**八皇后问题**

1. 实验目的：

了解八皇后问题，用具体算法问题理解八皇后问题算法的原理，学习用python来编写八皇后算法。

1. 实验内容：

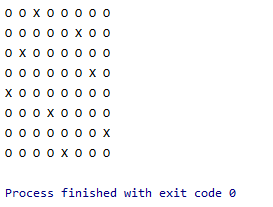
八皇后问题是以国际象棋为背景的问题：

我们要考虑怎样在 8×8 的国际象棋棋盘上放置八个皇后，使得任何一个皇后都无法直接吃掉其他的皇后，为此，任两个皇后都不能处于同一条横行、纵行或斜线上。

1. 实验代码

**import** random  
*#冲突检查，用state来标志每个皇后的位置***def** conflict(state, nextX):  
 nextY = len(state)  
 **for** i **in** range(nextY):  
  *#如果下一个皇后的位置与当前的皇后位置相邻（包括上下，左右）或在同一对角线上，则说明有冲突，需要重新摆放* **if** abs(state[i]-nextX) **in** (0, nextY-i):  
 **return True  
 return False***#产生每一个皇后的位置，并用递归来实现下一个皇后的位置。***def** queens(num, state=()):  
 **for** pos **in** range(num):  
 **if not** conflict(state, pos):  
 *#产生当前皇后的位置信息* **if** len(state) == num-1:  
 **yield** (pos, )  
 *#否则，把当前皇后的位置信息，添加到状态列表里，传递给下一皇后。* **else**:  
 **for** result **in** queens(num, state+(pos,)):  
 **yield** (pos, ) + result  
  
  
*#方便观察，用O表示空位，用X表示每个皇后的位置***def** prettyprint(solution):  
 **def** line(pos, length=len(solution)):  
 **return 'O '** \* (pos) + **'X '** + **'O '**\*(length-pos-1)  
 **for** pos **in** solution:  
 print (line(pos))  
  
**if** \_\_name\_\_ == **"\_\_main\_\_"**:  
 prettyprint(random.choice(list(queens(8))))

1. 实验结果截图



1. 实验总结

通过本次实验大大的提高了我个人对于算法的理解，也一定程度的锻炼了我的思考能力，对于八皇后问题的逻辑编排，细节完善等方面都应该有一个深刻的认识，也要继续对人工智能课程深入了解。