游戏规则

本游戏为**四人回合制**游戏，在矩形场地上进行。

规则比较复杂，但是#样例程序（C++版本）已经完全实现了规则的所有细节，并且有着充足的注释。

**地图**

游戏场地的长和宽都是[6, 12)的整数。

游戏场地上的基本单位是格子，每个格子的四周可能具有墙面、格子中央可能有内容。

裁判会自动生成随机地图，随机地图保证整个场地**完全连通**，场地沿着两条中轴线**轴对称**，这包括场地的初始道具、墙面、玩家初始位置等。

游戏场地的**最外部可能不是封闭的**。

**进行任何计算时，越界的坐标都会通过对宽或高取余而被折算回场地内**，换句话说，从场地最下方向下走，可以从场地最上方回来。

**人物**

每名玩家控制地图上的一个人物。人物只能处于格子中。同一格子中可能有多个人物。

人物的序号在程序中分别是0、1、2、3，在界面上是1、2、3、4.

在地图上，每个人物会被分配一个初始位置。

人物不能穿墙。尝试穿墙将会导致被判定为不合法输入而死亡。

人物的属性有

* 力量（代码中记为strength）：表示自己的实力，游戏结束时**仅依据游戏结束瞬间的力量（也包括力量增益）确定名次**。
* 增益剩余回合（代码中记为powerUpLeft）：表示自己由于使用大豆子而获得的力量增益还剩几回合。
* 是否死亡

**道具**

游戏中每个格子上只能有一个道具。

地图生成的时候会有初始道具，对称分布于场地上。

游戏中的道具包括

* 小豆子：玩家可以食用，永久增加1力量。
* 大豆子：可以在【LARGE\_FRUIT\_DURATION】回合内让玩家的力量增加【LARGE\_FRUIT\_ENHANCEMENT】（目前两个变量的值都是10）
* 豆子产生器（「翠柰木」）：每隔【GENERATOR\_INTERVAL】回合，在周围八个方向距离1的格子上尝试，如果格子上没有道具就会放置小豆子。（目前这个变量的值是20）

注意，**豆子产生器是障碍物**，玩家不能移动到它身上，否则会被判定为不合法输入而死亡。

（实际上这不用考虑，产生地图时会**保证豆子产生器被墙完全围住**）

**动作**

每回合，玩家需要决定自己人物的行动。行动包括选择上、右、下、左任意一个方向行进一格，或不动。

如果你有兴趣，在选择了行动之后，你还可以提供一个字符串，让人物进行叫嚣。**这不会影响你的行动**，不过字符串最大长度为40，过长会被自动截断。由于编码问题，建议不要使用中文，或保证输出的中文是经过UTF-8编码。在Botzone上，你可以通过*我的Bot*页面，*增加版本*，然后使用在线编辑器上传代码，而不是直接上传代码文件，来保证文件是UTF-8编码。

移动有可能失败，这种失败的情况称为被“恐吓”。当且仅当你将要移动到的格子上有比你力量强大的人物时，你的行动会被系统替换成“不动”。具体流程可以参看#游戏每回合的判定流程。

