

# “冰雪杯”赛道游戏细则

软院合开发组

2022 年 4 月 6 日

## 1 游戏介绍

“冰雪杯”组由软件学院主导开发的即时对抗游戏。

3 名选手各控制一支至多四个队员的队伍在圆形滑冰场上进行即时对抗，队员需要推冰橇来收集场地上刷新的“冰墩墩”和“雪容融”(下简称“冰墩墩”)，并送回自己的球门内；同时，还要防止其他队伍偷窃自己的“冰墩墩”。在规定时间内收集到最多数量的队伍将胜出。

支持平台：win32(x64), Linux(x64), Mac OS(x64)

参赛语言：C/C++, Python

### 1.1 详细规则

- 基础属性：每个角色碰撞箱为直径 0.48 m 的圆形；可以奔跑，可以步行，可以静止；雪橇中有“冰墩墩”的时候不能奔跑，收集“冰墩墩”时如果处于奔跑状态会立刻进入步行状态；每个角色有一个初始为 5 的耐力值，奔跑时每秒减少 4 点耐力，耐力 0 时不能再奔跑；静止时每秒回复 1 点耐力，步行时若雪橇中没有“冰墩墩”，每秒回复 0.5 点耐力，回复到 5 为止，否则不回复耐力。奔跑速度为 4m/s，步行速度为 2m/s，其中编号为 4 的整数倍的小人最大速度是其他角色的 1.2 倍，雪橇中有“冰墩墩”时速度为  $(3 - 1.07^{s-10})m/s$ (s 为“冰墩墩”分数，在 10-20 间随机)；游戏内部以每秒 60 的帧速率进行物理模拟，并每 6 帧(也就是 100ms)与选手 AI 进行一次信息交互，这是选手 AI 发出操作和获取最新场内信息的时间粒度，如果某回合 AI 未在指定时间内给出指令，那么选手会被判为掉线，不参与后续回合。
- 小心碰撞：**游戏中的所有碰撞均为弹性碰撞**。玩家如果发生碰撞则会停下，并在碰撞结束后失去控制一段时间(1.5 s)；
- 游戏场地：比赛场地主体为直径 40 m 的圆形，分为三个部分，每个部分都是扇形且圆心角 120°；每个扇形边缘有一个宽度 10 m，长度 5 m 的长方形，是各支队伍的得分区域。得分区域两个角刚好在圆周上，如图 1；在每条扇形分割线上均匀分布 5 个冰墩墩(考虑端点时满足等距离)；在场地外围有墙挡住；在场地内有一个内半径 18 m，外半径 20 m 的圆环减速带，角色在其上速度满足恒不大于 0.5 m/s；
- 冰上争夺：冰墩墩碰撞箱为一个直径 0.7 m 的圆形。角色成功抱起冰墩墩的条件是：
  1. 冰墩墩在地上且角色外边缘和冰墩墩表面距离不超过 0.1 m(即，角色与冰墩墩的圆心距小于半径之和加此距离)；如果多个角色满足此条件，且在同一个时刻尝试抱起，则距离冰墩墩最近的成功抱起；

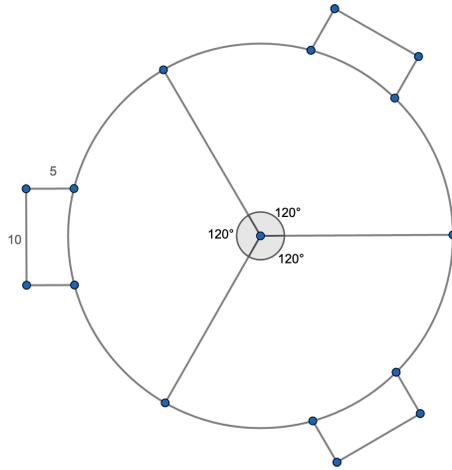


图 1: 比赛场地俯视图

- 冰墩墩在另一个角色身上（无论敌我）；那么，抱起冰墩墩的角色会把冰墩墩放在车中，即冰墩墩碰撞箱中心和角色碰撞箱中心重合；距离要求与上条相同；如果有 multiple 个人尝试抢，则最近的成功抢过来；
- 精准入篮：放下冰墩墩时，可以在所有保证冰墩墩和角色相切的位置放置；但是冰墩墩不能卡在别人身上，也不能和其他冰墩墩重叠或卡入地图边界；冰墩墩在外围 3 个区域内时判断为入篮，具体判断点为冰墩墩的中心。

