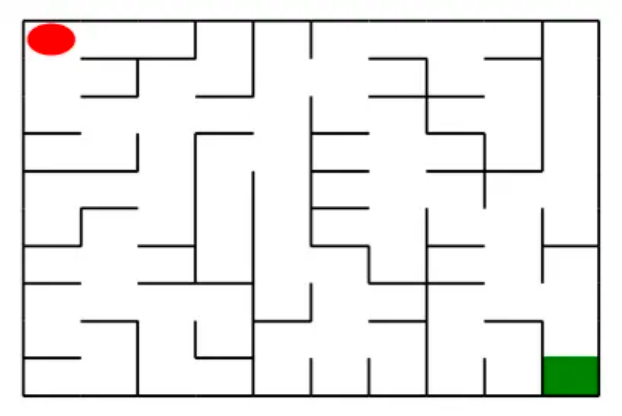
**程序报告**

学号：2212452 姓名：孟启轩

1. **问题重述**

在本实验中，要求分别使用基础搜索算法和Deep QLearning算法，完成机器人自动走迷宫。



左上角的红色椭圆既是起点也是机器人的初始位置，右下角的绿色方块是出口。

游戏规则为：从起点开始，通过错综复杂的迷宫，到达目标点(出口)。

在任一位置可执行动作包括：向上走'u'、向右走'r'、向下走'd'、向左走'l'。

执行不同的动作后，根据不同的情况会获得不同的奖励，具体而言，有以下几种情况：撞墙、走到出口、其余情况。

需要分别实现基于基础搜索算法和Deep QLearning算法的机器人，使机器人自动走到迷宫的出口。

1. **设计思想**
2. 基于实验平台实现的搜索树，来实现深度优先搜索算法，从而让机器人走出迷宫。

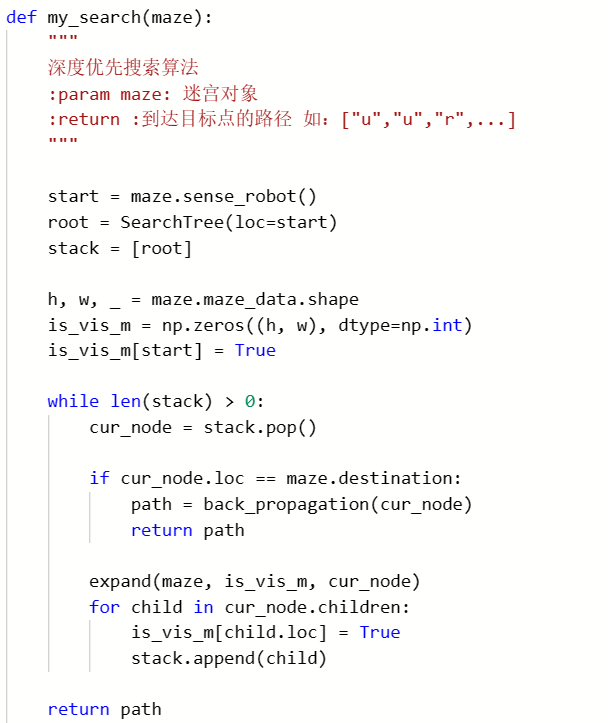
在此过程中用到了Maze类，它可以创建迷宫，并返回机器人的状态。此外，还用到了stack（栈）辅助实现深度搜索。对迷宫进行深度搜索，节点堆栈，用于层次遍历。标记迷宫的各个位置是否被访问过，并且记录路径。栈顶元素作为当前节点，标记当前节点位置已访问，当到达目标点，则退出。如果该点存在叶子节点且未拓展，则标记该点已拓展，并且开展一些列入栈操作，叶子节点入栈。如果无路可走则出栈。

1. 使用了强化学习的思想，运用了QLearning 算法实现机器人自动走迷宫。（即Deep QLearning 算法）

实现过程中用到了epsilon-greedy 算法，以及QRobot 类和Runner 类。QRobot 类实现了 QLearning 算法的 Q 值迭代和动作选择策略。在机器人自动走迷宫的训练过程中，需要不断的使用 QLearning 算法来迭代更新 Q 值表，以达到一个“最优”的状态，因此封装好了一个类 Runner 用于机器人的训练和可视化。

1. Q-learning算法是一种基于强化学习的算法，用于学习一个智能体在某个环境中采取行动的最优策略。在机器人自动走迷宫的应用中，可以将迷宫看作是一个环境，机器人则是智能体。机器人需要在迷宫中找到一条从起点到终点的最短路径。
2. **代码内容**

