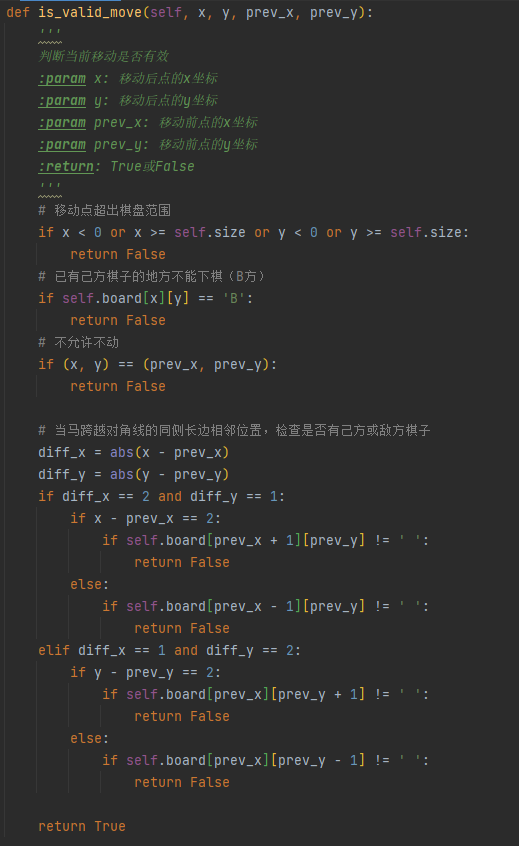
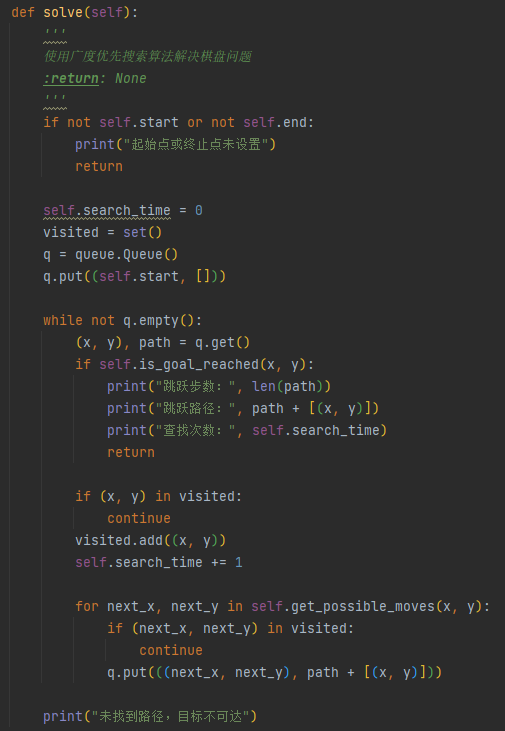
编程1报告

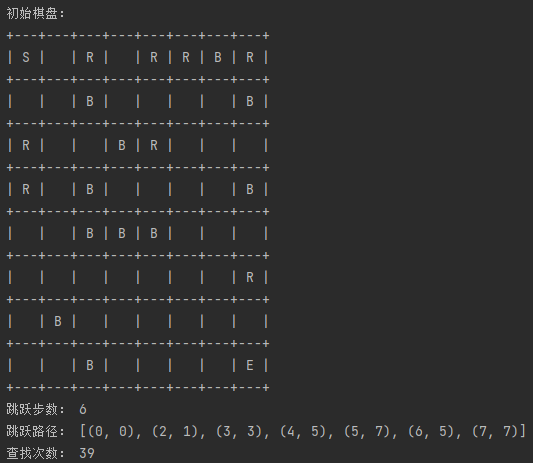
——李昭阳 2021013445

1. 算法思路

在规则设计部分，限制了棋子运动范围为棋盘内部非己方棋子已占有的点，同时根据“蹩马腿”规则制定了跳跃的判断规则。代码如下。

在实现部分，采用了广度优先搜索的策略，通过维护一个队列，访问棋盘上所有合法节点。同时，为了减小计算代价，将已访问的节点进行标记，使得每个合法节点仅被操作一次。代码如下。

1. 算法效果

在8x8的棋盘中，广度优先算法表现出较高的性能，在39次查找后找到了解，同时由于广度优先搜索具有最优性，所以可知该解为最优解。具体搜索效果如下图。

在100x100的棋盘中，广度优先算法查找了8353次才找到最优解，近乎查询了棋盘中的每一个合法点，效率过低。我认为可以采用A\*算法，以向右下角移动的距离为启发函数，鼓励算法向右下角探索终点，使得搜索速度更快。