## 2 参赛流程

### 2.1 初赛截止时间

2022 年 4 月 24 日。

### 2.2 加入小组

请在小组页面加入本次比赛的小组。

### 2.3 下载游戏包

在右上角的「下载游戏包」按钮下载游戏包。游戏包内含：

- 开发 SDK 与简单的样例 AI；
- 本地评测程序；
- 开发说明。

### 2.4 编写 AI

请参考游戏包中的开发说明与样例，选择你喜欢的语言编写 AI。详细的配置说明在开发说明中都有提及。

## 2.5 提交代码

在「我的 AI」处提交自己的 AI，可根据实际情况选择对应编译语言，编译成功后便可将 AI 派遣到对应比赛。

## 2.6 派遣 AI

您可以在天梯上派遣 AI，并发起对战，以观测己方与其他选手的策略博弈。天梯成绩仅供参考，不计入最终比赛名次成绩。

## 2.7 观看回放

您可以使用 Saiblo 在对局结束后打开的窗口中的播放器。

或者点击右上角的「下载回放」按钮将回放文件下载后，在 Saiblo 的「网页播放器」中查看回放文件。

## 2.8 决出名次

初赛时间截止后，我们会在提交了代码的选手间发起大量对局并统计积分，直至排名收敛为止。

## 2.9 测试方法

### 2.9.1 本地测试

我们提供了一个本地裁判程序（下面假设你把它放在 `./judge.py`）用于评测两个本地 AI，具体使用方法为：

1. 下载裁判程序、游戏逻辑（假设命名为 `eggs`），放在 SDK 根目录。
2. 我们假设你采用 Python SDK，且当前命令行在 `main.py` 所在目录。执行下列命令，进入调试模式：

```
$ python ./judge.py test_mode
```

进入调试模式后界面如下：

```
You can input help to know the instruction set
>
```

3. 在调试模式依次逐行输入下列语句，以使用 `main.py` 中的 AI 分别控制 3 个队伍，运行一次测试：

```
0 0 python+main.py
0 1 python+main.py
0 2 python+main.py
4 eggs dummyConfig replay.bin
```

对局完成后，会将对局文件存入 `replay.bin`。

注意：请确保本地执行 `python` 命令呼出的是 PYTHON 3。

### 2.9.2 评测逻辑文件下载

出于安全考虑，评测逻辑只提供构建好的 Linux 版本，及游戏包中的 `snow` 文件。也可以直接采用下列链接。下载后即可采用 `judge.py` 和评测逻辑文件配合，进行测试。Windows 可以使用 WSL，Mac 用户可以使用直接使用网页测评器。（如果后续需求较大，我们会额外提供 Windows 和 Mac 的版本）

请注意：考虑到选手电脑的性能问题，我们在用于本地评测的逻辑中放宽了时间限制；实际在 Saiblo 评测时，单次 `update` 的时间限制是 0.1 s. 如果任何一个回合超出这个限制，对应的 AI 此回合的回复会被抛弃，但是你的 AI 仍然可以参与后续回合的交互。

关于 AI 调试：如果你想要看到更多来自游戏逻辑的调试信息，只需要重命名逻辑文件名使得其包含 `verbose` 这一字符串即可。你可能需要对上述调试模式输入语句的最后一行做适当更改，以匹配重命名后的逻辑文件名。