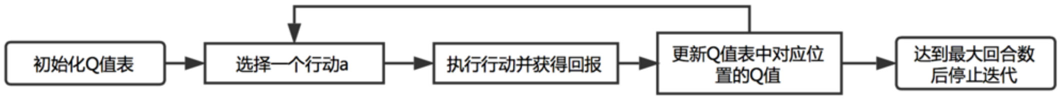
Q-Learning算法练习题（初级-中级）

题目1：Q-Learning 算法流程与更新公式

题干： Q-Learning 是一种基于价值的强化学习算法，其核心在于通过学习状态-动作值函数（Q值）来评估每个状态下各动作的长期收益。算法每一步按照以下流程进行：① 在当前状态下选择一个动作（通常采用 ϵ-贪婪策略）；② 执行动作获得即时奖励 r 并转移到下一个状态；③ 利用奖励和下一状态的最大预期Q值来更新当前状态-动作的Q值。整个过程不断迭代，直至Q值收敛或达到指定迭代次数。



问题设定： 请写出 Q-Learning 的状态-动作值更新公式，并解释其中各符号的含义（包括学习率 α 和折扣因子 γ）。此外，假设某次更新中当前状态s执行动作a得到奖励 $r = 1$，下一状态的最大Q值为 $\max_{a'} Q(s', a') = 2$，学习率 $\alpha = 0.5$，折扣因子 $\gamma = 0.9$，且当前 $Q(s, a) = 0$。请根据公式计算更新后 $Q(s, a)$ 的数值。

答案与解析： Q-Learning 的更新公式为：

$$Q(s,a) \leftarrow Q(s,a) + \alpha \big[r + \gamma \max_{a'} Q(s',a') - Q(s,a) \big] \tag{1}$$

其中，r 是当前执行动作获得的即时奖励，$\max_{a'} Q(s', a')$ 是下一状态 s' 所能得到的最大预期累积奖励，α 为学习率（控制新获信息对原有Q值影响的程度），γ 为折扣因子（权衡未来奖励的重要性）。根据公式(1)，利用新的奖励信号（现实价值）修正当前Q估计值：**旧值** $Q(s, a)$ 加上 **学习率** α 乘以 **TD误差** $\delta = [r + \gamma \max_{a'} Q(s', a') - Q(s, a)]$ 实现更新。

代入数值计算：$\delta = 1 + 0.9 \times 2 - 0 = 1 + 1.8 = 2.8$，再乘以学习率0.5，得到增量 $\alpha\delta = 0.5 \times 2.8 = 1.4$。因此更新后 $Q(s, a) = 0 + 1.4 = 1.4$。这个结果表示在当前状态下执行动作a的价值提高到了1.4。经过多次迭代，此更新公式能逐步修正Q表，使其逼近最优的状态-动作价值。公式中的α值决定更新幅度大小，γ值决定了对未来奖励的重视程度：γ 越大，更新时越考虑长期回报，γ 越小则更多考虑眼前奖励。

题目2：ϵ-贪婪策略的动作选择

题干：*在Q-Learning中，为了平衡*探索(exploration)和**利用**(exploitation)，常采用ϵ-贪婪策略选择动作。该策略由探索率ϵ控制：以概率ϵ随机选择一个动作（探索），以概率$1-\epsilon$选择当前Q值最大的动作（利用）。在训练初期通常设定较大的ϵ（例如1或0.5）以鼓励探索，随后随着训练进行逐渐减小ϵ值，以提高选择最优动作的概率。

问题设定： 假设智能体在某一状态下有3个可选动作，其对应的Q值为 $Q(s, a_1) = 5$、$Q(s, a_2) = 2$、$Q(s, a_3) = 1$。若采用ϵ-贪婪策略且当前 $\epsilon = 0.1$，请问智能体选择每个动作的概率各为多少？并简要说明ϵ值大小对策略的影响。

答案与解析： 根据ϵ-贪婪策略，智能体有10%概率随机探索动作，有90%概率选择最优动作。在该状态下动作a_1的Q值最高（5），因此"最优动作"是a_1。选择a_1的总体概率约为 $1 - \epsilon + \frac{\epsilon}{3} = 0.9 + 0.1/3 \approx 0.9333$（即93.33%）。其余两个动作作为非最优动作，仅可能在"探索"时被选中，各自的概率约为 $\epsilon/3 \approx 0.0333$（约3.33%）。

由此可见，当$\epsilon = 0.1$较小时，智能体大部分时间会利用当前学得的最佳动作（约93%的情况选择a_1），只有少部分时间随机尝试其他动作，从而在**利用**最优策略的同时保持少量**探索**以防止陷入次优解。如果ϵ值增大（例如接近1），智能体选择随机动作的比例会提高，探索行为加强，但可能造成短期内累积奖励降低。反之，ϵ值过小（接近0）则几乎总是选择当前最优动作，利用程度高但可能错过更优策略。训练初期通常选取较大的ϵ以充分探索，当Q值逐步趋于稳定后再降低ϵ让智能体更多利用已学策略。

题目3：Q值收敛性与最优策略

题干：*经过多次迭代更新，Q-Learning算法的Q表会逐渐*收敛到稳定值。此时各状态-动作对的Q值不再发生明显变化，意味着智能体对环境的策略评估趋于稳定。在有限状态和动作的环境下，只要每个状态-动作对被充分探索且学习率逐步衰减，Q-Learning 的Q值理论上将收敛到最优状态-动作值，这对应着环境的最优策略已被找到。

