



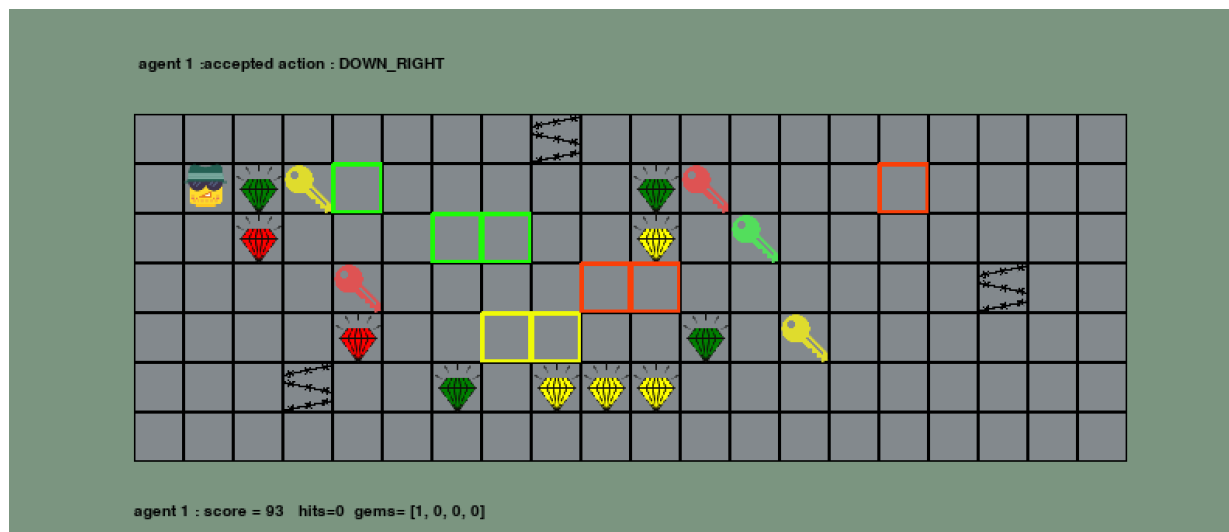
مقدمه

پروژه‌ی درس به صورت یک بازی است. دانشجویان باید دست به شناسایی و کاوش محیط بزنند و با بررسی مشخصات و ویژگی‌های آن، یک عامل هوشمند پیاده سازی کنند. در این بازی معیار score به معنای امتیاز و میزان سلامتی عامل برای ارزیابی عملکرد هوش مصنوعی قرار دارد.

توضیح محیط بازی

محیط گرافیکی بازی

در این بازی، یک عامل وجود دارد که در محیط بازی، اقدام به جمع آوری الماس می‌کند. شکل ۱ نمونه‌ای از حالت نمایشی این بازی می‌باشد.



شکل ۱: نمایش گرافیکی از محیط بازی

موجودیت‌های بازی

انواع مختلف موجودیت در این بازی به شرح زیر است:

- عامل (agent): همانطور که در شکل ۲ آمده است. عامل در محیط بازی حرکت کرده و اقدام به جمع آوری الماس‌ها می‌کند.



شکل ۲: شکل ظاهری عامل.

- دیوار (wall): همانطور که در شکل ۳ آمده است، در محیط بازی برخی از خانه‌ها دارای دیوار هستند و به معنای آن است که عامل نمی‌تواند در آن خانه ها وارد شود.



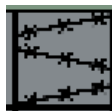
شکل ۳: دیوار

- الماس: همانطور که در شکل ۴ آمده است، بازی دارای ۴ نوع الماس (به ترتیب زرد، سبز، قرمز، آبی) می‌باشد. هر الماس دارای امتیاز مخصوص به خود می‌باشد. عامل با خوردن دنباله‌ای از الماس‌ها، امتیازات خود را افزایش می‌دهد.



شکل ۴: ۴ نوع الماس که هر کدام امتیاز مخصوص به خود را دارند.

- سیم خاردار (Barbed): همانطور که در شکل ۵ آمده است، در محیط بازی، برخی از خانه‌ها دارای سیم‌خاردار هستند و به معنای آن است که عامل نمی‌تواند وارد این خانه‌ها شود ولی مقداری امتیاز از دست می‌دهد.



