1. (MDP) Suppose we have an MDP with states $\mathcal{S}$ and a discount factor $\gamma < 1$, but we have an MDP solver that can only solve MDPs with discount factor of 1 . How can we leverage the MDP solver to solve the original MDP?

   Let us define a new MDP with states $\mathcal{S}' = \mathcal{S} \cup \{o\}$, where o is a new state. Let's use the same actions ($\mathcal{A}' = \mathcal{A}$), but we need to keep the discount $\gamma' = 1$. Your job is to define new transition probabilities $\mathcal{T}'(s' \mid s, a)$ and rewards $R(s, a, s')$ in terms of the old MDP such that the optimal values $V_{\text{opt}}(s)$ for all $s \in \mathcal{S}$ are equal under the original MDP and the new MDP.

   You are required to give a few transition probabilities and reward functions written in mathematical expressions, followed by a proof to show that the two optimal values are equal.

2. (Q-Learning) Recall that Q-learning converges to optimal policy - even if you're acting suboptimally. The task in this exercise is to prove this statement step by step. Let us first cover some preliminaries. Denote by (X, d) the metric space, where X is a space and the metric $d : X \times X \to \mathbb{R}$ is defined on pairs of elements in X. For example, let $X = \mathbb{R}^n$ and $d(x_1, x_2) = \|x_1 - x_2\|_2$ for any $x_1, x_2 \in X$, where $\|\cdot\|_2$ denotes the 2-norm. Then $d(x_1, x_2)$ denotes the Euclidean distance between x_1 and x_2. Let $H : X \to X$ be a mapping between the space X and itself. If there exists some $0 < L < 1$, s.t. $\forall x, y \in X, d(H(x), H(y)) \leq L \cdot d(x, y)$, then the mapping H is defined as a contradiction mapping. If there exists a point $x \in X$ s.t. $H(x) = x$, then x is defined as the fixed-point of mapping H. If H is a contradiction mapping, then it could be shown that H admits a unique fixed-point x^* in X.

   (a) Recall that the optimal Q^* satisfies the optimal Bellman equation

   $$Q^*(s, a) = \sum_{s'} T(s, a, s') \left[R(s, a, s') + \gamma \max_{a'} Q^*(s', a') \right]$$

   If we treat X as the space of Q functions (i.e., each element q in X is a Q function, which further specifies a Q value $q(s, a)$ for a given state-action pair (s, a)), and treat H as

   $$H : q(s, a) \mapsto \sum_{s'} T(s, a, s') \left[R(s, a, s') + \gamma \max_{a'} q(s', a') \right]$$

   Then it is clear that Q^* is the fixed-point of H since $Q^* = H(Q^*)$. Assume the discount factor $0 < \gamma < 1$. Prove that H is a contradiction mapping with respect to the metric

   $$d(q_1, q_2) = \|q_1 - q_2\|_\infty = \max_{s,a} |q_1(s, a) - q_2(s, a)|$$

   for any two Q functions q_1 and q_2, where $\|\cdot\|_\infty$ is the maximum norm

   (b) Consider a finite MDP$(\mathcal{S}, \mathcal{A}, T, R, \gamma)$ with finite state space $\mathcal{S}$ and finite action space $\mathcal{A}$. Assume the reward function R is bounded and deterministic and $0 < \gamma < 1$. Recall that Q-learning updates Q function as

   $$\begin{aligned} Q_{t+1}(s_t, a_t) &= Q_t(s_t, a_t) - \alpha_t \left(Q_t(s_t, a_t) - \text{sample}_t \right) \\ &= (1 - \alpha_t) Q_t(s_t, a_t) + \alpha_t \cdot \text{sample}_t \end{aligned}$$

   where $\text{sample}_t = R(s_t, a_t, s') + \gamma \max_{a'} Q_t(s', a')$, $0 \leq \alpha_t \leq 1$ is the learning rate at time t, and $\{s_t\}$ is the sequence of states obtained following policy π, which satisfies $\mathbb{P}_\pi [A_t = a \mid S_t = s] > 0$ for all state-action pairs (s, a). Further assume that $Q^*(s, a)$ is bounded for all the state action pair (s, a), and $Q_t(s, a)$ is bounded for all the state-action pair $(s, a), \forall t > 0$. Prove that Q-learning converges to Q^* if $\sum_t \alpha_t = \infty$ and $\sum_t \alpha_t^2 < \infty$.

   Hint: First construct a sequence $\Delta_{t+1}(s, a) = Q_t(s, a) - Q^*(s, a)$ using the update rule of Q - learning and Q^*. Then verify that the three assumptions in Lemma 1 hold. Finally apply Lemma 1 to finish this exercise.

Lemma 1. The random process $\{\Delta_t\}$ *taking values in* $\mathbb{R}$ *and defined as*

$$\Delta_{t+1}(x) = (1 - \alpha_t)\,\Delta_t(x) + \alpha_t F_t(x)$$

converges to 0 under the following assumptions:
(1) $0 \leq \alpha_t \leq 1, \sum_t \alpha_t = \infty$ *and* $\sum_t \alpha_t^2 < \infty$*;*
(2) $\|\mathbb{E}[F_t \mid \mathcal{F}_t]\|_\infty \leq \gamma \|\Delta_t\|_\infty$ *with* $\gamma < 1$*, where* $\mathcal{F}_t = \{\Delta_t, \Delta_{t-1}, \ldots, \Delta_1, F_{t-1}, \ldots, F_1\}$ *stands for the past information at time* t*, and* $\mathbb{E}[F_t \mid \mathcal{F}_t]$ *denotes the conditional expectation of* F_t *given* $\mathcal{F}_t$*;*
(3) $\mathbb{V}[F_t(x) \mid \mathcal{F}_t] \leq C\,(1 + \|\Delta_t\|_\infty)^2$ *for some constant* $C > 0$*, where* $\mathbb{V}[F_t(x) \mid \mathcal{F}_t]$ *denotes the conditional variance of* $F_t(x)$ *given* $\mathcal{F}_t$*.*

For the following coding exercise, please refer to Lab4.pdf and provided codes.

3. (Adversarial Search) In this problem, you will design agents for the classic version of Pacman.
   (a) Implement **Minimax Search** algorithm for the player.
   (b) Implement **Alpha-Beta Pruning** algorithm for the player.
   (c) Implement **Expectimax Search** algorithm for the player.
   (d) Implement **Minimax Search** algorithm for the ghost.
4. (MDP) In this problem, you will implement classical algorithms and use them to solve Peeking Blackjack game.
   (a) Implement algorithms of **Value Iteration** and **Policy Iteration**.
   (b) Construct an MDP that can represent the Peeking Blackjack game and use above algorithms to solve it.

Requirements for your report:

- The report should include an explanation of each component of your algorithm implementation .
- The report should include **all** the results of the commands we provide. You are also encouraged to try other commands and present your findings.
- The report should include some simple comparison and analysis of the algorithms and results.

Name your **zip** file containing the above five as 'xxxxxxxxxxxx.zip' with your student ID, and submit it on Canvas.