

CowaveSnake

控维贪吃蛇对抗平台开发版本

1 游戏规则

1. 以小队形式参赛，参赛队伍成员不超过三个；
2. 每个参赛小队需要提供两个程序，一个程序控制贪吃蛇，一个程序控制食物生成；
3. 现场随机抽签，捉对厮杀；
4. A、B双方对抗形式为：
 - A组的贪吃蛇程序与B组的食物刷新程序竞技；
 - B组的贪吃蛇程序与A组的食物刷新程序竞技；

获胜条件

- 游戏时间上限到达后得分更高的组获胜。
- 达到分数上限时间耗时更短的组获胜。

规则说明

- 游戏窗口为一个矩阵（大小暂定21*21）。
- 贪吃蛇移动过程中撞击游戏边界、身体或反向移动时游戏结束。
- 游戏达到时间上限（暂定3分钟）或分数上限（暂定120）时游戏结束。
- 食物刷新程序计算食物位置时间超时（暂定1秒），程序将自动在贪吃蛇头附近生成一个食物。（防止有的组不刷新食物导致无法得分）。
- 移动路线计算程序超时（暂定10秒），贪吃蛇将自动往随机方向移动一格，并进入下次"生成食物 > 计算路线"的游戏循环。
- 贪吃蛇必须把计算的路线全部移动完成后才会刷新食物。（所以不要多生成移动路线，节约时间成本）
- 贪吃蛇完成移动路线并未吃到食物后，本次游戏循环不得分，并进入下次"生成食物 > 计算路线"的游戏循环。（所以每次路线一定要计算准确）

以下内容中：

- x：横坐标，自左向右，从0开始。
- y：纵坐标，自上向下，从0开始。
- 游戏矩阵：二维列表。外层代表行，与纵坐标对应；内层代表列，与横坐标对应。

1.1 控制程序说明

- 控制程序入参为：游戏矩阵（"s"代表蛇身体，1代表食物）、食物位置（一个[x, y]）、蛇头坐标、蛇当前移动方向、蛇身体（列表，存放着所有矩阵中"s"元素的坐标，第一个为蛇尾，最后一个为蛇头）、游戏窗口大小（[rows, cols]）。
- 控制程序的返回值必须为移动路线的 List，["U", "D", "L", "R"] 分别代表移动方向为上、下、左、右。返回值例：["D", "D", "L", "U", "R"]。
- 控制程序的出参出现一下情况将导致游戏结束：
 - 非可迭代对象
 - 每次迭代的值不为 ["D", "U", "L", "R"] 其中之一
- 控制程序出现如下状况蛇将自动向附近移动一格：

- 超时

1.2 食物生成程序说明

- 食物生成程序入参为：游戏矩阵（"s" 代表蛇身体，1 代表食物）、蛇头坐标、蛇当前移动方向、蛇身体（列表，存放着所有矩阵中"s" 元素的坐标，第一个为蛇尾，最后一个为蛇头）、游戏窗口大小（[rows, cols]）。
- 食物生成程序的返回值必须为一个游戏矩阵内的 [x,y] 坐标，且该位置不可以是蛇身体。
- 食物生成程序出现如下情况将自动在蛇头附近生成一个食物：
 - 超时
 - 生成的食物超出矩阵范围
 - 生成的食物坐标为蛇的身体

2 使用说明

将参赛队员代码放置于 `players` 目录下即可，示例目录结构如下：

```
1 | players *
2 | -- 参赛A队 *
3 |   -- GenerateMove.py
4 |   -- GenerateSnack.py
5 | -- 参赛B队 *
6 |   -- GenerateMove.py
7 |   -- GenerateSnack.py
```

在打开的游戏窗口中按空格即可开始游戏。游戏结束后按空格可关闭窗口。

2.1 单独比赛

使用 `python run1 [参赛A队名称] [参赛B队名称]` 命令将使用A队的移动程序与B队的食物生成程序进行对抗。

2.2 同时比赛

使用 `python run2 [参赛A队] [参赛B队]` 命令，将同时打开两个窗口，使得A、B队的移动程序与食物生成程序分别对抗。**窗口名称代表移动程序队伍名称。**