شکل ۵: خانه‌های سیم‌خاردار که عبور از آن‌ها با جریمه همراه است.

- خانه‌ی درب‌دار (Door): همانطور که در شکل ۶ آمده است، در محیط بازی برخی از خانه‌ها دارای درب‌های رنگی هستند و به معنای آن است که عامل نمی‌تواند در آن خانه ها وارد شود مگر آن‌که کلید مربوط آن خانه (متناسب با رنگ درب) را داشته باشد.



شکل ۶: سه نوع خانه‌ی درب‌دار که با رنگ‌های سبز، قرمز و زرد مشخص شده است. شرط ورود عامل به این خانه‌ها داشتن کلید با رنگ متناظر آن می‌باشد.

- کلید (key): همانطور که در شکل ۷ آمده است، در محیط بازی برخی از خانه‌ها دارای کلیدهای رنگی هستند. عامل با خوردن هر کلید رنگی، از آن پس می‌تواند در خانه‌های درب‌دار متناظر با آن رنگ ورود کند.



شکل ۷: سه نوع کلید با رنگ‌های سبز، قرمز و سبز مشخص شده است. عامل با داشتن هر کدام از این سه کلید می‌تواند وارد خانه‌های درب‌دار با رنگ متناظر شود.

کنش‌های بازی (actions)

هر عامل در نوبت خود، کنشی انتخاب می‌کند، و متناسب با شرایط بازی کنش‌ها اعمال می‌شود.

- حرکت به سمت بالا (up)
- حرکت به سمت پایین (down)
- حرکت به سمت چپ (left)

- حرکت به سمت راست (right)
- حرکت به سمت بالا-راست (up-right)
- حرکت به سمت بالا-چپ (up-left)
- حرکت به سمت پایین-راست (down-right)
- حرکت به سمت پایین-چپ (down-left)
- بدون حرکت (noop)

عامل با اعمال کنش‌های حرکت، به آن خانه برود. اگر در خانه‌ی مقصد دیوار یا درب (در صورت نداشتن کلید) وجود داشت، عامل نمی‌تواند در آن خانه برود. اگر عامل کنش بدون حرکت (noop) را انتخاب کند، عامل در سرجای خود می‌ایستد.

قوانین بازی

قوانین بازی به صورت زیر است:

- در ابتدای بازی، بازیکن در خانه‌ی گوشه سمت چپ قرار می‌گیرد.
- در ابتدای بازی اطلاعات اولیه‌ی بازی، در اختیار بازیکن قرار می‌گیرد که به صورت زیر است:
 - طول و عرض زمین بازی
 - کاراکتر مربوط به بازیکن
 - کاراکتر A برای بازیکن (به دلیل وجود عامل‌های دیگر در فازهای بعدی این کاراکتر قرار داده شده است. در این فاز همیشه عامل کاراکتر A است.)
 - Id مربوط به بازیکن
 - عدد ۱ (به دلیل وجود عامل‌های دیگر در فازهای بعدی این Id قرار داده شده است. در این فاز همیشه عامل id ۱ را دارد.)
 - امتیاز اولیه‌ی بازیکن
 - حداکثر تعداد دور بازی
 - تعداد عامل‌های بازی (در این فاز برابر با ۱ است.)
- در هر نوبت از بازی، به بازیکنی که نوبت اوست اطلاعاتی داده می‌شود و از او کنشی خواسته می‌شود (چگونگی جزئیات دریافت و ارسال داده‌ها در ادامه ذکر خواهد شد.)، که به صورت زیر است:
 - شماره‌ی دور بازی
 - این عدد از ۱ شروع شده و یکی یکی اضافه می‌شود.
 - امتیاز هر کدام از بازیکنان به ترتیب
 - نقشه‌ی بازی در نوبت مورد نظر
- اگر بازیکنی در خانه‌ای قرار گیرد که دارای الماس است، الماس برداشته می‌شود و به اندازه‌ی مربوطه به امتیازات او افزوده می‌شود.
 - امتیاز کسب شده برای هر دنباله‌ی دوتایی از الماس‌ها در جدول شماره‌ی ۲ آمده است.
 - اولین الماس امتیاز خود را دارا است.
 - خانه‌ی دارای الماس، بعد از برداشت الماس تبدیل به یک خانه‌ی خالی می‌شود.
 - برای مثال ترتیب الماس زرد، سبز، آبی، زرد دارای ۴ امتیاز، هیچ-زرد، زرد-سبز، سبز-آبی، آبی-زرد می‌باشد.
- در هر نوبت از بازی، متناسب با کنش درخواستی عامل (چه انجام شود چه نشود.) مقداری امتیاز از عامل کم می‌شود.

