



مسابقات هوش مصنوعی چیلین وارز ChillinWars AI Competition

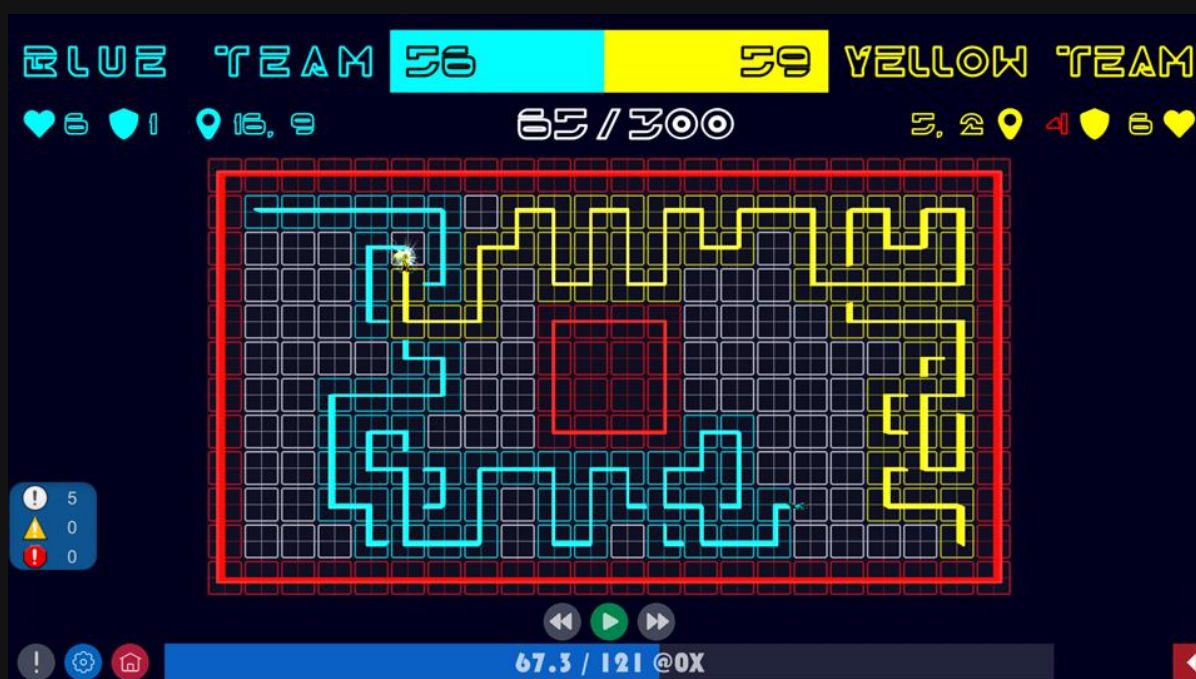
مستند بازی جونیور “Tron”

از طریق این داکيومنت با ساختار و شرح بازی از شروع تا پایان به طور کامل آشنا خواهید شد و همچنین تمام موارد فنی مورد نیاز شما (از جمله توابع و کلاس‌های پیاده‌سازی شده – اکشن‌های موجود در بازی و...) برای شرکت در این مسابقه در ادامه این مستند توضیح داده شده‌است.

زمستان ۱۳۹۸

شرح بازی:

بازی شامل دو تیم Blue و Yellow است که بر روی یک صفحه $n \times m$ انجام می‌شود. هر تیم دارای یک عامل (Agent) است که می‌تواند بر روی این صفحه حرکت کند. خانه‌های صفحه می‌توانند خالی یا دیوار باشند که دیوارها خود ۳ نوع‌اند؛ دیوارهای محیط (AreaWall) که با برخورد به آن‌ها می‌میرید، دیوارهای آبی (BlueWall) و دیوارهای زرد (YellowWall). هر عامل با حرکت بر روی صفحه در پشت سر خود دیواری از جنس خودش می‌سازد. تعداد دیوارهای هر تیم امتیاز مثبت آن تیم است. همچنین می‌توانید با برخورد با دیوارهای خود یا حریف آن‌ها را از بین ببرید که باعث کاهش امتیاز نیز می‌شود. هدف از بازی زنده ماندن بیشتر و تخریب دیوارهای حریف است.