问题设定： 请解释什么是Q值的"收敛"，以及Q值收敛后如何提取最优策略。为什么Q-Learning算法在充分探索下能够保证找到最优策略？请结合算法特点给出说明。

答案与解析：*Q值收敛是指随着训练进行，Q表中的数值逐渐稳定在接近真实的状态-动作价值，并且进一步训练不再导致显著变化。此时可以认为智能体对每个状态的行动价值评估已成熟。指出，当Q值"逐渐收敛到一个相对稳定的值"后，智能体在每个状态下选择Q值最大的动作就构成了*最优策略。 也就是说，Q表收敛后，在状态s选择 $\arg\max_a Q(s, a)$ 即可确保行动策略的长期回报最大。

Q-Learning 能够收敛并找到最优策略，源于其理论保障和更新机制。表明对于任意有限的马尔可夫决策过程（MDP），只要采用部分随机的策略进行足够次数的探索，Q-Learning 最终会收敛到可以使累积奖励期望值最大的策略。这一保证依赖以下因素：1）**充分探索**：智能体在训练过程中不断以ϵ概率尝试各动作，确保不会一直局限于某条路径，从而"遍历"到所有重要状态-动作对；2）**时间差分更新**：算法通过公式(1)不断修正对真实回报的估计，奖励信号会沿状态序列向前传播，使得与目标相关的状态价值逐步提升，最终逼近真实值；3）**学习率衰减**：理论上要求学习率α在无限次迭代下满足$\sum \alpha = \infty, ; \sum \alpha^2 < \infty$等条件，以保证无穷多次更新后Q值误差收敛于0。综合以上，随着训练迭代，Q表会收敛到最优Q值，此时由Q表即可提取出相应的最优策略。

题目4：学习率 α 和折扣因子 γ 对学习过程的影响

题干： Q-Learning算法中包含两个重要的超参数：学习率α和折扣因子γ。**学习率** α（0 ~ 1之间）决定每次更新中新获得的信息在多大程度上替代旧的Q值；**折扣因子** γ（0 ~ 1之间）用于权衡未来奖励的价值。当α取极端值时（如0或1），更新特性会发生明显变化；同样不同的γ取值会导致智能体更关注近期奖励或长期回报。

问题设定： 请分析以下情况对学习过程的影响：

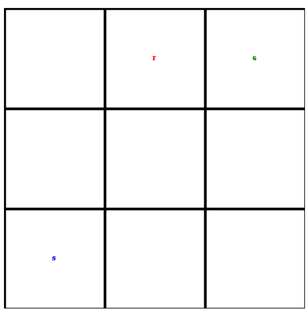
1. 学习率 $\alpha = 0$ 与 $\alpha = 1$ 分别意味着什么？α 取较大值或较小值会对Q值收敛速度和稳定性产生怎样的影响？
2. 折扣因子 γ 接近0 与 接近1 分别意味着什么？不同γ取值会如何影响智能体在路径规划时对短期奖励和长期回报的权衡？

答案与解析： (1) **学习率 α 的影响：** $\alpha = 0$ 意味着完全不学习——每次更新时新信息的权重为0，Q值将保持不变，智能体无法从新的经验中获益。$\alpha = 1$ 则表示完全依赖最新的信息，每次直接用新计算的值替换旧Q值。在这种极端情况下，学习速度最快，但容易受单一步骤的随机波动影响。一般地，较大的α会使Q值**更新幅度变大**、收敛**速度加快**，但过大会导致Q值在真值附近来回震荡，收敛不稳定；较小的α则每次只做少量更新，学习**步伐变小**，收敛速度放慢，但更新曲线更加平滑稳定。实际应用中常选取适中的α（如0.1~0.5），或在训练过程中逐步降低α，以兼顾初期的快速学习和后期的收敛稳定。

(2) **折扣因子 γ 的影响：** γ 决定智能体有多看重**未来的回报**。γ 接近0时，智能体几乎**只关注眼前的即时奖励**，不考虑长期后果，策略倾向于获取短期利益。例如$\gamma = 0$时，更新公式中只保留当前奖励r（因为未来项$\gamma \max Q(s', a')$被置0），智能体会贪图一步的得失。相反，γ 接近1时，智能体会**高度重视长期累积奖励**，为了最大化远期回报可以在眼前做出一定牺牲。$\gamma = 1$（在有限回合情形下）意味着认为未来奖励与当前同等重要，此时智能体将尽可能规划长远的最大奖励。需要注意，如果环境没有终止情景而$\gamma = 1$，则可能由于无限折扣和循环路径导致算法不收敛，通常在无终止的无限任务中γ会设小于1。综合而言，γ 大小影响策略的选取：较小γ使策略偏重**短视**，较大γ使策略偏向**长远规划**。实际设置中，一般根据任务的最长决策跨度选择γ，只要γ在算法可收敛范围内通常取值尽可能大以强调长期回报。（例如经验公式 $T \approx \frac{1}{1-\gamma}$ 可粗略估计智能体考虑的步数跨度。）

