

# Game Theory Project Specification

## Project Description:

This project involves building a Python-based simulator for strategic decision-making games. Students will create an interactive system that models several well-known Game Theory scenarios, allowing users to explore strategic behaviors, rational thinking, and equilibrium outcomes.

The system will allow users to define players, strategies, and payoffs and will analyze the outcomes using key game theory concepts.

---

## Objectives and Scope:

The project must implement all the Game Theory topics covered in the course, using Python code. Specifically, the system should:

- Support multiple predefined games (e.g., **Prisoner's Dilemma**, **Battle of the Sexes**, **Matching Pennies**, etc.)
  - Represent games in **Extensive Form** and **Normal Form**
  - Analyze **pure and mixed strategies**, and calculate **expected payoffs**
  - Identify **dominated strategies**, compute **best responses**, and determine **rationalizable strategies**
  - Find and display **Nash Equilibria**
  - Present results clearly using tables, diagrams, or charts (text-based or graphical)
-

## Game Examples to Simulate:

- Prisoner's Dilemma
  - Battle of the Sexes
  - Matching Pennies
  - Hawk-Dove Game
  - Custom User-defined Game
- 

## Team Structure:

Each team must consist of **5 students**. Each student will be responsible for a major part of the project:

1. Game Modeling and Extensive Form Representation
  2. Normal Form Conversion and User Input Design
  3. Mixed Strategies and Expected Payoff Calculations
  4. Dominance, Best Response, and Rationalizability Implementation
  5. Nash Equilibrium Computation and Output Visualization
- 

## Expected Deliverables (by each team):

1. Python source code files.
  2. A brief report (2–4 pages, PDF) that explains the games simulated, the results, and how the game theory concepts were applied.
  3. Diagrams of game trees, strategy tables, and results.
- 

## Deadline:

 **Saturday, May 24, 2025**