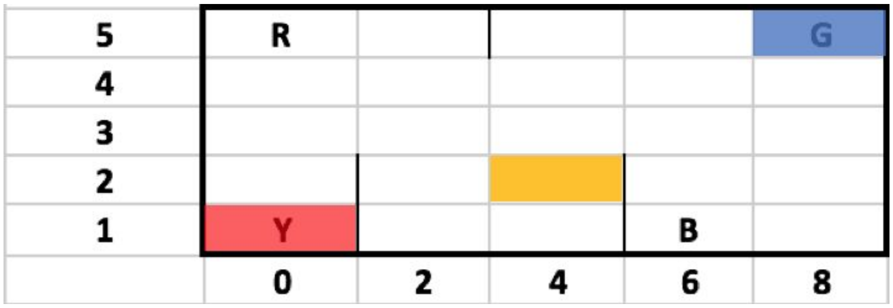
ELEN 6885 Reinforcement Learning

Assignment #01 Part II - Coding

The Taxi game overview

1. Background knowledge & introduction

There are 4 locations (labeled by different letters) and your job is to pick up the passenger at one location and drop him off in another. You receive +20 points for a successful drop-off, and lose 1 point for every timestep it takes. There is also a 10 point penalty for illegal pick-up and drop-off actions.



- blue: passenger
- magenta: destination
- yellow: empty taxi
- green: full taxi
- other letters: locations

2. Task description:

In this assignment, you need to implement following algorithms on the Taxi Game:Value-iteration, Policy iteration, Q-learning and SARSA.

For each implementation, give out the optimal solutions and graphs of all your experiments. You are suggested to include following parts in your report to receive full points:

1. You will implement *value iteration* and *policy iteration* algorithm described in class to solve the MDP problem for Taxi game. Discuss and compare your results in your report.

(1) Value iteration: (coding)
Implement value iteration algorithm in ***vi.py***
[Parameter setting] gamma = 0.95, tolerance for convergence= 10-5
[Your Output] The optimal value function and the optimal policy.
[Your Output] The average rewards for running 100 episodes using the optimal policy.

(2) Policy iteration: (coding)
Implement policy iteration algorithm in ***pi.py***
[Parameter setting] gamma = 0.95, tolerance for convergence= 10-5
[Your Output] The optimal value function and the optimal policy.
[Your Output] The average rewards for running 100 episodes using the optimal policy.

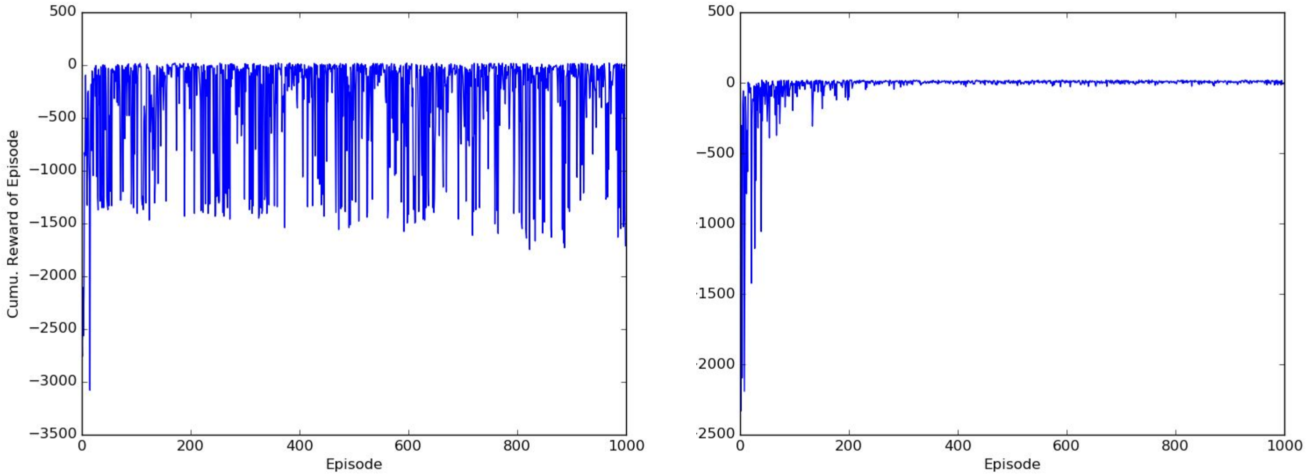
(3) How many iterations do the two algorithms converge? What are the final value function and optimal policy for the two algorithms? Discuss and compare the results of your two algorithms.

2. You will implement the model free methods, *Q-learning* and *Sarsa* algorithms, discussed in the class to solve the reinforcement learning for Taxi game. Discuss and compare your results in your report.

