

# Summer of Innovation 2025 - IIT Dharwad

## AdversaRL: Reinforcement Learning & Game Development

HardlyHuman, AI Club  
Code Geass, Coding Club

June 8, 2025

### 1. Overview

This challenge invites participants to design and develop games that integrate Reinforcement Learning (RL) techniques into the gameplay.

### 2. Objectives

- **Create a novel adversarial game:** Design and develop a *unique* game that features an adversarial setup, where players can engage with a Reinforcement Learning (RL) agent. Games that offer innovative mechanics, graphics, and challenges that differentiate them from existing ones always stand out!
- **Enable player interaction with the RL agent:** Implement player controls that allow players to compete against the RL agent. An intriguing concept would be to have multiple RL agents compete against each other.
- **Explore and apply relevant algorithms:** Investigate various Reinforcement Learning algorithms, such as Q-learning, Deep Q-Networks (DQN), or Proximal Policy Optimization (PPO), or consider exploring other approaches, such as evolutionary algorithms. Each has its own strengths and can be adapted to fit the specific needs of your game.

**What is expected:** A fully playable game in which a **Reinforcement Learning agent** (pre-trained/live) competes directly against a **human player**.

### 3. Evaluation Criteria

- **Game Creativity:** Originality of the game, "fun" factor, overall gameplay experience, look and feel.
- **Integration Quality:** How effectively RL is embedded into the game mechanics or design.
- **Technical Execution:** Code quality, documentation, stability, and performance of both the game and RL components.

### 4. Tools & Frameworks

Participants may use any game engines and RL frameworks, such as:

- **Game Engines:** Unity, Unreal Engine, Godot, etc.
- **RL Frameworks:** OpenAI Gym, Stable-Baselines, TensorFlow, PyTorch, etc.

### 5. Submission Guidelines

- Submit the
  - source code (GitHub)
  - a playable game binary/script showcasing the RL agent in-game, and
  - a video demo showcasing the project (training phase timelapse, human - agent interaction, etc.).
- Include documentation describing the game design, RL integration approach, and technical details.
- Clear instructions must be given on how to build and run the game and agent in the video demo/README.
- **IMPORTANT: Prebuilt binaries (or scripts) are also expected in the submission for both Windows and Linux (or browsers, if a browser based game is made).**
- Deadlines and additional submission instructions will be provided separately.

**Content suitability and appropriateness:** While there are no restrictions on the theme of the game, it is crucial to ensure that all content is appropriate for a wide audience. The game should avoid any vulgar, offensive, or inappropriate material. This includes careful consideration of language, imagery, and themes to ensure that the game is respectful and enjoyable for all players.

## 6. Resources

- Introduction to RL (Theory)
  - **An Introduction to Reinforcement Learning**  
<https://www.youtube.com/watch?v=JgvyzIkgxF0>
  - **Q Learning Explained (tutorial)**  
<https://www.youtube.com/watch?v=aCEvtRtNO-M>
  - **Deep Q Learning Explained!**  
<https://www.youtube.com/watch?v=x83WmnbRa2I>
  - **An Introduction to Policy Gradient Methods: Deep Reinforcement Learning**  
<https://www.youtube.com/watch?v=5P7I-xPq8u8>
  - **What are Genetic Algorithms?**  
<https://www.youtube.com/watch?v=XP2sFzp2Rig>
- Getting started with Game Development
  - **How to make a Video Game - Godot Beginner Tutorial**  
<https://www.youtube.com/watch?v=L0hfqjmasi0>
  - **The Unity Tutorial For Complete Beginners**  
<https://www.youtube.com/watch?v=XtQMytORBmM>
  - **The ultimate introduction to Pygame**  
<https://www.youtube.com/watch?v=AY9MnQ4x3zk>
- Implementing RL in Games
  - **Pygame & OpenAI Gym: Snake Game**  
<https://www.youtube.com/watch?v=P0yBab-M6tM>
  - **How to use Machine Learning AI in Unity! (ML-Agents)**  
<https://www.youtube.com/watch?v=zPFU30tbyKs>
  - **Flappy Bird AI (TensorFlow, Keras, Genetic Algorithm) - Artificial Intelligence at UCI**  
<https://www.youtube.com/watch?v=yH21FdWP-0o>

- Other interesting stuff
  - **AI Learns to Walk (deep reinforcement learning)**  
[https://www.youtube.com/watch?v=L\\_4BPjLBF4E](https://www.youtube.com/watch?v=L_4BPjLBF4E)
  - **Genetic Algorithms in Action: BoxCar2D**  
[https://rednuht.org/genetic\\_cars\\_2/](https://rednuht.org/genetic_cars_2/)

## 7. Suggestions

Start small. Do not go through entire playlists of Reinforcement Learning or Game Development before you even begin to program. Learn along the way and pick up concepts as and when you need them.

Keep your game design simple. Reinforcement Learning on a huge state space is practically very hard unless you have a lot of compute ***and time*** to fix suboptimal agents. For example, if making a 2D game, a simple approach could be to keep the state space restricted to a bounded area and the action space restricted to four directions in the game (This is just a suggestion. If you are confident that you can implement better agents in larger state spaces, go for it).

Good luck! Hope you learn some new and interesting concepts :)

- Coding Club, AI Club