شروع بازی:

در ابتدای بازی هر عامل در نقطه‌ای با جهت مشخص قرار داده شده‌است. دور تا دور نقشه نیز AreaWall است. همچنین در برخی نقاط داخل نقشه نیز ممکن است AreaWall وجود داشته باشد. نقشه‌ها همیشه قرینه خواهند بود و هیچ تیمی در ابتدا برتری‌ای نسبت به دیگری ندارد در نتیجه در هنگام پیاده سازی کد خود لازم نیست تیم (Blue یا Yellow بودن) را در نظر بگیرید.

اکشن‌های بازی:

- تغییر جهت (ChangeDirection):
 - عامل میتواند به چهار جهت بالا، راست، پایین و چپ تغییر جهت دهد.
 - تغییر جهت به خلاف جهتی که دارید ممکن نیست. (برای مثال وقتی به سمت راست حرکت می‌کنید نمیتوانید به چپ تغییر جهت دهید)
- فعال کردن قابلیت دیوار شکن (ActivateWallBreaker):
 - با فعال کردن این قابلیت می‌توانید دیوارهای BlueWall و YellowWall را بدون کم شدن جان (Health) بشکنید و تبدیل به خانه‌ی خالی (Empty) کنید.
 - با فعال کردن این قابلیت، برای مدت زمان معلومی این قابلیت فعال است. بعد از اتمام نیز باید مدتی صبر کنید تا دوباره بتوانید آن را فعال کنید.
 - این قابلیت تاثیری در برخورد با عامل (Agent) حریف یا دیوارهای محیط (AreaWall) ندارد.
- حرکت:
 - در هر سیکل در جهتی که دارید یک خانه به جلو می‌روید.
 - با هر حرکت، دیواری در خانه‌ی قبلی خود می‌سازید.
 - اگر خانه‌ی جدیدی که در آن قرار می‌گیرید از نوع AreaWall باشد، می‌میرید و امتیازتان به اندازه‌ی معلومی کم می‌شود.
 - اگر خانه‌ی جدید از نوع BlueWall یا YellowWall باشد:
- اگر قابلیت دیوار شکن فعال **باشد** یا تعداد جان‌های (Health) عامل **بیش از یک** باشد، دیوار شکسته می‌شود و به اندازه‌ی امتیاز ساخت دیوار، از تیمی که دیوار متعلق به آن بود کم می‌شود (برای مثال بدون توجه به اینکه شما کدام تیم هستید، در صورت برخورد با YellowWall امتیاز از تیم Yellow کم می‌شود). در صورت فعال نبودن دیوار شکن یکی از جان‌های عامل از بین می‌رود.
- در غیر اینصورت (اگر قابلیت دیوار شکن فعال **نباشد** و تعداد جان‌های عامل **یک** باشد)، تعداد جان‌های (Health) عامل صفر می‌شود و می‌میرد و با توجه به تیم دارنده‌ی دیوار (مانند حالت قبلی)، مقداری از امتیازش کم می‌شود.

نحوه امتیازدهی:

تعداد دیوارهای متعلق به هر تیم ضرب در عددی، امتیاز مثبت آن تیم است. همچنین در صورت مردن، با توجه به نوع مردن، امتیازی از تیم کم خواهد شد.

شرط پایان بازی:

بازی در یکی از شرایط زیر به اتمام می‌رسد:

- اگر تعداد سیکل‌ها به حداکثر تعداد خود برسد.
- اگر عامل‌ها به هم برخورد کنند.
- اگر یک تیم با AreaWall برخورد کند.
- اگر تعداد جان‌های یک تیم صفر شود.

نتیجه بازی:

- تیمی که در پایان بازی امتیاز بیشتری داشته باشد، برنده‌ی بازی است.



توابع و کلاس های بازی:

کد هوش مصنوعی خود را می بایست در تابع `decide` و در فایل `ai.*` قرار دهید. یک کد نمونه برای آشنایی بیشتر پیاده سازی شده که توصیه می شود در ابتدا آن کد را حتما کامل بخوانید و متوجه شوید.

توجه: با توجه به اینکه در زبان های مختلف `code-style` متفاوت است، در این مستند سعی می کنیم که از یک زبان واحد برای نشان دادن نام ها، ویژگی ها و موجودیت های بازی استفاده کنیم و سینتکس و اطلاعات دقیق تر در کد نمونه ای که قرار داده شده قابل مشاهده است.

در ادامه به تشریح ویژگی های (attributes) موجودیت های بازی می پردازیم.

