

# Homework 4 - RL

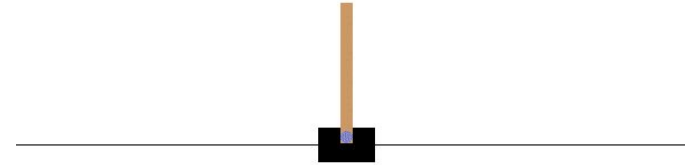
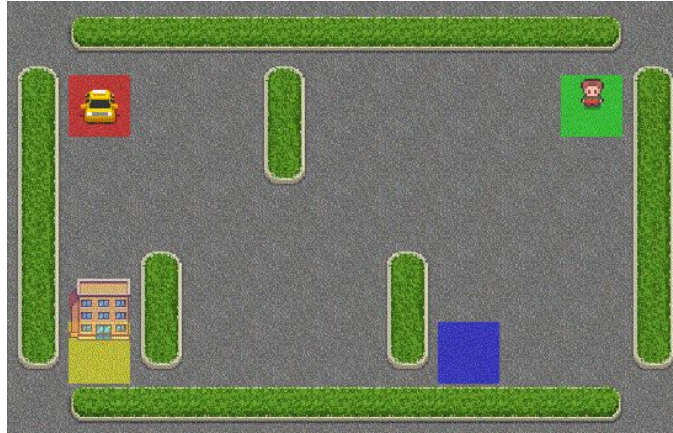
Due Date: 5/13 (Mon) 23:59



# Introduction

In this assignment, you will implement basic RL algorithm, Q-learning and its variants in OpenAI Gym environments i.e.,

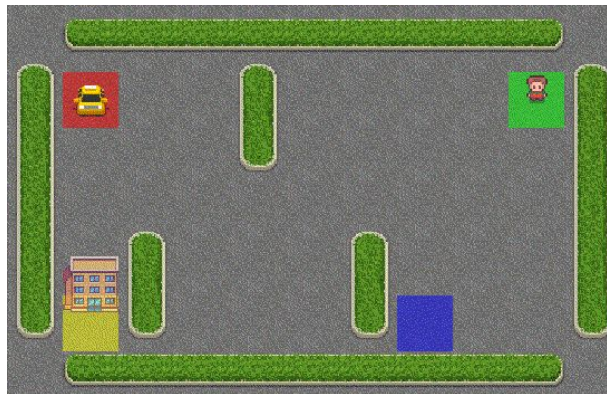
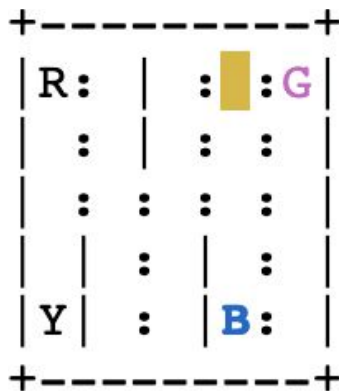
- Taxi-v3
- CartPole-v0



# Introduction

## Taxi-v3

- In Taxi environment, when the episode starts, the taxi(**Yellow block**) drives to the passenger's location(**B**), picks up the passenger, drives to the passenger's destination(**G**), and then drops off the passenger.

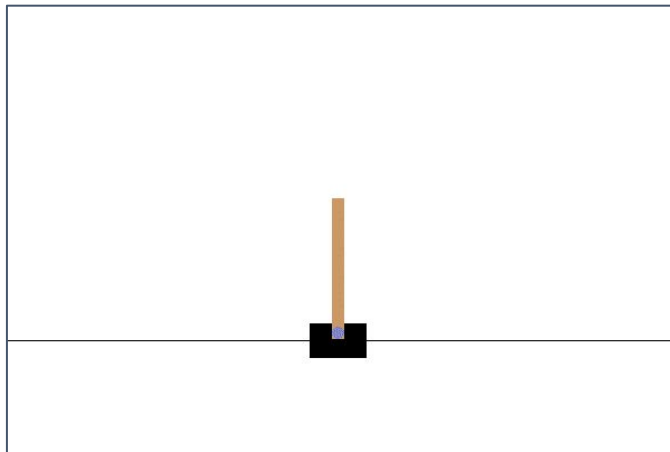


[https://www.gymnasium.dev/environments/toy\\_text/taxi/](https://www.gymnasium.dev/environments/toy_text/taxi/)

# Introduction

## CartPole-v0

- In CartPole environment, the goal is to balance the pole by applying forces in the left and right direction on the cart.



