

AI-Based Number Guessing Game

- Name-Naveen Kumar
- Branch-CSE(AI&ML)
- Section-B
- Roll no.-202401100400122



INTRODUCTION:

The objective of this game is to create an AI that can guess a randomly chosen number as efficiently as possible .

The AI will given 7 chances in the game and then without choosing randomly it will pick the middle values of the current range and guess the correct number accordingly.

Approach:

- The AI starts by guessing the **middle number** of the range.
- If the guess is **correct**, the game ends.
- If the guess is **too low**, the AI eliminates the lower half and searches in the upper half.
- If the guess is **too high**, the AI eliminates the upper half and searches in the lower half.
- This process continues, reducing the search space by **half in each step**, ensuring an efficient solution.

Code:

```
import random

def ai_guess_number(low, high):
    """AI uses binary search to guess the number."""
    return (low + high) // 2

def number_guessing_game():
    """Main function to run the AI number guessing game."""
    print("Welcome to the AI Number Guessing Game!")
    low, high = 1, 100 # Range of numbers
    number_to_guess = random.randint(low, high) # Random number generation
    attempts = 0 # Counter for attempts

    while True:
        ai_guess = ai_guess_number(low, high) # AI makes a guess
        attempts += 1
        print(f"AI guesses: {ai_guess}")

        if ai_guess == number_to_guess:
            print(f"AI guessed the correct number {ai_guess} in {attempts} attempts!")
            break
        elif ai_guess < number_to_guess:
            print("AI's guess is too low.")
            low = ai_guess + 1 # Adjust lower bound
        else:
            print("AI's guess is too high.")
            high = ai_guess - 1 # Adjust upper bound

if __name__ == "__main__":
    number_guessing_game() # Run the game
```

Output:

```
Welcome to the AI Number Guessing Game!  
AI guesses: 50  
AI's guess is too high.  
AI guesses: 25  
AI's guess is too high.  
AI guesses: 12  
AI's guess is too low.  
AI guesses: 18  
AI's guess is too high.  
AI guesses: 15  
AI's guess is too low.  
AI guesses: 16  
AI guessed the correct number 16 in 6 attempts!
```

References and Credits:

- **GeeksforGeeks:** Binary Search Algorithm
Expectiminimax Algorithm
- **Wikipedia:** [Binary Search](#)