题目5：小型网格世界中的路径规划与奖励机制设计

题干： 上述3x3网格世界示意图：S为起点，G为目标状态（到达后获得 +5 奖励），T为陷阱（进入则 -10 分惩罚），其余为空白格子奖励为0。智能体可在网格中上下左右移动一步，试图从起点S出发到达目标G。每个回合以到 达目标或者跌入陷阱结束。初始时刻所有状态-动作的Q值均为0，智能体将通过不断与环境交互来学习各状态下的最优动作。设计这样的奖励机制是为了鼓励智能体尽快抵达目标、避免陷阱。



问题设定： 请基于上述网格环境：

1. 写出该网格世界的奖励设置方案，并解释为何这样设计奖励有助于路径规划（提示：正奖励引导目标，负奖励惩罚陷阱）。
2. 随着Q-Learning训练进行，智能体将逐渐找到从S到G的最优路径。请指出该最优路径（可用格子序号或方向序列表示），并结合Q值更新过程解释智能体为何会选择该路径而避开陷阱。

正确答案与解析：

1. **奖励机制设计：*目标格G给予正的高额奖励（例如 +5），表示达到目标的收益；陷阱格T给予较大的负奖励（例如 -10），表示掉入陷阱的严重惩罚；其余普通格子奖励为0，表示中间状态本身无特殊奖励。这样的设计使智能体在尝试过程中逐渐意识到"到达G有显著收益"以及"陷入T会导致损失"，从而在策略上倾向于朝目标前进并远离陷阱。正奖励提供了明确的导航方向，负奖励确保了智能体在探索时学会*避开高风险区域。**

2. **最优路径规划：**在该3x3网格中，智能体从S出发可选择的路径有多条。由于陷阱T在 (第一行第二列) 位置，最优路径应避开此位置。例如，一条绕开陷阱的最优路径为：从S先向右移动两次，再向上移动两次直达G（用坐标表示路径序列：S=(2,0) → (2,1) → (2,2) → (1,2) → G=(0,2)）。该路径成功绕过了陷阱T。

   为何Q-Learning选择该路径：*在训练初期，智能体对环境一无所知，可能随机走入陷阱T而得到-10奖励，这会通过Q值更新将陷阱附近状态的价值大幅调低，促使后续选择避开这些状态。描述了Q值传播的过程：一开始智能体常因靠近起点的陷阱而受罚，但*一旦智能体偶然到达了目标G，情况就发生变化——到达G获得的正奖励会沿着该回合经历的路径逐步向前传播。具体来说，当智能体首次到达G（+5），它上一状态的Q值将更新增大；随着训练反复，这种正向的价值信号逐步传播回起点S。于是，各状态对**通向目标的动作**的Q值不断累积正收益，而对通向陷阱的动作则保持负值。最终，起点S及其周围状态朝向目标G方向的Q值显著高于朝向陷阱T方向的Q值，智能体由此学会选择通往G的路线。

   随着训练迭代次数增加，Q表逐渐收敛：通往G路径上的状态-动作对Q值趋近于某个正值，而进入T方向的Q值趋近于较大的负值。收敛后，智能体在每个状态都会选择Q值最高的动作，因此自然避开负值高的陷阱方向，**选择累积奖励期望最大的路径**到达目标。这个过程体现了奖励机制对路径规划的引导作用：正奖励像"指引灯"，使得与目标相连的路径被逐步强化；负奖励如"警戒线"，让智能体对高风险区域敬而远之。经过充分学习后，智能体成功规划出从S到G的最优路线，并对环境的策略有了基本理解。

高级 Q-Learning 综合练习题

下面设计了 5 道偏难的 Q-Learning 强化学习综合题目，适用于已掌握基本理论的学习者进行深入练习。每题包含详细的场景描述、明确的问题设定，以及所要求解决的问题。每题后附有清晰的解析与正确答案。

问题1：简易网格世界中的 Q 值更新计算

情景描述： 一个 2×3 的网格世界，智能体从起点 S 移动到目标 G。网格如下图所示（S 为起点，G 为目标，`.` 表示空白格）：

```
.    .    G(奖励+10)
S    .    .
```

起点 S 在左下角，目标 G 在右上角。智能体可采取上下左右动作移动单格，移动越界视为不合法。每步移动都有行动成本，即奖励 -1（表示时间/能量消耗），到达目标时获得奖励 $+10$，任务结束。设折扣因子 $\gamma = 0.9$，学习率 $\alpha = 0.5$，初始 Q 表格中所有状态-动作对的 Q 值均为 0。

经历轨迹： 智能体从起点 S 出发，按照 ϵ-贪心策略（ϵ 较大保证探索）选择动作，经历了如下一个情节（episode）：

1. 在状态 $S(1,0)$ 采取动作 **上**，到达状态 $(0,0)$，获得奖励 -1。
2. 在状态 $(0,0)$ 采取动作 **右**，到达状态 $(0,1)$，获得奖励 -1。
3. 在状态 $(0,1)$ 采取动作 **右**，到达目标状态 $G(0,2)$，获得奖励 $+10$，回合结束。

