

CAPI接口(python)

接口解释

主动指令

移动

使用技能

人物

攻击

学习与毕业

勉励与唤醒

地图互动

道具

信息获取

队内信息

查询可视范围内的信息

查询特定位置物体的信息

其他

辅助函数

接口一览

CAPI接口(python)

接口解释

主动指令

移动

- `def Move(self, timeInMilliseconds: int, angle: float) -> Future[bool]`:移动, `timeInMilliseconds` 为移动时间, 单位毫秒; `angleInRadian` 表示移动方向, 单位弧度, 使用极坐标, **竖直向下方向为x轴, 水平向右方向为y轴**
- `def MoveRight(self, timeInMilliseconds: int) -> Future[bool]` 即向右移动, `MoveLeft`、`MoveDown`、`MoveUp` 同理

使用技能

- `def UseSkill(self, skillID: int) -> Future[bool]`:使用对应序号的主动技能

人物

- `def EndAllAction(self) -> Future[bool]`:可以使不处在不可行动状态中的玩家终止当前行动

攻击

- `def Attack(self, angle: float) -> Future[bool]`:`angleInRadian` 为攻击方向

学习与毕业

- `def StartLearning(self) -> Future[bool]` :在教室里开始做作业
- `def StartOpenGate(self) -> Future[bool]` :开始开启校门
- `def Graduate(self) -> Future[bool]` :从开启的校门或隐藏校门毕业。

勉励与唤醒

- `def StartEncourageMate(self, mateID: int) -> Future[bool]` :勉励对应玩家ID的学生。
- `def StartRouseMate(self, mateID: int) -> Future[bool]` : 唤醒对应玩家ID的沉迷的学生。

地图互动

- `def OpenDoor(self) -> Future[bool]` :开门
- `def CloseDoor(self) -> Future[bool]` :关门
- `def SkipWindow(self) -> Future[bool]` :翻窗
- `def StartOpenChest(self) -> Future[bool]` :开箱

道具

- `def PickProp(self, propType: THUAI6.PropType) -> Future[bool]` 捡起与自己处于同一个格子 (cell) 的道具。
- `def UseProp(self, propType: THUAI6.PropType) -> Future[bool]` 使用对应类型的道具
- `def ThrowProp(self, propType: THUAI6.PropType) -> Future[bool]` 将对应类型的道具扔在原地

信息获取

队内信息

- `std::future<bool> SendMessage(int64_t, std::string)` : 给同队的队友发送消息。第一个参数指定发送的对象, 第二个参数指定发送的内容, 不得超过256字节。
- `bool HaveMessage()` :是否有队友发来的尚未接收的信息。
- `std::pair<int64_t, std::string> GetMessage()` :从玩家ID为第一个参数的队友获取信息。

查询可视范围内的信息

- `std::vector<std::shared_ptr<const THUAI6::Student>> GetStudents() const` : 返回所有可视学生的信息。
- `std::vector<std::shared_ptr<const THUAI6::Tricker>> GetTrickers() const` : 返回所有可视捣蛋鬼的信息。
- `std::vector<std::shared_ptr<const THUAI6::Prop>> GetProps() const` : 返回所有可视道具的信息。
- `std::vector<std::shared_ptr<const THUAI6::Bullet>> GetBullets() const` : 返回所有可视子弹 (攻击) 的信息。

查询特定位置物体的信息

下面的 CellX 和 CellY 指的是地图格数，而非绝对坐标。

- `def GetPlaceType(self, cellX: int, cellY: int) -> THUAI6.PlaceType` : 返回某一位置场地种类信息。场地种类详见 structure.h。
- `def IsDoorOpen(self, cellX: int, cellY: int) -> bool`: 查询特定位置门是否开启
- `def GetChestProgress(self, cellX: int, cellY: int) -> int`: 查询特定位置箱子开启进度
- `def GetGateProgress(self, cellX: int, cellY: int) -> int`: 查询特定位置校门开启进度
- `def GetClassroomProgress(self, cellX: int, cellY: int) -> int`: 查询特定位置教室作业完成进度
- `def GetHiddenGateState(self, cellX: int, cellY: int) -> THUAI6.HiddenGateState` : 查询特定位置隐藏校门状态
- `def GetDoorProgress(self, cellX: int, cellY: int) -> int`: 查询特定位置门开启状态

其他

- `def GetGameInfo(self) -> THUAI6.GameInfo`: 查询当前游戏状态\
- `def GetPlayerGUIDs(self) -> List[int]`: 获取所有玩家的GUID\
- `def GetFrameCount(self) -> int`: 获取目前所进行的帧数\
- `def GetSelfInfo(self) -> Union[THUAI6.Student, THUAI6.Tricker]` : 获取自己的信息
- `def GetFullMap(self) -> List[List[THUAI6.PlaceType]]` : 返回整张地图的地形信息。

辅助函数

`def CellToGrid(cell: int) -> int`: 将地图格数 cell 转换为绝对坐标 grid。

`def GridToCell(grid: int) -> int`: 将绝对坐标 grid 转换为地图格数 cell。

下面为用于DEBUG的输出函数，选手仅在开启Debug模式的情况下可以使用

```
1     def Print(self, cont: str) -> None:
2     def PrintStudent(self) -> None:
3     def PrintTricker(self) -> None:
4     def PrintProp(self) -> None:
5     def PrintSelfInfo(self) -> None:
```