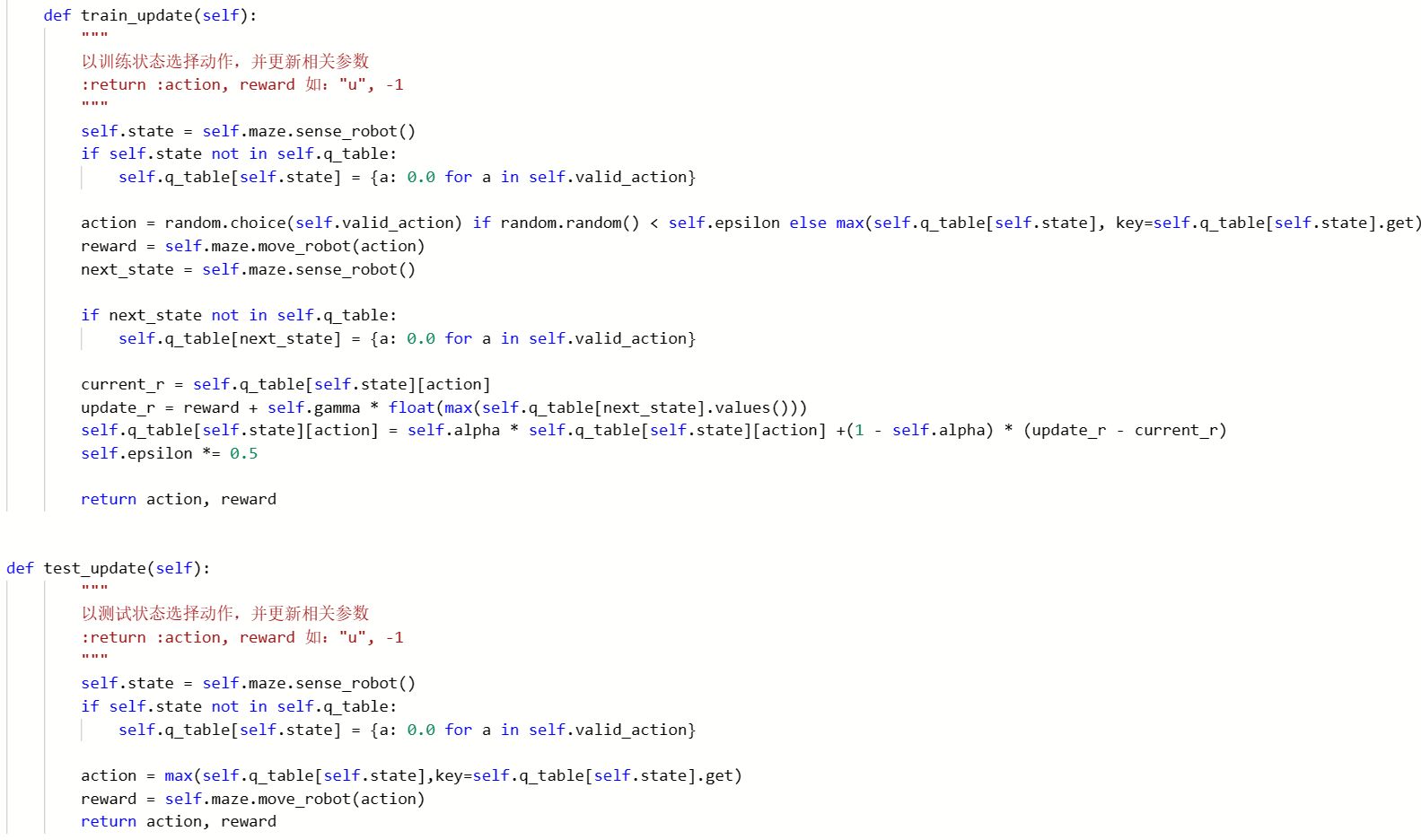
基础搜索算法：



====================================================================

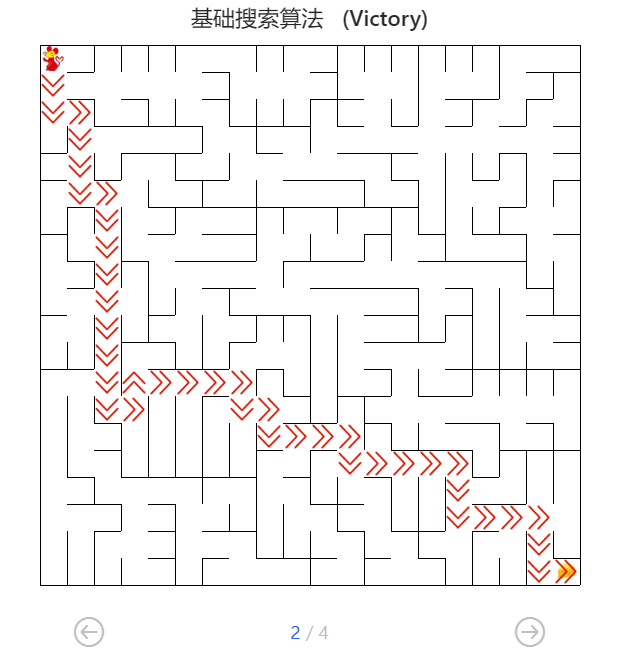
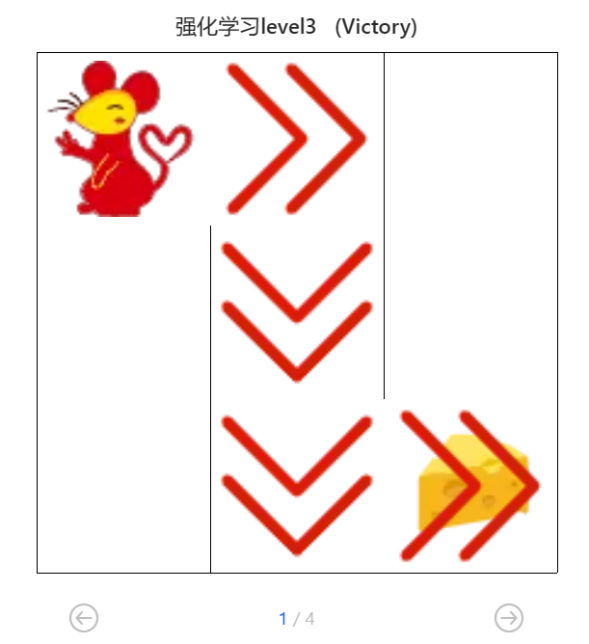
**Deep QLearning**算法：

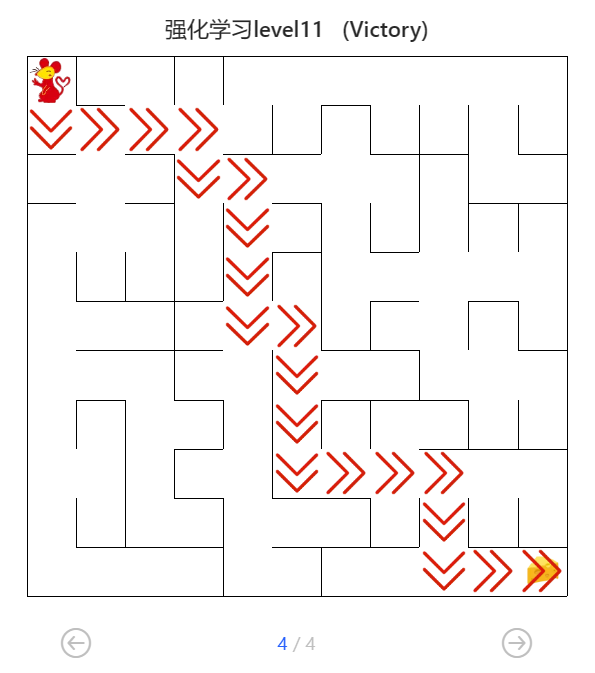
