ECE 553/653 Neural Networks

# Project 3: Multi-agent reinforcement learning

You will design machine learning algorithms for optimizing the game predators and prey. Your algorithms will be responsible for optimizing the behaviors of predators and prey.

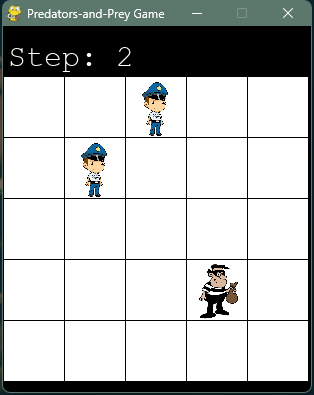
# Requirements

You will write **two** programs: a program that optimizes the behaviors (strategies) of predators, and a program that optimize the behavior of the prey. It is recommended that your programs are written in python.

For the program of the predators (police), you will propose a reinforcement learning (RL) (i.e., Q-learning, deep Q) to optimize the strategy of one predator or multiple predator. You have to design the neural network architecture of predators, and complete the VDN algorithm with reference to the IQL algorithm.

For the program of the prey (thief), you can either optimize the default strategy of the prey or propose a RL method to optimize the strategy of the prey.

# Program Introduction



The Scenario.

The program contains the following parts:

1. main.py, which contains the environment initialization, MARL parameters, Game parameters selction.
2. controller.py, which designed the MARL algorithms, such as IQL, VDN, and QMIX.
3. agent.py, which contains the predators agent design.
4. env.py, which is the Predators and Prey Game board.
5. result\_saved, which saved the training result.

From these parts, you can figure out the following problems:

1. The neural network design in “Brain.build\_model” function in agent.py. We provide basic solution which only contain a signal Dense Layer from state to action. You can try to apply more complex network designs to achieve this function.
2. The parameter selection for training following the guide in main.py. We provide basic training parameters selection in the Readme.md, you can try to apply other adjustable parameters to experiment and obtain a better model.
3. Complete the VDN algorithm in “agent.py”. We provide the IQL and QMIX algorithms as reference, you can try to complete the missing parts in “controller.VDNreplay” function in “agent.py”.
4. Design the prey’ escape strategy in “PredatorsPrey.actor\_prey\_designed” in env.py.

For more code details, please refer to README.md.

# Report

You will write a report for this project. There is no strict format of this report but below are the essential parts:

1. How you design the predator’ neural network architecture.
2. A screen shot of the results in a successful run of your programs.
3. Try different configurations of training and report the results.
4. How you design the prey’ escape strategy.
5. Evaluate the efficiency of your algorithms (e.g. the time used for training, the size of the model).

There are other aspects that you can explore and report, which are up to you. More comprehensive study will receive more credits. Considering multiple predators instead of one predator will also receive more credits.

\* Do not paste a lot of codes in the report. Focus on the design and analysis. Your codes will be submitted separately.

# Submitting your project

You should submit your project in Blackboard. The submission includes: a project report and your source codes. Please put everything in a zip/rar file, name it “FirstName\_LastName\_Project3.zip”.

# Advice

Start early! Do not wait until the last few days.