[https://www.gymnasium.dev/environments/classic\\_control/cart\\_pole/](https://www.gymnasium.dev/environments/classic_control/cart_pole/)

# Setup

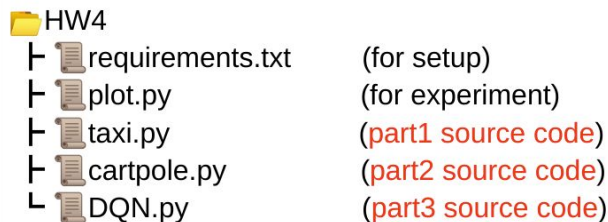
We recommend you to use python **3.7** and all the packages you need are listed in the requirements.txt. Please run the command to install the packages:

```
pip install -r requirements.txt
```

# Implementation (50%)

The sections you need to implement are specified with **# Begin your code** and **# End your code**. Please read all the comments to comprehend the source code before implementation. **Do not modify** the rest of the code.

- Part 1: Q learning in Taxi-v3 (10%)
- Part 2: Q learning in CartPole-v0 (15%)
- Part 3: DQN in CartPole-v0 (25%)



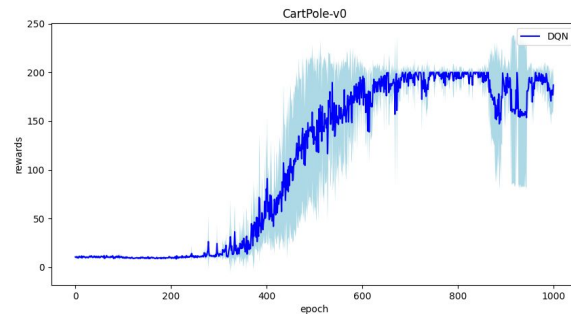
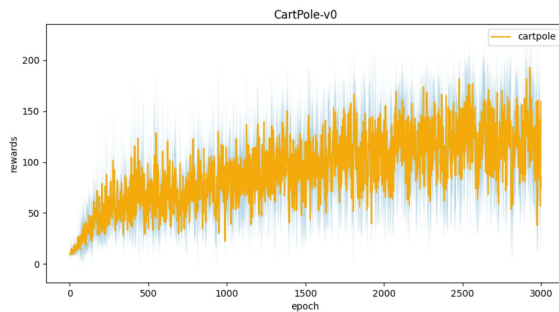
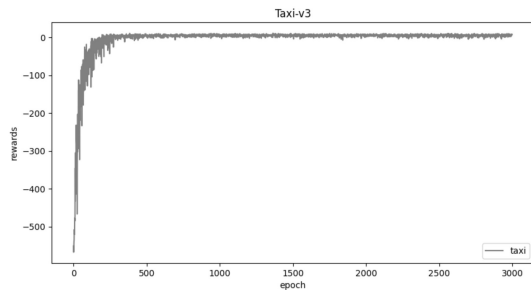
A file explorer view showing the contents of the HW4 directory. The directory contains five files: requirements.txt, plot.py, taxi.py, cartpole.py, and DQN.py. Each file is preceded by a folder icon and a file icon. To the right of each file name is a brief description of its purpose.

HW4	
└ requirements.txt	(for setup)
└ plot.py	(for experiment)
└ taxi.py	(part1 source code)
└ cartpole.py	(part2 source code)
└ DQN.py	(part3 source code)

# Experiment

```
python plot.py [-h] [--taxi] [--cartpole] [--DQN] [--compare]
```

You can use plot.py to plot the learning curves, this will help you verify if you train the model correctly.



# Report (50%)

- You should write your report following the report template
- The report should be written in **English**.
- Please save the report as a **.pdf** file. (font size: 12)
- Answer the questions in the report template **in detail**.

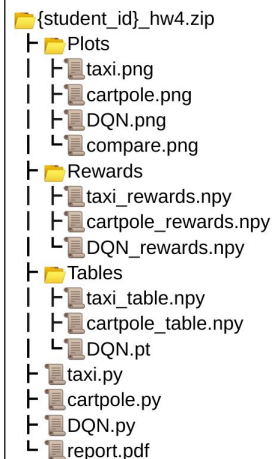


# Submission

**Due Date: 2024/5/13 23:59**

Please compress your [source code](#), [results](#) and [report \(.pdf\)](#) into `STUDENTID_hw4.zip`.

The file structure should look like:



```
{student_id}_hw4.zip
├── Plots
│   ├── taxi.png
│   ├── cartpole.png
│   ├── DQN.png
│   └── compare.png
├── Rewards
│   ├── taxi_rewards.npy
│   ├── cartpole_rewards.npy
│   └── DQN_rewards.npy
├── Tables
│   ├── taxi_table.npy
│   ├── cartpole_table.npy
│   └── DQN.pt
├── taxi.py
├── cartpole.py
├── DQN.py
└── report.pdf
```

**There should NOT be a {student\_id}\_hw4 folder in the zip file.**

**Wrong submission format leads to -10 point.**

**Late Submission Policy**

**20% off per late day**

Please check out the spec  
for more details!