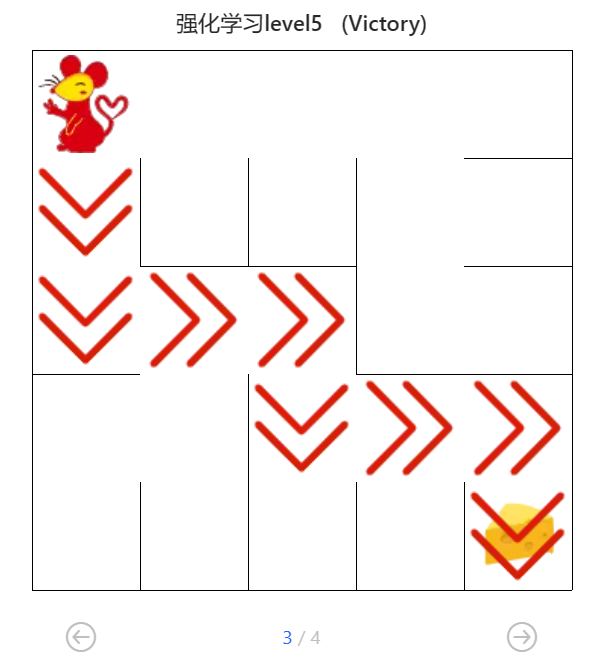




1. **实验结果**

====================================================================





1. **总结**

通过本次学习，完成了基础深度优先搜索算法以及**Deep QLearning算法为核心的机器人类**。在实验中我不仅对几种基础搜索算法有了更深刻的理解认识，也对**Deep QLearning**算法有了理解认识。后续的优化实验可以改进机器人类探索的策略，或者修改网络结构，修改奖励策略来提升机器人的性能。