如果你移动到的格子上有道具，那么道具会生效，除非该格子上的玩家超过1个，此时道具无法生效。

**战斗**

如果一个格子上有多名玩家，而且玩家的力量有不同，那么玩家会进行战斗判定。具体流程可以参看#游戏每回合的判定流程。

**力量最大的所有玩家是战胜者**，其他玩家是战败者。

战败者会失去一半的力量并死亡。（向上取整）

战胜者会获得另一半力量并平分。（向下取整）

**游戏每回合的判定流程**

1. 分析输入，所有不合法输入者被杀死，这样死亡的玩家**失去所有力量**。
   * **不合法输入**包括移动方向有墙的情况，也包括程序输出格式不对、超时、超内存等情况。
   * 目的地（考虑边界折回）有比自己力量大的玩家的话，自己的移动会被强制替换为不动（“恐吓”）
2. 所有选择移动的玩家向指定方向移动一格（**超过边界，则折回到另一侧**）
3. 现在检查玩家重叠的情况，如果**一个格子上有多个玩家**，那么在这几个玩家中
   * 力量最大的玩家（们）将会吞噬其他玩家
   * 被吞噬者将会全部死亡，其分数锁定为各自力量的一半（**向上取整**）
     + 具有大豆子增益效果的玩家，大豆子赋予的力量不会丢失，且再也不会消失
   * 存活者们平分败者力量的另一半（**向下取整，平分时也向下取整**）
   * 此时格子上玩家力量均相等，结束
4. 豆子产生器工作（超过边界折回，不能生成在产生器上）
5. 如果玩家所在的格子上有豆子，且**仅当格子上只有一个玩家**时
   * 玩家吃掉小豆，力量增加1
   * 玩家吃掉大豆
     + 如果增益剩余回合还未归零，那么累加【LARGE\_FRUIT\_DURATION】
     + 如果增益剩余回合为0，力量增加【LARGE\_FRUIT\_ENHANCEMENT】，并且设置增益剩余回合为【LARGE\_FRUIT\_DURATION】
6. 所有玩家的大豆剩余时间减少1，变为0的力量减少【LARGE\_FRUIT\_ENHANCEMENT】
7. 回合序号加一
8. 如果此时只剩一位玩家，开始进行游戏结束的结算
   * 只剩一位玩家时，该玩家立刻获得场上所有小豆
9. 回合超限，开始进行游戏结束的结算

**分数**

玩家的分数是玩家力量（并列）倒数第几。

比如1~4号玩家力量分别是23,233,233,2333，那么他们的分数为1,2,2,4。

游戏交互方式

**强烈建议**初学者直接参看[#样例程序（C++版本）](http://wiki.botzone.org/index.php?title=Pacman#.E6.A0.B7.E4.BE.8B.E7.A8.8B.E5.BA.8F.EF.BC.88C.2B.2B.E7.89.88.E6.9C.AC.EF.BC.89)，样例程序隐藏了具体交互的细节。

**与[Botzone](http://wiki.botzone.org/index.php?title=Botzone" \o "Botzone)上其他游戏一样，本游戏每步（每次程序运行）限时1秒。**

**如果需要输出调试信息，请在输出的JSON对象中加debug域（或修改WriteOutput函数的ret["debug"]），调试信息将可在Log可视化工具中查看。**

**概述**

本游戏与Botzone上其他游戏一样，使用相同的交互格式：[Bot#交互](http://wiki.botzone.org/index.php?title=Bot" \l ".E4.BA.A4.E4.BA.92" \o "Bot)

另外该游戏是支持初始化数据的游戏。初始化数据格式如下（最好写成单行[JSON](http://wiki.botzone.org/index.php?title=JSON" \o "JSON)）：

{

"seed": Number, *// 种子。拥有一个种子，就拥有了全世界。*

"width": Number, *// 地图的宽度*

"height": Number, *// 地图的高度*

"GENERATOR\_INTERVAL": Number, *// 产生器间隔*

"LARGE\_FRUIT\_DURATION": Number, *// 大豆子增益回合数*

"LARGE\_FRUIT\_ENHANCEMENT": Number, *// 大豆子增益力量*

"static": number[][], *// 记录场地不变信息的二维数组（解析方法请参看PrepareInitialField）*

"content": number[][] *// 记录场地可变信息的二维数组*

}

**具体交互内容**

再次**强烈建议**初学者直接参看[#样例程序（C++版本）](http://wiki.botzone.org/index.php?title=Pacman#.E6.A0.B7.E4.BE.8B.E7.A8.8B.E5.BA.8F.EF.BC.88C.2B.2B.E7.89.88.E6.9C.AC.EF.BC.89)，样例程序隐藏了具体交互的细节。

第一回合玩家会收到上述的初始化数据，是一个JSON对象。

此后回合玩家收到一个JSON对象，表示每个玩家的动作。具体格式如下：

{

"0": {

"action": Number,

"tauntText": String

},

"1": {

"action": Number,

"tauntText": String

},

"2": {

"action": Number,

"tauntText": String

},

"3": {

"action": Number,

"tauntText": String

}

}

注意死亡的玩家将没有对应的动作信息对象。

每回合玩家也应该输出自己的动作，格式如下：

{

"action": Number,

"tauntText": String

}

其中，action是[-1, 4)的整数，分别表示“不动、向上、向右、向下、向左”五种动作类型。

tauntText是本回合叫嚣的文字。可以留空，超过40个字节部分将会被截断。由于编码问题，建议不要输出中文，或保证输出的中文是UTF-8编码。在Botzone上，你可以通过*我的Bot*页面，*增加版本*，然后使用在线编辑器上传代码，而不是直接上传代码文件，来保证文件是UTF-8编码。