

五子棋 (five-in-a-row)

项目设计人：白涛

1、游戏说明

五子棋是一种两人对弈的纯策略型棋类益智游戏，相传起源于四千多年前的尧帝时期，比围棋的历史还要悠久，可能早在“尧造围棋”之前，民间就已有五子棋游戏。古代五子棋棋盘与围棋棋盘是通用的，汉魏时为十七路（ 17×17 ）棋盘，至南北朝时即已流行十九路（ 19×19 ）棋盘，直至 1931 年出现所谓五子棋专用棋盘（ 15×15 ）。



图 1 古人下五子棋

五子棋容易上手，老少皆宜，而且趣味横生，引人入胜；不仅能增强思维能力，提高智力，而且富含哲理，有助于修身养性。主要流行于华人和汉字文化圈的国家以及欧美一些地区。各国对五子棋都有不同的爱称，例如，韩国人把五子棋称为“情侣棋”，暗示情人之间下五子棋有利于增加情感的交流；欧洲人称其为“绅士棋”，代表下五子棋的君子风度胜似绅士；日本人则称其为“中老年棋”，说明五子棋适合中老年人的生理特点和思维方式。

五子棋游戏规则是两人对局，轮流下子，先将五子连成一行者为胜。

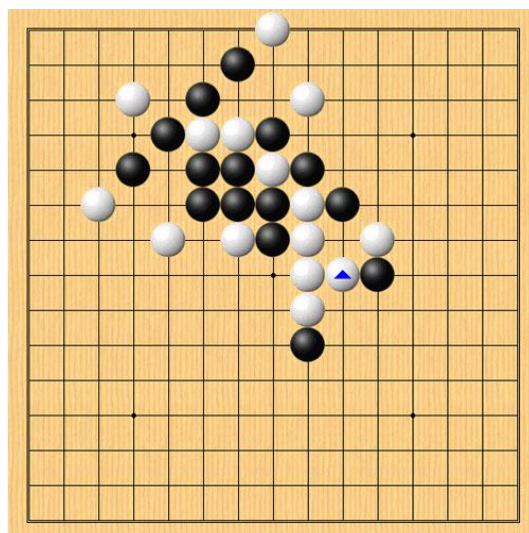


图2 五子棋棋盘

本项目所开发的五子棋在规则上与传统五子棋略有不同，游戏双方所执棋子除了黑白颜色区分外，还有中国古代四大名著人物姓名。白方人物来自《水浒》和《西游记》，黑方人物来自《三国》和《红楼梦》，游戏胜利的条件不但要求同色棋子5枚连成一行，这5枚棋子上的人物还必须来自同一部名著。这给该游戏增加了新的趣味，不但要求玩家棋艺高超，还要了解四大名著中的代表人物。

2、程序说明

为了控制程序设计的难度，本项目设计的游戏没有提供人机对战模式，只有两人对战模式，游戏双方依次用鼠标操作进行游戏。程序中判断胜利条件是一个难点，需要深刻理解自顶向下、逐步求精的程序设计思想。

在整个游戏的开发中，你需要完成以下几个任务：

(1) 利用课程提供的资源绘制游戏主界面

游戏主界面用到的图片资源包括背景图片（background.jpg）、棋盘图片（chessboard.jpg）、棋子图片（white.png 和 black.png）和游戏获胜图片（white_win.png 和 black_win.png）。

本项目中的图片的绘制只需要把装载好的图片资源按指定的大小放置在指定的位置上即可，指定位置是通过图片中心点坐标确定的。绘制时，背景大小设置为 800×600 ；棋盘居左，大小为 600×600 ，棋子图片放置在界面右上方；游戏获胜图片是由纯文字制作的图片，仅当某一方获胜时绘制。

图 4 二维矩阵示意图

如果仅考虑传统五子棋的胜利条件，即同色棋子 5 枚连续成行，对胜利条件的判断就可以转化为判断该二维矩阵中行方向、列方向、左下到右上和左上到右下四个方向是否连续出现 5 个 1 或 5 个 2。出现 5 个连续的 1 即白方获胜，出现 5 个连续的 2 即黑方获胜。