接口一览

```
1     class IAPI(metaclass=ABCMeta):
2
3         # 选手可执行的操作
4         # 指挥本角色进行移动, `timeInMilliseconds` 为移动时间, 单位为毫秒; `angleInRadian`
          表示移动的方向, 单位是弧度, 使用极坐标——竖直向下方向为 x 轴, 水平向右方向为 y 轴
5
6         @abstractmethod
```

```
7     def Move(self, timeInMilliseconds: int, angle: float) -> Future[bool]:
8         pass
9
10    # 向特定方向移动
11
12    @abstractmethod
13    def MoveRight(self, timeInMilliseconds: int) -> Future[bool]:
14        pass
15
16    @abstractmethod
17    def MoveLeft(self, timeInMilliseconds: int) -> Future[bool]:
18        pass
19
20    @abstractmethod
21    def MoveUp(self, timeInMilliseconds: int) -> Future[bool]:
22        pass
23
24    @abstractmethod
25    def MoveDown(self, timeInMilliseconds: int) -> Future[bool]:
26        pass
27
28    # 道具和技能相关
29
30    @abstractmethod
31    def PickProp(self, propType: THUAI6.PropType) -> Future[bool]:
32        pass
33
34    @abstractmethod
35    def UseProp(self, propType: THUAI6.PropType) -> Future[bool]:
36        pass
37
38    @abstractmethod
39    def ThrowProp(self, propType: THUAI6.PropType) -> Future[bool]:
40        pass
41
42    @abstractmethod
43    def UseSkill(self, skillID: int) -> Future[bool]:
44        pass
45
46    @abstractmethod
47    def Attack(self, angle: float) -> Future[bool]:
48        pass
49
50    @abstractmethod
51    def OpenDoor(self) -> Future[bool]:
52        pass
53
54    @abstractmethod
55    def CloseDoor(self) -> Future[bool]:
56        pass
57
58    @abstractmethod
```

```
59     def skipwindow(self) -> Future[bool]:
60         pass
61
62     @abstractmethod
63     def startOpenGate(self) -> Future[bool]:
64         pass
65
66     @abstractmethod
67     def startOpenChest(self) -> Future[bool]:
68         pass
69
70     @abstractmethod
71     def endAllAction(self) -> Future[bool]:
72         pass
73
74     # 消息相关，接收消息时无消息则返回(-1, '')
75
76     @abstractmethod
77     def sendMessage(self, toID: int, message: str) -> Future[bool]:
78         pass
79
80     @abstractmethod
81     def haveMessage(self) -> bool:
82         pass
83
84     @abstractmethod
85     def getMessage(self) -> Tuple[int, str]:
86         pass
87
88     # 等待下一帧
89
90     @abstractmethod
91     def wait(self) -> Future[bool]:
92         pass
93
94     # 获取各类游戏中的消息
95
96     @abstractmethod
97     def getFrameCount(self) -> int:
98         pass
99
100    @abstractmethod
101    def getPlayerGUIDs(self) -> List[int]:
102        pass
103
104    @abstractmethod
105    def getTrickers(self) -> List[THUAI6.Tricker]:
106        pass
107
108    @abstractmethod
109    def getStudents(self) -> List[THUAI6.Student]:
110        pass
```

```

111
112     @abstractmethod
113     def GetProps(self) -> List[THUAI6.Prop]:
114         pass
115
116     @abstractmethod
117     def GetBullets(self) -> List[THUAI6.Bullet]:
118         pass
119
120     @abstractmethod
121     def GetSelfInfo(self) -> Union[THUAI6.Student, THUAI6.Tricker]:
122         pass
123
124     @abstractmethod
125     def GetFullMap(self) -> List[List[THUAI6.PlaceType]]:
126         pass
127
128     @abstractmethod
129     def GetPlaceType(self, cellx: int, celly: int) -> THUAI6.PlaceType:
130         pass
131
132     @abstractmethod
133     def IsDoorOpen(self, cellx: int, celly: int) -> bool:
134         pass
135
136     @abstractmethod
137     def GetChestProgress(self, cellx: int, celly: int) -> int:
138         pass
139
140     @abstractmethod
141     def GetGateProgress(self, cellx: int, celly: int) -> int:
142         pass
143
144     @abstractmethod
145     def GetClassroomProgress(self, cellx: int, celly: int) -> int:
146         pass
147
148     @abstractmethod
149     def GetDoorProgress(self, cellx: int, celly: int) -> int:
150         pass
151
152     @abstractmethod
153     def GetHiddenGateState(self, cellx: int, celly: int) ->
THUAI6.HiddenGateState:
154         pass
155
156     @abstractmethod
157     def GetGameInfo(self) -> THUAI6.GameInfo:
158         pass
159
160     # 用于DEBUG的输出函数，仅在DEBUG模式下有效
161

```

```

162     @abstractmethod
163     def Print(self, cont: str) -> None:
164         pass
165
166     @abstractmethod
167     def PrintStudent(self) -> None:
168         pass
169
170     @abstractmethod
171     def PrintTricker(self) -> None:
172         pass
173
174     @abstractmethod
175     def PrintProp(self) -> None:
176         pass
177
178     @abstractmethod
179     def PrintSelfInfo(self) -> None:
180         pass
181
182
183 class IStudentAPI(IAPI, metaclass=ABCMeta):
184
185     # 人类阵营的特殊函数
186
187     @abstractmethod
188     def Graduate(self) -> Future[bool]:
189         pass
190
191     @abstractmethod
192     def StartLearning(self) -> Future[bool]:
193         pass
194
195     @abstractmethod
196     def StartEncourageMate(self, mateID: int) -> Future[bool]:
197         pass
198
199     @abstractmethod
200     def StartRouseMate(self, mateID: int) -> Future[bool]:
201         pass
202
203     @abstractmethod
204     def GetSelfInfo(self) -> THUAI6.Student:
205         pass
206
207
208 class ITrickerAPI(IAPI, metaclass=ABCMeta):
209
210     # 屠夫阵营的特殊函数
211
212     @abstractmethod
213     def GetSelfInfo(self) -> THUAI6.Tricker:

```

