

Artificial Intelligence

2025/2026

Course Project – Adversarial Search

In **Lab 07 & 08**, you've explored a different branch of search strategies known as **Adversarial Search**, which is particularly relevant in competitive environments like **games**. The lab introduced you to the concept of adversarial search where *agents make decisions in the presence of opponents*, each trying to *maximize their own outcome while minimizing the other's*. You learned about key environment properties that affect such search problems, including *the nature of two-player games, turn-based interactions*. The focus was on the **Minimax algorithm**, a foundational approach in decision-making for adversarial scenarios. Through the **implementation of a Tic-Tac-Toe game**, you applied the Minimax strategy to evaluate possible moves, simulate outcomes, and determine the optimal move at each step. This hands-on exercise helped reinforce the logic behind adversarial decision-making and provided a deeper understanding of how AI can act strategically in competitive environments.

- **You are required to form a team consisting of 4 to 6 Members only with the same Instructor you are attending to.**

(مفیش آی استثناءات ، أصل صحابی فی السکشن الغلانی ... أصل عایز الدحیح معایا)

- **As a team, you are required to Design a game as the instructions below, which contains an AI agent that humans can compete, using one of the Adversarial search strategies.**

• **Project Methodology and Implementation**

The team should choose one of these games (**Connect 4, Chess, Checkers, Othello**), only **Tic-Tac-Toe is not allowed**. If there is any other ideas you would like to implement, just discuss them with the *Instructor* before going to do it.

- Design the environment of the game with interfaces to interact with
- Validate any wrong inputs that comes from the user
- Make the game playable either by Human or computer
- Implement MinMax Algorithm to provide AI competence to the game
- Provide a full clean-code with no redundancy using classes and functions

- **Project Tracking**

In order to guarantee a perfect delivery of the project, and fair grading for all team members. The team should provide **GitHub Repository** which all members participate and contribute to it. The link of the Repo will be provided with the team registration.

- The project should be divided into submodules, each submodule can be made by one student of the team, and it will be as contribution
- Each student should contribute to the project through the Repository on github, the instructor will track every team

- **Project Documentation and Delivery**

The team should organize its work and document every single detail of the project in well-organized and well-detailed documentation. Then, proceeding to the discussion of the project.

- A well-detailed documentation with illustrations showing steps of the implementation of the project
- In the Project Discussion, each student should prove his understanding of all aspects of the project that the team implemented

- **Optional Bonus Tasks**

These tasks are optional for the team, if it is implemented in the correct way, whether **one of them or all**, then all team members will get the points of the bonus task based on the following:

- Implement a well-designed **Graphical-User Interface (GUI)** for the game using Python. Where you can use any framework of your choice
- Use **any other adversarial search** than MinMax Algorithm
- **Optimize** the algorithm as much as possible for faster performance

Good Luck 😊

Dr. Sara Sweidan

AI Course Team