本项目中不仅要求同色而且还要求连续的棋子上人物出处一致，这样我们可以将棋子类型细分，1 代表白方《水浒》人物，2 代表白方《西游记》人物，3 代表黑方《三国》人物，4 代表黑方《红楼梦》人物，那么上述的判断就扩展为判断是否出现连续的 1、2、3 或 4，其中出现连续 1 或 2 则白方获胜，出现连续的 3 或 4 则黑方获胜。

（5）记录双方获胜局数

该功能实现非常简单，只需记录游戏过程中黑白双方获胜的次数，并绘制在界面右下角合适的位置即可。

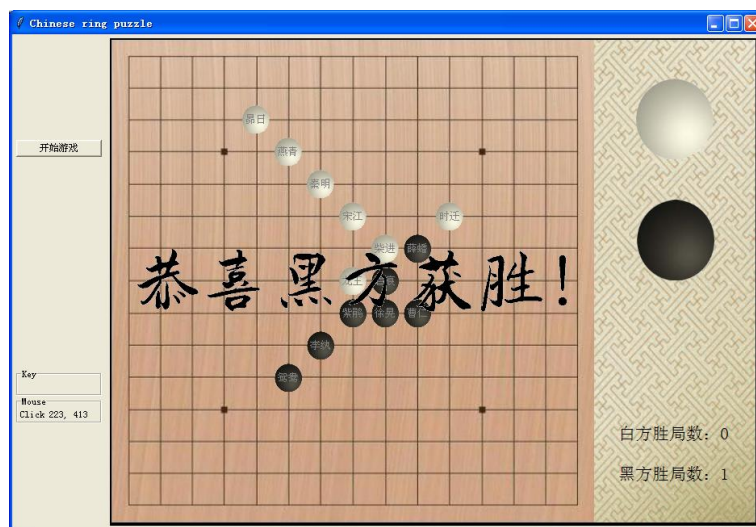


图 5 黑方获胜游戏界面

3、编码步骤

编写程序之前理解程序模版的结构非常重要，了解各个类、函数的设计目的和基本功能会使后续的编程工作轻松很多。首先程序模版中定义了两个类：ImageInfo 和 Sprite，ImageInfo 类用于描述图片资源相关信息，Sprite 类用于仅用于对环对象的描述，其中 label 属性用于设置棋子对象上显示的人物姓名，draw() 方法不仅需要绘制棋子自身还需将 label 绘制在棋子的中心位置。

模版中的其他功能的实现逐一描述如下：

(1) `start()` 函数用于初始化游戏，主要完成各个全局变量（包括棋子对象列表 `white_group`、`black_group`，棋子标签 ID `wlabelID`、`blabelID` 等）的初始化。注意，对棋子标签 ID `wlabelID`、`blabelID` 的初始化需要利用 `random.randint` 或 `random.randrange` 方法来随机生成，注意这两个方法的区别。

(2) `draw()` 函数中需要完成以下工作：

① 利用 `canvas.draw_image()` 方法绘制游戏界面的静态部分，包括背景、棋盘、棋子图标。

② 根据 `start()` 中初始化的各个变量，利用 `canvas.draw_image()` 方法绘制获胜信息，利用 `canvas.draw_text()` 绘制双方获胜局数；

③ 遍历白棋棋子列表 `white_group` 和黑棋棋子列表 `black_group`，调用自身的 `draw()` 方法进行绘制。

(3) `mouseclick(pos)` 方法中需要做两件事：

① 根据 `pos` 判断鼠标是否点击了棋子图标，如果条件成立，必须以全局变量 `oldlabelID` 记录棋子上当前的人物姓名 ID 用于创建棋子对象，并根据当前人物姓名 ID 设置全局变量 `piecesID` 的值。以点击白棋图标为例，当 ID 小于 20，全局变量 `piecesID` 设置为 1；否则全局变量 `piecesID` 设置为 2。以 20 为判定条件的原因是在白方棋子标签列表 `white_label` 中，0 至 19 的人物来自《水浒》，20 至 39 的人物来自《西游记》。最后再随机生成下一个棋子对应的人物姓名 ID。

