Ping Pong Game 🎮

💡 Project Idea:

The idea is to build a Ping Pong game using Python where the second player (controlled by the computer) uses the **A*** (A-star) algorithm to move intelligently. The computer calculates the optimal movement path to reach the ball and block it, just like a human player would.

🎯 Project Objectives:

- Apply the A* search algorithm in an interactive real-time game.
- Enable the computer to decide movements based on the ball's position.
- Allow adjustable difficulty levels based on the AI's speed and reaction.
- Connect artificial intelligence concepts to a practical and fun game.

🤖 PEAS Description:

Component	Description
P - Performance	Number of balls blocked, goals conceded, and reaction time.
E - Environment	The game arena: ball, paddles, and boundaries.
A - Actuators	Paddle movement (up/down).
S - Sensors	Ball position and paddle position.

🧩 ODESA Framework:

Component	Description
O - Objective	Reach the ball's position and block it.
D - Data	Ball position, speed, and paddle position.
E - Environment	Ping Pong game field.
S - Solution	Use A* algorithm to decide best movement direction.
A - Algorithm	A* (A-star) pathfinding algorithm.

🤖 Agent Type:

The computer paddle is a **Goal-Based Agent**, because it chooses actions based on achieving the goal of intercepting the ball.

🧠 Problem Formulation:

- **States**: All possible vertical positions the paddle can move to.
- **Initial State**: Paddle starts at the center of the screen.
- **Goal State**: Paddle reaches the Y-position of the ball (or close enough).
- **Actions**: Move up or down (e.g., ±5 pixels).
- **Transition Model**: Movement changes paddle's Y-position.
- **Path Cost**: Number of moves or total distance needed to reach the ball.