请依据 Q-Learning 更新公式逐步计算上述轨迹中各步执行后的 Q 值更新，并回答以下问题：

1. 给出每一步更新后相关状态-动作对的 Q 值。特别地，求 $Q(S, \text{上})$、$Q((0,0), \text{右})$ 和 $Q((0,1), \text{右})$ 在执行完上述步骤后的新值。
2. 在完成上述更新后，若智能体采用 ϵ-贪心策略选择下一轮在起点 S 的动作（设 $\epsilon = 0.1$），智能体会选择哪个动作？请说明依据。

解析：

根据 Q-Learning 的更新规则，在执行动作后按公式

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma maxa\prime Q(s\prime, a\prime) - Q(s,a)]Q(s,a) \leftarrow Q(s,a) + \alpha\Big[r + \gamma \max_{a'} Q(s', a') - Q(s,a)\Big]$$

进行更新。初始时所有 $Q = 0$。逐步计算每一步后的 Q 值：

- **第1步：** 从状态 S 执行动作"上"到达状态 $(0,0)$，奖励 $r = -1$。更新：
  Q(S,上)=0+0.5[−1+0.9maxa'Q((0,0),a')−0].Q(S,\text{上}) = 0 + 0.5 \big[-1 + 0.9 \max*{a'}Q((0,0),a') - 0\big].*
  *到达状态 $(0,0)$ 时其各动作的 Q 值仍为0（尚未更新），故 $\max*{a'}Q((0,0),a')=0$。因此：
  Q(S,上)=0.5×(−1+0)=−0.5.Q(S,\text{上}) = 0.5 \times (-1 + 0) = -0.5.
- **第2步：** 从状态 $(0,0)$ 执行动作"右"到达状态 $(0,1)$，奖励 $r = -1$。更新：
  Q((0,0),右)=0+0.5[−1+0.9maxa'Q((0,1),a')−0].Q((0,0),\text{右}) = 0 + 0.5 \big[-1 + 0.9 \max*{a'}Q((0,1),a') - 0\big].*
  *此时状态 $(0,1)$ 尚未有更新过的 Q 值，$\max*{a'}Q((0,1),a')=0$，所以：
  Q((0,0),右)=0.5×(−1+0)=−0.5.Q((0,0),\text{右}) = 0.5 \times (-1 + 0) = -0.5.

- **第3步：** 从状态 $(0,1)$ 执行动作"右"到达目标状态 G，奖励 $r=+10$。目标为终止状态，可取 $\max_{a'} Q(G,a')=0$（终止状态后续回报视为0）。更新：
  Q((0,1),右)=0+0.5[10+0.9×0−0]=0.5×10=5.0.Q((0,1),\text{右}) = 0 + 0.5 \big[10 + 0.9 \times 0 - 0\big] = 0.5 \times 10 = 5.0.

经过上述轨迹，Q 表中发生变化的值如下：

- $Q(S,\text{上})=-0.5$（其余如 $Q(S,\text{右})$ 仍为0，因为起点向右的动作未在此轨迹中执行）。
- $Q((0,0),\text{右})=-0.5$。
- $Q((0,1),\text{右})=5.0$。

其它未经历的状态-动作对仍保持初始值 0。

第2问： 更新后在起点 S 的 Q 值：目前只有 $Q(S,\text{上})=-0.5$，而 $Q(S,\text{右})$ 等其他可能动作仍为 0。采用 ϵ-贪心策略（$\epsilon=0.1$）时，智能体以 $1-\epsilon=90$ 的概率选择当前估计最优动作，以 10 概率探索随机动作。由于 $0>-0.5$，估计最优动作是在 S 向 **右**（即使先前未探索过，该动作的 Q 估值为0，高于向上的 -0.5）。因此，智能体下次在 S **很可能选择"右"方向**（90% 概率），只有10%概率探索其他动作。这展示了 ϵ-贪心策略：即使"右"尚未尝试过，其当前 Q 值最大，故作为贪心动作被选，大概率被执行。

问题2：非确定性转移环境下的 Q 值更新

情景描述： 一个单状态 S 的情景中，智能体有两个可选动作，环境转移具有随机性：

- **动作 A（冒险型）**：以 80% 概率达到目标状态 G，获得奖励 $+10$；以 20% 概率掉入陷阱状态 P，获得奖励 -10。到达 G 或 P 后回合结束。
- **动作 B（保守型）**：确定性地到达一个安全终点 D，获得奖励 $+3$，回合结束。

假设折扣因子 $\gamma=1.0$（因到达终点即结束，没有后续回报可折扣），学习率 $\alpha=0.5$，初始时 $Q(S,A)=Q(S,B)=0$。智能体将通过多次试验来学习哪种动作更优。

请回答下列问题：

1. **第一次试验：** 智能体在 S 选择动作 A。设此次结果为以概率 0.8 实现了正向结果（到达 G 奖励 +10）。请根据 Q-learning 更新规则计算，此次更新后 $Q(S,A)$ 的值。【提示：终点状态后续 $\max Q=0$】
2. **第二次试验：** 智能体再次在 S 执行动作 A。这次不幸以 20% 概率掉入陷阱 P（奖励 -10）。请计算此次更新后 $Q(S,A)$ 的值（此时应基于上一步更新后的 $Q(S,A)$）。
3. 根据上述结果，分析动作 A 的 Q 值是如何随着正负不同结果而更新的。理论上，动作 A 在充分试验下的期望价值应接近多少？动作 B 的价值又是多少？基于期望回报，哪个动作最终会是最优策略？