- **AI:** در این کلاس دو تابع `initialize` و `decide` وجود دارند. تابع `initialize` فقط یکبار در سیکل اول (سیکل صفر) قبل از تابع `decide` اجرا می شود و می توان مقادیر اولیه را در آن مقداردهی کرد (استفاده از این تابع اختیاری است). تابع `decide` در هر سیکل بازی یک بار اجرا می شود و کد هوش مصنوعی در این تابع پیاده سازی خواهد شد. بازی به صورت `real-time` است یعنی در هر سیکل هر دو تیم قادر به انجام فعالیت های خود هستند.
- کلاس `AI` دارای ویژگی های زیر است:

Name	Type	Description
World	World	جهان بازی
my_side	string	نام تیم خودی ^[۱]
other_side	string	نام تیم حریف ^[۱]
current_cycle	int	سیکل فعلی بازی
cycle_duration	float	مدت زمان تصمیم گیری در هر سیکل به ثانیه

[۱] نام تیم یکی از مقادیر `Blue` یا `Yellow` است.



- **World:** کلاس جهان بازی است و تمام موجودیت‌های مربوط به بازی در این کلاس و ویژگی‌های آن قرار دارد.

Name	Type	Description
board	2D Array<ECell>	صفحه‌ی بازی ^[۲]
agents	map<string, Agent>	اطلاعات عامل هر تیم (نام تیم key و عامل value است)
scores	map<string, int>	امتیاز هر تیم (نام تیم key و امتیاز value است)
constants	Constants	مقادیر ثابت

[۲] صفحه‌ی بازی یک آرایه‌ی دوبعدی $n \times m$ از جنس ECell است.

[۲] مبدا مختصات در بالا و سمت چپ صفحه قرار دارد.

x=0	x=1	x=2	
y=0	y=0	y=0	
x=0	x=1	x=2	
y=1	y=1	y=1	→ board[1][2]
x=0	x=1	x=2	
y=2	y=2	y=2	

- **ECell:** enum که نوع خانه‌ی صفحه را نشان می‌دهد و دارای ویژگی‌های زیر است:

enum attribute	Description
Empty	سلول خالی
AreaWall	دیوار محیط
BlueWall	دیوار تیم آبی
YellowWall	دیوار تیم زرد

• Agent: کلاس عامل

Name	Type	Description
health	int	تعداد جان‌های باقی مانده ^[۳]
position	Position	مکان عامل
direction	EDirection	جهت حرکت عامل
wall_breaker_cooldown	int	تعداد سیکل باقی مانده تا امکان فعال کردن قابلیت دیوار شکن ^[۴]
wall_breaker_rem_time	int	تعداد سیکل باقی مانده از قابلیت دیوار شکن ^[۵]

[۳] هنگامی که تعداد جان‌های شما یک باشد، با برخورد به دیوار بدون داشتن دیوار شکن خواهید مرد.

[۴] هنگامی که مقدار این ویژگی برابر صفر باشد، می‌توانید قابلیت دیوار شکن را فعال کنید.

[۵] هنگامی که مقدار این ویژگی برابر یک است، در سیکل بعدی قابلیت دیوار شکن فعال نیست و با برخورد به دیوار از جان‌های شما کم خواهد شد.

• Position: کلاس مکان عامل که دارای دو ویژگی زیر است:

Name	Type	Description
x	int	طول مکان عامل
y	int	عرض مکان عامل

- **enum :EDirection**: ای با ویژگی‌های زیر است که بیانگر جهت حرکت عامل است:

enum attribute	Description
Up	جهت بالا
Right	جهت راست
Down	جهت پایین
Left	جهت چپ

- **Constants**: کلاس مقادیر ثابت که در طول اجرای بازی تغییر نمی‌کنند:

Name	Type	Description
max_cycles	int	حداکثر تعداد سیکل‌های بازی
init_health	int	تعداد جان‌ها در شروع بازی
wall_breaker_cooldown	int	تعداد سیکل‌های مورد نیاز برای فعال کردن دوباره دیوار شکن
wall_breaker_duration	int	تعداد سیکل‌های فعال بودن قابلیت دیوار شکن
wall_score_coefficient	int	میزان امتیاز به ازای هر دیوار متعلق به تیم
area_wall_crash_score	int	میزان امتیازی که در صورت مردن بر اثر برخورد با AreaWall کم می‌شود
my_wall_crash_score	int	میزان امتیازی که در صورت مردن بر اثر برخورد با دیوارهای خودی کم می‌شود
enemy_wall_crash_score	int	میزان امتیازی که در صورت مردن بر اثر برخورد با دیوارهای حریف کم می‌شود

مدت زمان اجرای کد:

مدت زمان اجرای سیکل اول بازی (سیکل صفر) ۳ ثانیه است و توابع initialize و decide اجرا خواهند شد. این سیکل با توجه به طولانی‌تر بودن، برای پیش پردازش‌های سنگین و ذخیره نتایج برای استفاده در سیکل‌های بعدی مناسب است.