- مقدار امتیاز کسر شده برای هر کنش در جدول ۱ آمده است.
- اگر بازیکنی در خانه‌ای که سیم خاردار قرار دارد، به اندازه‌ی هر دور از بازی که در آن خانه است، به اندازه‌ی مربوطه امتیاز کسر می‌شود.
- مقدار امتیاز کسر شده برای حضور در خانه‌ی سیم خاردار حریف در جدول ۱ آمده است.
- در صورت کنش اشتباه و غیر منطقی از هر بازیکن، عامل در جای خود می‌ماند.

مقدار	مولفه‌های بازی
۱	میزان کسر امتیاز برای حرکات مستقیم (بالا، پایین، چپ، راست)
۲	میزان کسر امتیاز برای حرکات مورب (بالا-راست، بالا-چپ، پایین-راست، پایین-چپ)
۰	میزان کسر امتیاز برای حرکت NOOP
۲۰	میزان کسر امتیاز برای قرار گرفتن در خانه‌های سیم خاردار (به ازای هر دور از بازی)

جدول 1: مقدار مولفه‌های بازی

الماس آبی	الماس قرمز	الماس سبز	الماس زرد	الماس دوم ----- الماس اول
۰	۰	۰	۵۰	هیچکدام
۰	۱۰۰	۲۰۰	۵۰	الماس زرد
۱۰۰	۲۰۰	۵۰	۱۰۰	الماس سبز
۲۰۰	۵۰	۱۰۰	۵۰	الماس قرمز
۵۰	۱۰۰	۵۰	۲۵۰	الماس آبی

شکل ۲: جدول امتیازات اضافی به ازای خوردن ترتیب هر دوتایی از الماس‌ها. الماس اول در سطر و الماس دوم در ستون قرار دارد.

توضیحات اجرای بازی

کدهای مربوط به بازی در گیت با آدرس https://github.com/amirsed76/AI_Game_2021 موجود است.

اجرای سرور بازی

نحوه‌ی چگونگی اجرای بازی، در README واقع github موجود است. برای اجرای سرور، سیستم نیازمند پایتون است.

نحوه‌ی کدنویسی برای کلاینت بازی (عامل بازی)

برای زبان‌هایی همچون پایتون و جاوا، کدهای کلاینتی نوشته شده است که افراد درگیر اتصالات و رفت و آمد داده‌ها در سوکت نشوند. افراد باید در فایل‌های مربوطه، در تابع `do_turn` کنش مربوطه را با توجه به اطلاعات موجود در AI، انتخاب کنند.

توضیحات بیشتر اطلاعات، در قالب یک فیلم در اختیار افراد قرار می‌گیرد.

همچنین اطلاعات مربوط به نقشه‌ی فعلی بازی که در قسمت‌های قبل به آن اشاره شد، این بخش شامل یک آرایه‌ی دو بعدی است (که در کلاینت به متغیر مربوط به آن قرار دارد.) که در خانه‌ی `i,j` کاراکترهایی موجود است که کاراکتر اول وضعیت حالت هر خانه را نشان می‌دهد که به صورت زیر است. (دقت شود در هر خانه از نقشه، یک حالت از شرایط زیر رخ می‌دهد.)

