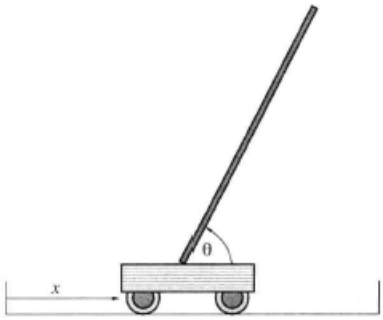
**1. Consider the use case (application) of a Robot driving a car. In this context, what is RL? How can the ADP and TD methods be used for this? What about the Active RL method? [ 30 points]**

Reinforcement learning is a learning mechanism that learns how to map states to actions in order to maximize rewards. The learner is not told which actions to take but must try to find which ones will produce the greatest return. In this case, reinforcement learning is learning how to follow the right path by driving the car to different locations, with different rewards available. When the transition dynamic is known, the ADP method can be used to carry out optimal control. TD refers to learning from the incomplete state sequence obtained by sampling, requiring the robot to estimate an optimal line. Through reasonable bootstrapping, this method first estimates the possible return of a certain state after the state sequence (episode) is complete, and on this basis obtains the value of the state by progressive updating the average, and then continuously updates the value through continuous sampling. Active RL method needs to make decisions on its own, so it is necessary to learn a complete model containing the outcome probability of all actions, and then make choices for its own actions

**2. Based on Ch. 21 from textbook Fig. 21.9** [50 points]

For the problem shown in Fig. 21.9 (balancing a long pole on a moving cart):

a. Construct a Q-Learning representation and explain this as an Active RL problem. Show the details of Policy and Transitions and explain why it is an Active RL rpoblem.




Start State: All observations are assigned a uniform random value in [-0.05..0.05]
Reward：1 for every step taken, including the termination step
Policy：car run to left, car run to right.
Transitions：the position of the car, the speed of the car, the angle of the pole, the speed of the angle.
Episode Termination: Pole Angle is more than 12 degrees, Cart Position is more than 2.4 (center of the cart reaches the edge of the display, solved requirements

Since the car already knows the extreme value of position and the extreme value of the Angle of the bar when it moves, the system contains the possibility of all cases. And every time the car moves, the system needs to judge whether the bar is balanced and choose the next direction of movement, so this is an active RL problem.

**21.1** Implement a passive learning agent in a simple environment, such as the $4 \times 3$ world. For the case of an initially unknown environment model, compare the learning performance of the direct utility estimation, TD, and ADP algorithms. Do the comparison for the optimal policy and for several random policies. For which do the utility estimates converge faster? What happens when the size of the environment is increased? (Try environments with and without obstacles.)

See 21-1.py

4. Implement a Q Learning algorithm similar to this tutorial:
https://www.learndatasci.com/tutorials/reinforcement-q-learning-scratch-python-openai-gym/
but to use the maze problem we learned in class (see Q-Learning Example.docx) and prove your implementation using this data set. [140 points]

See Maze.py