



تمرین محاسباتی چهارم (پروژه نهایی)

دانشکده مهندسی برق

علوم اعصاب یادگیری، حافظه، شناخت

استاد درس: دکتر کربلایی آقاجان

نیمسال اول سال ۱۴۰۳-۱۴۰۴

آخرین مهلت تحویل: ۱۰ بهمن ۱۴۰۳ (بدون تأخیر مجاز)

نکات و ضوابط مرتبط با تمرین محاسباتی چهارم

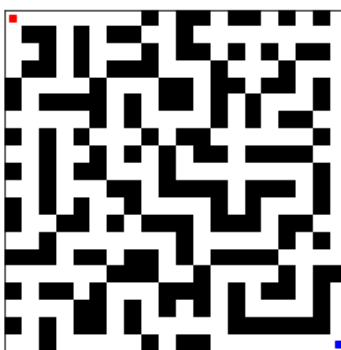
- در این سری از تمرینات به دلیل امتیازی بودن نمره آن، برخلاف رویه تمرینات گذشته، تمرینات تنها به صورت فردی قابل انجام است و انجام گروهی آن تقلب محسوب می‌شود.
- تأکید می‌شود که کیفیت گزارش تمرینات، به اندازه خروجی نهایی تمرین‌ها اهمیت داشته و درصد مهمی از بارم تمارین را شامل می‌شود. بنابراین لازم است که استدلال‌ها و دلیل استفاده از دستورات و نتایج را به صورت دقیق در گزارش خود شرح دهید.
- بخش‌هایی از این سری تمرین که نمره امتیازی برای آن‌ها مشخص شده است، در نمره همین تمرین لحاظ خواهد شد.
- نمره حاصل از این تمرین، در جهت جبران نمرات تمرینات پیشین بوده و در نمره امتحانات میان‌ترم و پایان‌ترم تأثیر نخواهد داشت.
- تمرین خود را در قالب فایل ZIP یا RAR شامل فایل PDF گزارش و تمامی کدهای مربوطه را با ذکر نام و نام خانوادگی خود به همراه شماره دانشجویی در فایل ارسالی، بخش تعیین شده در سامانه درس ارسال کند.
- استفاده از ابزارهای هوش مصنوعی در کدنویسی تمرینات عملی بلامانع است اما در گزارش نویسی و تفسیر نتایج و توضیحات مربوط به کدها مجاز نمی‌باشد.
- با توجه به حجم تمرینات، توصیه می‌شود که انجام آن‌ها را به روزهای آخر موکول نکنید.
- در صورت وجود هرگونه ابهام یا سوال از تمرینات، از طریق گروه درس یا [ایمیل](#) با دستیاران آموزشی درس مطرح کنید.

معرفی: در این پروژه قرار است به حل یک هزارتو (Maze) بپردازیم. و برای این کار به سراغ الگوریتم یادگیری تقویتی (Reinforcement Learning) خواهیم رفت.

گام‌های پروژه: (مجموع امتیاز:)

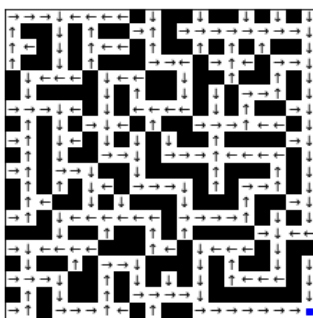
ساخت یک maze به کمک فایل پایتون موجود در پروژه. دقت کنید که این کد به صورت رندوم هر بار یک maze جدید تولید می‌کند اما بعد از اجرا آن را در قالب یک numpy array در همان پوشه ذخیره می‌کند. و برای تکرار آزمایش‌های خود می‌توانید از آن استفاده کنید. بدیهی است تکراری بودن maze هر دو پروژه تحویلی به منزله تقلب خواهد بود و می‌توانید از یک seed متناسب با شماره دانشجویی خود استفاده کنید. ابعاد نیز همان ۲۰ در ۲۰ خواهد بود. (به عنوان مثال: ۲ یا ۳ رقم آخر) (۱۰ نمره)

- تضمین می‌شود از نقطه شروع agent یعنی بالاترین نقطه سمت راست به نقطه هدف یعنی پایین ترین نقطه سمت چپ مسیری وجود دارد. با خواندن کد دلیل این گزاره را شرح دهید. (۵ نمره)
- پس از تولید و ذخیره maze مختص خود، اقدام به رسم آن کنید. (۵ نمره)



- نمایش‌های خلاقانه‌تر نیز شامل امتیاز اضافه خواهد شد. (تا ۵ نمره اضافه)
- به کمک الگوریتم Q-Learning اقدام به حل مسئله خود کنید. پس از یافتن پاسخ آن را به صورت یک تصویر بهترین policy بر روی maze خود رسم کنید. (۲۵ نمره)

- کد خوانا و تمیز و استفاده مناسب از شی گرایي امتیاز اضافه خواهد داشت. (تا ۱۰ نمره اضافه)



- نمایش‌های خلاقانه‌تر نیز شامل امتیاز اضافه خواهد شد. (تا ۵ نمره اضافه)
- برای توضیحات کامل پیاده‌سازی الگوریتم Q-Learning می‌توانید به [اینجا](#) مراجعه کنید.

راهنمایی: پس از مطالعه لینک بالا و آشنایی با کلیت الگوریتم به کمک مفاهیم شی گرای یک شی به اسم محیط enviroment و یک شی دیگر به اسم agent بسازید و در مراحل مختلف الگوریتم از مزایای شی گرای مانند encapsulation و غیره استفاده کنید. (البته کاملاً اختیاری است.)

- نحوه عملکرد کد خود به صورت بخش به بخش و با جزئیات کامل توضیح دهید. (۱۵ نمره)
- در این الگوریتم برای یافتن بهترین پاسخ، عامل شما بارها و بارها هر خانه را بررسی خواهد کرد. نمایش تمامی iteration ها در قالب یک فایل تصویری با فرمت gif یا یک ویدیو که درواقع حاصل به هم چسباندن policy های متفاوت تا رسیدن به بهترین پاسخ است. (۱۵ نمره) (تا ۵ نمره اضافه)
- (امتیازی) در مورد Deep Q-learning تحقیق کرده و تا جای امکان برداشت خود را با کتابخانه PyTorch پیاده سازی کنید. (تا ۳۰ نمره اضافه)
- (امتیازی) بررسی نتایج خود بر روی یک میز با ابعاد (۱۰۰ در ۱۰۰) و مقایسه سرعت عملکرد دو الگوریتم (تا ۲۰ نمره اضافه)

توجه: در پیاده سازی های خود امکان استفاده از GPT را دارید اما در توضیحات هرگونه استفاده از ابزارهای generative منجر به از دست دادن کل نمره آن بخش خواهد شد.