解析：

1. **第一次试验（正向结果）**：初始时 $Q(S,A)=0$。执行动作 A 得到奖励 $r=+10$，到达终点状态 G。依据更新公式：
   Q(S,A)←0+0.5[10+1.0×0−0]=0.5×10=5.0.Q(S,A) \leftarrow 0 + 0.5\big[\,10 + 1.0 \times 0 - 0\big] = 0.5 \times 10 = 5.0.
   更新后有 $Q(S,A)=5.0$。动作 B 此时尚未执行过，仍有 $Q(S,B)=0$。
2. **第二次试验（负向结果）**：此时 $Q(S,A)=5.0$（由上次更新所得）。再次执行 A，这次奖励 $r=-10$（掉入陷阱），更新：
   Q(S,A)←5.0+0.5[(−10)+1.0×0−5.0].Q(S,A) \leftarrow 5.0 + 0.5\big[\,(-10) + 1.0 \times 0 - 5.0\big].
   计算括号内的TD误差：$(-10-5.0)=-15.0$。因此：
   Q(S,A)=5.0+0.5×(−15.0)=5.0−7.5=−2.5.Q(S,A) = 5.0 + 0.5 \times (-15.0) = 5.0 - 7.5 = -2.5.
   可以看到，负奖励使 $Q(S,A)$ 从正值下降到了负值。
3. **价值分析：** 动作 A 的 Q 值在反复试验中会根据遇到正/负结果不断调整：正反馈使估计增大，负反馈使估计减小。理论上，若不断尝试，$Q(S,A)$ 将收敛于动作 A 的**期望回报**。动作 A 每次有 80% 概率获得 +10，20% 概率获得 -10，其期望奖励为：
   E[R∣A]=0.8×10+0.2×(−10)=8−2=6.E[R|A] = 0.8 \times 10 + 0.2 \times (-10) = 8 - 2 = 6.
   因为到达终点无后续回报且 $\gamma=1$，最优 Q 值应为该期望值，即 $Q^*(S,A)\approx 6$。动作 B 每次固定得到 +3，没有不确定性，期望回报即 3。因此 $Q^*(S,B)=3$。比较两者可知，动作 A 的长期平均收益更高（约为 6），尽管偶尔会有损失。**最优策略**最终会选择动作 A，因为其期望回报大于动作 B。【当然，为了准确估计 $Q(S,A)$ 接近 6，智能体需要足够多的探索尝试，以经历足够次数的正反两种结果。随着试验次数增加，$Q(S,A)$ 会在 6 附近波动并逐渐收敛于该值。】

提示： 以上过程说明了在非确定性环境中，Q-Learning 通过多次采样更新估计期望回报的机制。如果智能体仅凭前两次尝试就贪心地放弃了动作 A（例如第2次后 $Q(S,A)$ 变为 -2.5，低于 $Q(S,B)=0$），它可能错过实际更优的动作。因此需要 ϵ-贪心等策略持续探索，以避免过早收敛到次优策略。

问题3：折扣因子对多目标导航的影响

情景描述： 一个 3×3 的网格世界，包含两个目标点，智能体需要导航选择路径。环境如下图：

```
.   .   G2(奖励+10)
.   .   .
S   .   G1(奖励+5)
```

- 起点 S 在左下角；目标1 ($G1$) 在右下角，奖励 +5；目标2 ($G2$) 在右上角，奖励 +10。
- 智能体每步移动的代价为奖励 -1（鼓励更短路径）；到达任一目标状态即任务结束并获得相应正奖励。
- 假设智能体总是选择最短路径前往任一目标（无障碍物干扰时，最短路径等步数的路径收益相同）。

显然，$G2$ 奖励较高但距离更远；$G1$ 奖励较低但距离较近。折扣因子 γ 将影响智能体对远期回报的重视程度。我们以 $\gamma = 0.9$ 和 $\gamma = 0.5$ 为例进行分析：

1. 在折扣因子 $\gamma = 0.9$ 情况下，计算智能体从 S 分别前往 $G1$ 和 $G2$ 的最优路径的折扣累计回报（即总回报）。哪一目标对应的回报更高？由此推断智能体倾向选择哪个目标。
2. 将折扣因子降为 $\gamma = 0.5$，重复上述计算并比较两条路径回报高低。此时智能体更可能选择哪个目标？
3. 综合分析折扣因子对策略的影响：为什么较大的 γ 值会使智能体更偏好远大回报（远处目标），而较小的 γ 值会让智能体倾向于就近获得较小回报？

解析：

首先确定从 S 出发到各目标的最短路径长度：

- 到目标 $G1$（右下角）的最短路径需要向右移动 2 步（共 2 步）。
- 到目标 $G2$（右上角）的最短路径需要向上移动 2 步再向右移动 2 步（共 4 步）。

在每步奖励 -1、最终目标奖励按所示的情况下，我们计算折扣累计回报：
记路径上的即时奖励序列为 $r_0, r_1, \dots, r_{n-1}, r_n$，其中前$n$步为移动奖励 -1，最后一步n达到目标奖励为正值。折扣累计回报为 $R = \sum_{t=0}^{n} \gamma^t r_t$。