○ E: خانه‌ی خالی

○ W: دیوار

- 1: الماس زرد
- 2: الماس سبز
- 3: الماس قرمز
- 4: الماس آبی
- G: خانه‌ی درب‌دار سبز
- R: خانه‌ی درب‌دار قرمز
- Y: خانه‌ی درب‌دار زرد
- g: کلید سبز
- r: کلید قرمز
- y: کلید زرد
- * (ستاره): سیم خاردار

اگر کاراکتر A (در این فاز) در یک خانه از آرایه موجود باشد به معنای آن است که عامل در آن خانه قرار دارد.

همچنین روند هر دور از بازی می‌توانید در فایل outs.txt واقع در server مشاهده کرد. برای مثال دور ششم از یک بازی به صورت زیر است:

```
-----
TURN 5
E E E E E E E * E E E E E E E E E E
E EA 2 y G E E E E E 2 r E E E R E E E E
E E 3 E E E G G E E 1 E g E E G R Y E E E
E E E E r E E E E R R E E E E E E * E E E
E E E E 3 E E Y Y E E 2 E y E E E E E E E
E E E * E E 2 E 1 1 1 E E E E E E E E E E
E E E E E E E E E E E g r y E E E E E E
```

نمایش گرافیکی بازی

بعد از ایجاد لاگ بازی، شما می‌توانید با اجرای بخش visualizer، محیط گرافیکی و رویدادهای واقع در بازی را مشاهده کنید. (توضیحات مربوط به بخش visualizer در بخش README واقع در github موجود می‌باشد.) در محیط گرافیکی بازی، شما می‌توانید با استفاده از دکمه‌ی e، سرعت نمایش را زیاد کنید، با استفاده از دکمه‌ی w، سرعت را کم و دکمه‌ی s، بازی را متوقف کنید.

ساخت نقشه

شما می‌توانید، نقشه‌های مورد نظر خود را بسازید. فایل txt با قرار دادن کاراکترهای مورد نظر ساختار نقشه‌ی خود را بسازید.

نکات ساخت نقشه به شرح زیر است:

- فایل txt باید در فولدر maps قرار گیرد.
 - هر سطر از محیط بازی، یک خط از فایل text را در بر می‌گیرد.
 - باید تعداد کاراکترهای هر سطر با هم برابر باشند.
 - در هر سطر برای هر المان از محیط یک کاراکتر در نظر گرفته شده است.
- E: خانه‌ی خالی
 - W: دیوار
 - 1: الماس زرد
 - 2: الماس سبز
 - 3: الماس قرمز

- 4: الماس آبی
- G: خانه‌ی درب‌دار سبز
- R: خانه‌ی درب‌دار قرمز
- Y: خانه‌ی درب‌دار زرد
- g: کلید سبز
- r: کلید قرمز
- y: کلید زرد
- * (ستاره): سیم خاردار

تغییر تنظیمات بازی

با تغییر فایل config.json واقع در فولدر server، می‌توان تنظیمات بازی را عوض کرد. المان‌های موجود در این فایل به شرح زیر است:

- map: اسم نقشه‌ی مورد نظر در فولدر server/maps
 - server_ip و server_port: آدرس کامپیوتری که به عنوان server در نظر گرفته می‌شود.
 - player_count: تعداد بازیکنان که می‌تواند ۱ یا ۲ باشد. (در این فاز فقط عدد ۱ باید قرار گیرد.)
 - time_out: حداکثر مقدار زمانی که یک عامل می‌تواند در هر دور پاسخ بدهد. (این مقدار در ارزیابی برابر ۱ ثانیه است.)
 - Init_score: مقدار اولیه‌ی امتیاز هر عامل
 - Init_random_location: اگر این مقدار برابر true باشد، عامل‌ها به طور رندوم از چهار گوشه‌ی زمین شروع به بازی می‌کنند و در صورت false بودن آن، عامل اول از گوشه‌ی چپ بالا و عامل دوم از گوشه‌ی راست پایین شروع به حرکت می‌کنند.
- نمونه‌ای از یک فایل config به صورت زیر است.

```
{
  "map": "map1_11.txt",
  "server_ip": "127.0.0.1",
  "server_port": 9921,
  "player_count": 2,
  "time_out": 1,
  "init_score": 45,
  "trap_count": 3,
  "turn_count": 100,
  "init_random_location": false
}
```