

用 Python 实现四连环游戏

1. 四连环是一个双人游戏，两位玩家轮流将自己的棋子放在一个 6 行 8 列的筒状棋盘上，棋盘的列号从左至右依次为 1, 2, 3, ..., 8。下棋时每个玩家只需选择列号，棋子则自动落在该列的底部，如果该列已经有棋子，则新投的棋子叠放在已有的棋子之上。游戏的目标是在对手之前将自己的 4 个相邻棋子放在一行、一列或一条斜线上。
2. 程序扮演一位玩家(劳拉)，用户作为其对手。程序采用随机方式选择要放棋子的列，用户则是手工输入要放棋子的列。
假设用户输入的都是整数，当用户输入的列号不在 1 到 8 之间时，程序会提示用户，并让用户重新输入；当用户输入的列号已经放满了棋子时，程序也会提示用户，并让用户重新输入。而当程序自己在随机地选择要放棋子的列时，它会自动地**忽略那些已经放满棋子的列**。
3. 由程序判断双方的胜负，分出胜负则程序结束。当出现棋盘满了而无法放棋子的時候，判为平局。
4. 每局游戏时 Python 界面上所显示的内容均同步保存在当前文件夹下文件名为“四连环 Log-XXXXX.txt”文件内，“XXXXX”表示一个 5 位的随机整数。
注意：不是程序运行后把界面显示的内容手工地复制粘贴到文件中。文本附件是每局的从头至尾记录，共列举了六局，各有胜负，以及一次平局。
下图是操作界面的部分截图，黑色字体的数字为用户输入，蓝色字体均为程序输出。根据题目描述，每局结束时会自动地在当前文件夹下生成一个记录文件。

```
>>>
Hi,我是劳拉，我们来玩一局四连环。我用O型棋子，你用X型棋子。
游戏规则：双方轮流选择棋盘的列号放进自己的棋子，
           若棋盘上有四颗相同型号的棋子在一行、一列或一条斜线上连接起来，
           则使用该型号棋子的玩家就赢了！

开始了！这是棋盘的初始状态：
 1 2 3 4 5 6 7 8
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
-----

>>>轮到我了,我把O棋子放在第2列...
 1 2 3 4 5 6 7 8
| | | | | | | |
| O | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
-----

>>>轮到你了,你放X棋子,请选择列号(1-8): 4
 1 2 3 4 5 6 7 8
| | | | | | | |
| O | | | | | |
| | X | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
-----
```

以下是另一局的若干截图：

开始了！这是棋盘的初始状态：

1	2	3	4	5	6	7	8

>>>轮到我了,我把0棋子放在第5列...

1	2	3	4	5	6	7	8
				0			

>>>轮到你了,你放X棋子,请选择列号(1-8): 4

1	2	3	4	5	6	7	8
			X	0			

>>>轮到我了,我把0棋子放在第2列...

1	2	3	4	5	6	7	8
	0		X	0			

>>>轮到你了,你放X棋子,请选择列号(1-8): 2

1	2	3	4	5	6	7	8
	X						
	0		X	0			

>>>轮到我了,我把0棋子放在第6列...

1	2	3	4	5	6	7	8
	X						
	0		X	0	0		

经过若干步骤之后:

>>>轮到你了,你放X棋子,请选择列号(1-8): 4

1	2	3	4	5	6	7	8
			X				
	0		X		0		
	X		X		X		
0	0	X	X	0	0		0

***** 好吧,你赢了! ^_*

>>>