inferior = intento + 1`
- `elif respuesta == "menor":`
- `limite_superior = intento - 1`

# EJEMPLO DE EJECUCIÓN

- Usuario piensa: 37
- Computador propone: 50 → respuesta: menor
- Computador propone: 25 → respuesta: mayor
- Computador propone: 37 → respuesta: igual
- ¡El computador adivinó en 3 intentos!



# CONCLUSIONES

- El programa aplica búsqueda binaria de forma eficiente.
- Refuerza conceptos de iteración, condicionales y lógica.
- Demuestra la interactividad entre usuario y computadora.
- Es un ejemplo práctico y didáctico de programación en Python.