- **$\gamma = 0.9$ 时：**
 到 $G1$ 路径（2 步）： 两步移动，每步 -1，最终获得 +5。计算：
 $RG1 = -1 + 0.9\times(-1) + 0.92\times(+5). R_{G1} = -1 + 0.9 \times (-1) + 0.9^2 \times (+5).$
 $= -1 - 0.9 + 0.81 \times 5$
 $= -1.9 + 4.05 = **2.15**.$

 到 $G2$ 路径（4 步）： 四步移动，每步 -1，最终获得 +10。计算：

 $RG2 = -1 + 0.9\times(-1) + 0.92\times(-1) + 0.93\times(-1) + 0.94\times(+10). R_{G2} = -1 + 0.9 \times (-1) + 0.9^2 \times (-1) + 0.9$

 合并计算前四项移动惩罚：$-1 - 0.9 - 0.81 - 0.729 = -3.439$。最后到达目标奖励：$0.9^4 \times 10 = 0.6561 \times 10 = 6.561$。因此
 RG2=−3.439+6.561=∗∗3.122∗∗.R_{G2} = -3.439 + 6.561 = **3.122**.

 比较两者，$R_{G2} \approx 3.122$ 大于 $R_{G1} \approx 2.15$。所以在 $\gamma = 0.9$ 下，虽然 $G2$ 更远，折扣因子较高使得智能体并未严重低估远期的奖励，**远处高奖目标 $G2$ 的总回报更大**。智能体倾向选择前往 $G2$ 的策略。

- **$\gamma = 0.5$ 时：**
 到 $G1$ 路径：
 RG1=−1+0.5×(−1)+0.52×(+5).R_{G1} = -1 + 0.5\times(-1) + 0.5^2\times(+5).
 $= -1 - 0.5 + 0.25 \times 5$
 $= -1.5 + 1.25 = **-0.25**.$

 到 $G2$ 路径：
 RG2=−1+0.5×(−1)+0.52×(−1)+0.53×(−1)+0.54×(+10).R*{G2} = -1 + 0.5\times(-1) + 0.5^2\times(-1) + 0.5^3\times(-1) + 0.5^4\times(+10).*
 前四步惩罚和：$-1 - 0.5 - 0.25 - 0.125 = -1.875$。最后奖励：$0.5^4 \times 10 = 0.0625 \times 10 = 0.625$。所以
 RG2=−1.875+0.625=∗∗−1.25∗∗.R{G2} = -1.875 + 0.625 = **-1.25**.

 在 $\gamma = 0.5$ 下，$R_{G1} \approx -0.25$ 高于 $R_{G2} \approx -1.25$（数值上 -0.25 > -1.25，表示损失更小或者说累计回报更高）。**近处的目标 $G1$ 虽奖励较小但折扣影响也小，因而总体回报优于远处的 $G2$**。智能体会倾向于先取近处的小奖励，而放弃远处的大奖励，因为对于短视的决策者而言远期收益被大幅折扣。

1. **影响分析：** 折扣因子 γ 决定了智能体对未来奖励的重视程度。

- 当 γ 较接近 1 时（如0.9），智能体是**远视的**：它不会过度贬低延后获得的奖励，因此愿意走更远的路去获取更大的回报【即使中间多付出几步的代价】。这使得远距离高回报的目标在总回报上可能胜过近距离低回报目标，正如上述 $\gamma = 0.9$ 情况下 $G2$ 占优。
- 当 γ 较小时（如0.5），智能体变得**短视**：它对未来的回报折扣非常厉害，隔得越久价值衰减越多。远处目标的高奖励在折扣下大打折扣，甚至抵消不了途中成本。因此智能体更倾向于快速获取稍小的奖励。以上 $\gamma = 0.5$ 的计算表明 $G1$ 由于近在眼前而显得相对划算。

总而言之，**较大的折扣因子鼓励策略为了长远更大利益可以忍受短期成本，较小的折扣因子则使策略更重视眼前利益。**在实际应用中，γ 的选取需要平衡：太大会使智能体考虑过远未来而学习收敛慢甚至欠优化，太小则可能使其贪图近利、陷入次优策略。

问题4：状态价值函数 V 与动作价值函数 Q 的关系

情景描述： 考虑某环境中的两个非终止状态 A 和 B（各有两个可选动作 a_1 和 a_2）。经过一段时间的学习，我们得到以下近似的动作价值函数表（Q 表）：

- 状态 A：$Q(A, a_1) = 3, \quad Q(A, a_2) = 5.$
  当前策略 π 在状态 A 选择动作 a_2。
- 状态 B：$Q(B, a_1) = 4, \quad Q(B, a_2) = 1.$
  当前策略 π 在状态 B 选择动作 a_2。

> （注：策略 π 给出了每个状态下要执行的动作。本例中，$\pi(A) = a_2, \pi(B) = a_2$。可以看出，在 B 状态策略选择的并非当前最高 Q 值动作，即策略 π 可能尚未收敛为最优。）

请回答下列问题：

1. 对于上述策略 π，求状态 A 和 B 的状态价值 $V^\pi(A)$ 和 $V^\pi(B)$。
2. 分别判断在状态 A 和 B 哪个动作的 Q 值更大。给出这两个状态的最优价值 $V^*(A)$ 和 $V^*(B)$（即由最优策略得到的状态价值），以及对应的最优动作。
3. 根据本例，简要说明状态价值函数 V 与动作价值函数 Q 之间的关系，并给出一般性的公式关系。