مدت زمان اجرای سایر سیکل‌ها (از سیکل یک به بعد) ۰.۵ ثانیه است و فقط تابع decide در این سیکل‌ها اجرا می‌شود.

ترتیب اتفاقات در بازی:

۱. اجرای کامندهای ارسال شده در صورت صحیح بودن
۲. بررسی میزان باقی مانده از قابلیت دیوار شکن و cooldown آن
۳. ساخت دیوار در مکان فعلی عامل‌ها
۴. حرکت رو به جلوی عامل‌ها با توجه به جهت
۵. بررسی برخورد با عامل حریف
۶. بررسی برخورد با دیوار محیط
۷. بررسی برخورد با دیوار خودی یا حریف و شکستن آن‌ها یا مردن با توجه به شرایط گفته شده

عوض کردن نقشه

به صفحه کانفیگ پروژه در آدرس `PythonServer/gamecfg.json` بروید و نام نقشه مورد نظر را در محل نشان داده شده وارد کنید:

```
39 → "game_handler": {
40 →   →   "": "base game handler configs",
41 →   →   "start_time": "2017-11-08 00:50",
42 →   →   "start_waiting_time": 3,
43 →   →   "": "realtime game handler configs",
44 →   →   "cycle_duration": 0.5,
45 →   →   "max_total_commands": 200,
46 →   →   "max_team_commands": 100,
47 →   →   "max_agent_commands": 100,
48 →   →   "": "my configs",
49 →   →   "map_path": "maps/Test.json"
50 →   }
51 }
```

****** مقادیر اولیه‌ی ویژگی‌های گفته شده (مانند مقادیر ثابت، نقطه و جهت شروع عامل‌ها) در فایل نقشه مشخص شده‌اند.



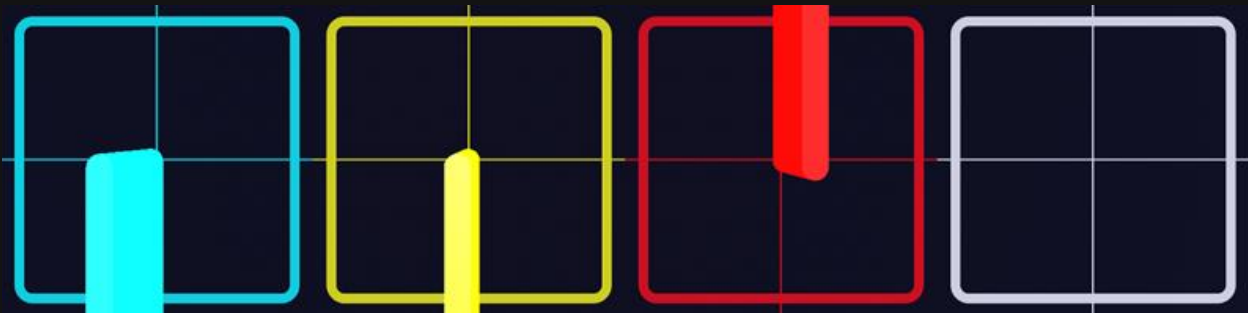
راهنمای مانیتور



از بالا چپ:

- نام تیم
- امتیاز تیم
- علامت قلب: تعداد جان‌های باقی مانده
- علامت سپر: وضعیت دیوار شکن را نمایش می‌دهد. اگر این عدد قرمز باشد (مانند قسمت مشخص شده برای تیم زرد) نمایانگر `wall_breaker_rem_time` است و اگر هم‌رنگ تیم باشد (مانند قسمت مشخص شده برای تیم آبی) نمایانگر `wall_breaker_cooldown` است.
- علامت موقعیت: مکان عامل به صورت x, y
- سیکل فعلی بازی
- حداکثر تعداد سیکل‌های بازی

خانه‌های صفحه (ECell):



از راست به چپ: Empty, AreaWall, YellowWall, BlueWall