(4) Q-learning: (coding)
Implement Q-learning in ***ql_sarsa.py***
[Your output] The array of Q value for each state-action pair.

(5) SARSA: (coding)
Implement Sarsa in ***ql_sarsa.py***
[Your output] The array of Q value for each state-action pair.

(6) Plot the learning progress of both algorithms for training 1000 episodes. The x-axis is the episode numbers and the y-axis is the average rewards over all episodes. Discuss and compare the result of your algorithms.



For example, the left one can be a performance of Sarsa, and the right one can be a result of Q-learning. (Note: this is just a reference for your report, your results might be different from above ones)

(7) Plot the episodes length for training 1000 episodes of both algorithms. The x-axis is the episode numbers and the y-axis is the number of steps for each episodes. Discuss and compare the result of your algorithms.

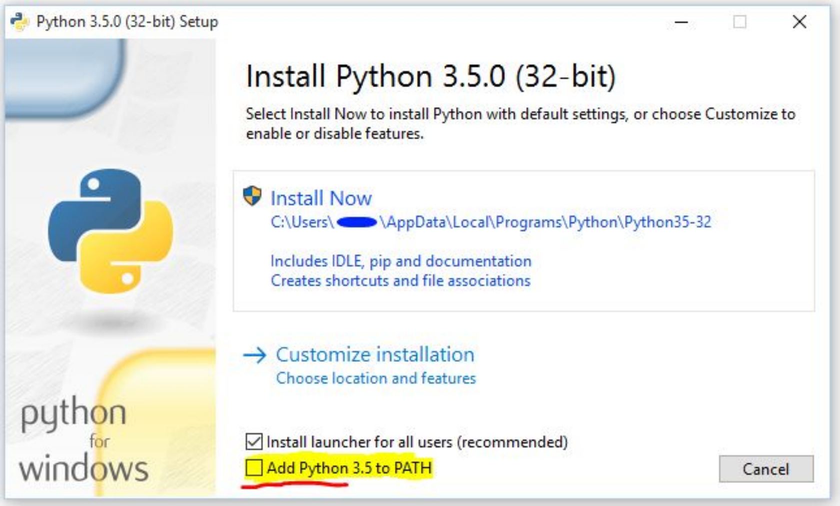
Configure your environment

In this assignment, we will use Python 3.6 to test the code. If you have already successfully set Python environment and installed packages on your laptop, please skip this part.

Windows User

Step 1: Install Python 3.6

Option 1: Download Python 3.6 **here** and install it manually



Python 3.5.0 (32-bit) Setup

Install Python 3.5.0 (32-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

Install Now
C:\Users\[redacted]\AppData\Local\Programs\Python\Python35-32

Includes IDLE, pip and documentation
Creates shortcuts and file associations

→ Customize installation
Choose location and features

☑ Install launcher for all users (recommended)
□ Add Python 3.5 to PATH

Remember to add Python 3.6 into your PATH to enable pip function.

Option 2: Install Anaconda here
Anaconda is a powerful tool widely used in machine learning and data analysis

Step 2: Install Packages

It is recommended to use **virtualenv** as a virtual environment to build your own project. It could reduce the conflict between different packages if you are working on multiple projects. The instruction of using virtualenv can be found here.

Simply run the following command in CMD

```
-pip3 install gym
-pip3 install scipy
```

It will automatically install the packages for you to use.

Mac/Linux User

Step 1: Please follow this guide here to install Python 3 on your Mac

Step 2: Install Packages

Simply run the following command in Terminal

```
-pip3 install gym
-pip3 install scipy
```

Other Options

You are also welcome to use IDEs such as pycharm, a smart platform that integrated with all tools mentioned above

Test the Game

Make sure you finish all steps mentioned above and go to your project folder
You can use the main() function to test your game. Input range from 0 to 5:

```
python3 xxx.py
```

Input:

0 : South (Down)
1 : North (Up)
2 : East (Right)
3 : West (Left)
4: Pick up
5: Drop off

What to submit & How we run your code & Grade

1. Code submission

- A pdf file: **FirstName_LastName_UNI_report.pdf**
- All uncompressed .py files, please using this format:
  FirstName_LastName_UNI_vi.py
  FirstName_LastName_UNI_pi.py
  FirstName_LastName_UNI_ql_sarsa.py

2. How do we run your code

We will use a script to test your code and evaluate the result, so please don't fully copy code and results from internet or your classmates. Your code must follow the output types in the algorithm description.

3. Grading:

Code integrity: 50%
Performance: 20%
Report quality: 30 %