解析：

1. **按照策略 π 的状态价值：** 状态价值函数 $V^\pi(s)$ 定义为在状态 s 按策略 π 行动时的预期累积回报。这对于无折扣确定性环境可简单理解为所执行动作的 Q 值。因此：
   - $V^\pi(A) = Q\big(A, \pi(A)\big) = Q(A, a_2) = 5.$
   - $V^\pi(B) = Q\big(B, \pi(B)\big) = Q(B, a_2) = 1.$

   也就是说，在当前策略下，状态 A 的价值由其选择的动作 a_2 的价值决定，为5；状态 B 在策略下采取 a_2，价值为1。
2. **最优价值及动作：** 比较每个状态下两动作的 Q 值大小：
   - 状态 A：$Q(A, a_1) = 3, ; Q(A, a_2) = 5$，则**最优动作**为 a_2，对应最优价值 $V^*(A) = \max_a Q(A, a) = 5$。策略 π 在 A 恰好选择了这个最优动作 a_2，因此 A 状态策略已经是最优的。
   - 状态 B：$Q(B, a_1) = 4, ; Q(B, a_2) = 1$，则**最优动作**为 a_1，对应最优价值 $V^*(B) = \max_a Q(B, a) = 4$。当前策略在 B 却选了次优的 a_2（价值1），显然不是最优策略。如果策略改为在 B 选择 a_1，状态价值会提高到4。
3. **V-Q 关系概述：** 动作价值函数 $Q(s, a)$ 和状态价值函数 $V(s)$ 密切相关：
   - 对于**给定的策略** π，状态价值是所选动作价值的体现：$V^\pi(s) = Q\big(s, \pi(s)\big)$。也就是说，状态 s 在策略 π 下的价值等于该策略让智能体在此状态执行的动作的 Q 值。本例中，我们直接使用了这一关系来计算 $V^\pi(A)$ 和 $V^\pi(B)$。
   - **最优状态价值**是对最优动作价值的选择：$V^*(s) = \max_a Q^*(s, a)$。换言之，某状态的最优价值等于在该状态可以获得的最大动作价值。智能体寻找最优策略时，会在每个状态选择使 Q 值最大的动作，从而使状态价值达到最大。本例中，$V^*(A) = 5$ 来自于选择了使 Q 最大的 a_2，$V^*(B) = 4$ 来自于选择动作 a_1。

   因此，V 函数可以视为 Q 函数在动作维度上取最大（针对最优策略）或取策略指定的动作（针对特定策略）的结果。当策略达到最优时，有 $V^*(s) = \max_a Q(s, a)$，此时策略会选择使 Q达到最大值的动作。

问题5：Q-Learning 策略收敛性的分析与步数估计

情景描述： 我们考虑一个极简的马尔可夫决策过程：**单状态循环**。智能体只有一个状态 S（既非终点也没有其他状态），且只有一个动作可以执行。每次在状态 S 执行动作都会：获得固定奖励 $r = +1$，然后转移回状态 S 自身。这个过程将无限进行下去（相当于一个持续任务，没有终止状态）。我们希望通过 Q-Learning 来估计这个动作的价值。

设折扣因子 $\gamma = 0.9$（也就是奖励会递减地累积），初始估计 $Q_0(S, \mathrm{act}) = 0$（第一次还未尝试前价值估计为0），尝试采用较大的学习率 $\alpha = 1.0$（即每次完全采纳新样本，有助于分析收敛速率）。在这一设定下，请分析并解答：

1. 理论上，动作在状态 S 的**最优 Q 值**应为多少？（提示：计算该动作从状态 S 开始的累积折扣回报期望）。
2. 列举前几次 Q-Learning 更新的值：例如第一次执行后的 $Q_1(S, \mathrm{act})$，第二次执行后的 $Q_2(S, \mathrm{act})$，第三次 Q_3，以此类推，总结其变化规律。
3. 推导一般情况下，第 n 次执行该动作后得到的 $Q_n(S, \mathrm{act})$ 表达式。
4. 根据推导的公式，估计需要多少次左右的更新，才能使 $Q_n(S, \mathrm{act})$ 达到最优值的 90%以上（即 Q_n 相对于理论最优值的误差在10%以内）。

解析：

1. **最优 Q 值：** 由于每次执行动作都会获得立即奖励 +1，并返回同一状态 S 继续，最优策略就是一直执行该动作无穷次。折扣因子 $\gamma = 0.9$ 表示未来奖励按 0.9 的倍数递减。因此从状态 S 开始一直执行下去的预期累积回报为一个折扣无限和：
   Q∗(S,act)=1+0.9×1+0.92×1+0.93×1+⋯Q^*(S,\text{act}) = 1 + 0.9 \times 1 + 0.9^2 \times 1 + 0.9^3 \times 1 + \cdots*
   这是首项为1、公比为0.9的无限级数。其和为：
   *Q∗(S,act)=11−0.9=10.Q^*(S,\text{act}) = \frac{1}{1-0.9} = 10.
   因此理论上该动作的最优 Q 值为 **10**。

