

Juego de Adivinar un Número con Estrategia Dividir-y-Vencer

Juan Esteban Villamil Ardila^a

^a*Pontificia Universidad Javeriana, Bogotá, Colombia*

Abstract

Este documento presenta un algoritmo basado en la estrategia dividir y vencer para el juego de adivinar un número, donde un humano actúa como el pensador y el algoritmo como el adivinador, este algoritmo reduce el conjunto de posibles números en cada iteración hasta adivinar el número pensado.

Keywords: algoritmo, número, juego, adivinar, dividir y vencer.

1. Análisis del problema

El juego de adivinar un número implica que el adivinador debe reducir eficientemente el conjunto de posibles números a través de preguntas al pensador, la estrategia dividir y vencer es una buena elección para este problema ya que permite descartar rápidamente la mitad de los posibles números en cada iteración.

2. Diseño del problema

El pensador elige un número natural y el adivinador debe adivinarlo proporcionando conjuntos de números, el pensador responde si el número pensado es mayor menor o igual a cada número en el conjunto proporcionado.

2.1. Valores de entrada

- n : Número pensado por el humano.
- C : Conjunto de números proporcionado por el adivinador.

2.2. Valores de salida

- Respuestas del pensador indicando si el número pensado es mayor, menor o igual a cada número en el conjunto.

Email address: villamila_j@javeriana.edu.co (Juan Esteban Villamil Ardila)

3. Algoritmo de solución

El algoritmo se basa en la estrategia dividir y vencer. En cada iteración el adivinador selecciona el número medio del rango actual de posibles números, luego presenta este número al pensador quien responde con la relación del número pensado respecto al número propuesto, el adivinador ajusta el rango de posibles números en consecuencia y repite el proceso hasta adivinar el número.

Require: n : Número pensado por el humano

```

1: procedure ADIVINARNUMERO( $n$ )
2:    $inicio \leftarrow 1$ 
3:    $fin \leftarrow \infty$ 
4:   while Verdadero do
5:      $numero\_propuesto \leftarrow \lfloor \frac{inicio+fin}{2} \rfloor$ 
6:     Enviar  $numero\_propuesto$  al pensador
7:     Recibir respuesta del pensador
8:     if Respuesta es "mayor" then
9:        $fin \leftarrow numero\_propuesto - 1$ 
10:    else if Respuesta es "menor" then
11:       $inicio \leftarrow numero\_propuesto + 1$ 
12:    else
13:      return  $numero\_propuesto$  ▷ Número adivinado
14:    end if
15:  end while
16: end procedure

```

3.1. Notas de implementación

El algoritmo utiliza la búsqueda binaria para reducir eficientemente el rango de posibles números en cada iteración, aprovechando la estrategia dividir y vencer.

3.2. Análisis de complejidad

La complejidad de este algoritmo es $O(\log_2 N)$, donde N es la diferencia entre el mayor y el menor número posible en el rango inicial de posibles números.