② 当 `piecesID ∈ (1, 2, 3, 4)` 时，表示玩家已经选取了对应的棋子，因此需要根据 `pos` 来计算棋子应该放置的坐标，并调用 `pieces_spawner()` 函数创建棋子对象。同时，还需将二维数组对应位置的值设置为 `str(piecesID)`：

```
matrix[pos[1] // 40][pos[0] // 40] = str(piecesID)
```

注意，二维数组 `matrix` 的定义和初始化以及 `pieces_spawner()` 函数的实现将在下面描述。

(4) 定义二维数组 `matrix` 是为了存储棋子的分布信息，以计算胜利条件。在 Python 中，可以通过 `list` 来定义二维数组，如下所示：

```
matrix = [[0 for col in range(15)] for row in range(15)]
```

初始化函数 `init_matrix()` 通过两层循环，将二维数组的元素设置为字符 '0'，注意此处不能是整数 0，否则后续判断会比较麻烦。

(5) `pieces_spawner()` 函数需要完成 3 项任务：

① 创建棋子对象，创建的棋子对象类型由参数 `piecesID` 确定，1 代表白方《水浒》人物，2 代表白方《西游记》人物；3 代表黑方《三国》人物，4 代表黑方《红楼梦》人物；棋子类型的选择以及放置的位置和鼠标点击的坐标有关，所以 `mouseclick(pos)` 方法在调用 `pieces_spawner(piecesID, pos)` 时需要传递 `piecesID` 和 `pos` 两个参数，此外创建棋子对象时，`label` 属性对应的 ID 应设置为 `oldlabelID`。

② 放置前必须遍历两个棋子对象列表 `white_group` 和 `black_group`，通过 `get_position()` 获得位置并和参数 `pos` 比较，判断当前位置是否已经有棋子，以此来确定是否能够放置新棋子，此过程中可能会用到 `list.extend()` 方法。

③ 当创建并放置棋子后，还需立即调用 `check_matrix()` 函数判断是否满足胜利条件，如果满足则更新胜利方获胜局数，重置全局变量 `started`。

(6) `check_line(line)` 函数的功能比较简单，主要是判断列表 `line` 中是否存在连续出现 5 次字符 ‘1’、‘2’、‘3’ 或 ‘4’，连续出现 5 次 ‘1’ 或 ‘2’，返回 1；连续出现 5 次 ‘3’ 或 ‘4’，返回 2。该函数被 `check_matrix()` 调用以扫描整个二维数组 `matrix`，实现判断胜利条件的功能。

(7) `check_matrix()` 函数主要对二维数组 `matrix` 四个方向（行方向、列方向、左下到右上和左上到右下）的扫描，对扫描到的元素创建一个临时列表，并在调用 `check_line(line)` 时作为参数传递，判断整个二维数组 `matrix` 中是否连续 5 次出现字符 ‘1’、‘2’、‘3’ 或 ‘4’，返回 1 表示白方获胜，返回 2 表示黑方获胜。

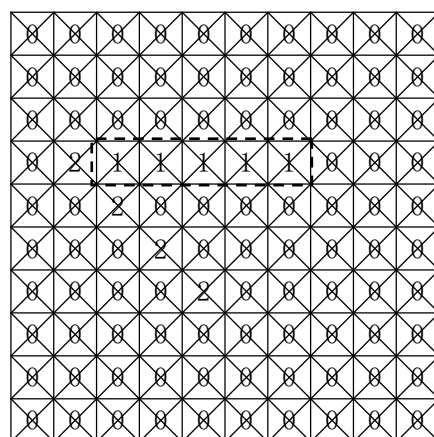


图 6 四个方向扫描二维矩阵示意图

该函数功能的实现难点在于对扫描方向和循环的控制，行列方向非常简单，单层循环即可实现，斜方向比较复杂，每次只能通过两层循环扫描矩阵的上三角

或下三角。对于循环变量的边界值和二维数组元素坐标的选择要非常小心。

(8) 添加 `button` 控件，并调用 `start()` 函数启动游戏。