2. **迭代更新前几次值：** 使用 Q-Learning 更新公式。此处由于 $\alpha = 1$，每次更新实际上**完全使用当前样本覆盖**旧值，相当于执行贝尔曼方程的一步更新。具体更新规则为：
   Qnew=r+γ⋅Qold.Q_{new} = r + \gamma \cdot Q_{\text{old}}.
   （因为下一状态仍是 S 本身，且只有一个动作，故 $\max_{a'} Q(S, a') = Q(S, \mathrm{act}) * \mathrm{old}$。）每次执行后得到的新样本回报为 $r + \gamma Q * \mathrm{old}$。根据这一关系：
   - 第1次执行后：$Q_1 = 1 + 0.9 \times Q_0 = 1 + 0.9 \times 0 = 1.0$。
   - 第2次执行后：$Q_2 = 1 + 0.9 \times Q_1 = 1 + 0.9 \times 1.0 = 1 + 0.9 = 1.9$。
   - 第3次执行后：$Q_3 = 1 + 0.9 \times Q_2 = 1 + 0.9 \times 1.9 = 1 + 1.71 = 2.71$。
   - 第4次执行后：$Q_4 = 1 + 0.9 \times Q_3 = 1 + 0.9 \times 2.71 \approx 1 + 2.439 = 3.439$。

   可以看到 Q_n 值在逐渐增加但增量变小，逐步逼近理论最优值10。

3. **一般形式推导：** 从递推公式 $Q_n = 1 + 0.9, Q_{n-1}$ 可以解出 Q_n 关于 n 的表达式。这个递推关系是线性的，解法如下：
   首先根据特征方程或累乘法可猜测 Q_n 形式为 $Q_n = A \cdot (0.9)^n + B$。将其代入递推：
   A⋅0.9n+B=1+0.9(A⋅0.9n−1+B).A \cdot 0.9^n + B = 1 + 0.9(A \cdot 0.9^{n-1} + B).
   展开右侧：$= 1 + 0.9A \cdot 0.9^{n-1} + 0.9B = 1 + A \cdot 0.9^n + 0.9B$。对比两边，可得：
   - 系数比较：$A \cdot 0.9^n$ 两边抵消，恒等成立。
   - 常数项比较：左边常数为 B，右边常数为 $1 + 0.9B$。因此满足 $B = 1 + 0.9B$，解得 $B = \frac{1}{1-0.9} = 10$。

   带入初始条件 $Q_0 = 0$ 解 A：
   Q0=A⋅0.90+B=A+10=0,Q_0 = A \cdot 0.9^0 + B = A + 10 = 0,
   得 $A = -10$。所以：
   Qn=−10⋅(0.9)n+10=10[1−(0.9)n].Q_n = -10 \cdot (0.9)^n + 10 = 10\Big[\,1 - (0.9)^n\Big].
   这一公式也可通过累加级数直接得到：$Q_n = 1 + 0.9 + 0.9^2 + \cdots + 0.9^{n-1}$，其和为上式所示。可以验证前述数值：当 $n = 4$ 时 $Q_4 = 10[1 - 0.9^4] = 10(1 - 0.6561) = 10 \times 0.3439 \approx 3.439$，符合迭代计算结果。

4. **收敛步数估计：** 我们要求 Q_n 达到最优值的 90%以上，即：
   Qn≥0.9×Q∗=0.9×10=9.Q_n \ge 0.9 \times Q^* = 0.9 \times 10 = 9.
   利用推导的公式：
   10[1−(0.9)n]≥9.10\big[1 - (0.9)^n\big] \ge 9.
   等价于 $1 - (0.9)^n \ge 0.9$，即 $(0.9)^n \le 0.1$。两边取对数求解 n：
   nln(0.9)≤ln(0.1).n \ln(0.9) \le \ln(0.1).

由于 $\ln(0.9)$ 为负数，双侧不等式方向会反转。计算得：
n≥ln(0.1)ln(0.9).n \ge \frac{\ln(0.1)}{\ln(0.9)}.
$\ln(0.1) \approx -2.3026$，$\ln(0.9) \approx -0.1053$，比值约为 21.87。因此 n 至少约等于 **22** 次。也就是说，大约经过 22 次更新，这一状态下的 Q 值将达到 10 的 90%（即 9.0）左右。

为了直观验证，可列举几个关键迭代：$Q_{10} = 10[1 - 0.9^{10}] \approx 10[1 - 0.348678] = 6.513,$ $Q_{20} \approx 10[1 - 0.121577] = 8.784, Q_{22} \approx 10[1 - 0.090] = 9.10.$ 可以看到 $n = 22$ 时 $Q_{22} \approx 9.1$，首次超过9，满足要求。此后 Q 值继续缓慢逼近 10（理论上永远不超出10，但可无限接近）。

补充说明： 在本例中我们使用较大的 $\alpha = 1$ 来方便地观察收敛速度。实际 Q-Learning 理论收敛要求逐渐衰减的学习率等条件。但在这个确定性循环任务中，$\alpha = 1$ 也能收敛，并且如上所示按几何级数快速接近最优值。上述计算为评估**策略收敛速度**提供了依据：大约20余次迭代后，智能体对该单状态的动作价值评估已相当接近最优，策略将稳定采取该动作。这种分析方法也可用于估计更复杂任务中 Q-Learning 的收敛速度，但复杂环境下精确估计步数往往更加困难。