

## علوم اعصاب یادگیری، حافظه، شناخت (25645-1)



پژوهشنهایی

ترم بهار 1403-04

دانشکده مهندسی برق

دانشگاه صنعتی شریف

استاد: دکتر حمید کربلائی آقاجان

### نکات مهم در مورد تمرین

- در این تمرین، قسمت عمده‌ی نمره به گزارش تعلق می‌گیرد، لذا مطابق توضیحات گزارش خود را بنویسید.
- از آوردن کدها در گزارش خود خودداری کنید، این کار موجب کسر نمره خواهد شد، علاوه بر این از اسکرین شات گرفتن از خروجی کدهای خود نیز پرهیز کنید. (مگر برای نمودارها که تا حد امکان باید مرتب و تمیز باشند)
- در صورتی که تمرینات عملی را به صورت گروهی انجام می‌دهید، نام هم گروهی خود را حتما در گزارش تمرین ذکر کنید. علاوه بر این، هر دو عضو گروه باید تمرین را به صورت جداگانه در سامانه ارسال کنند. عدم رعایت این نکته منجر به از دست دادن نمره تمرین برای هر دو عضو گروه می‌شود. همچنین استفاده از ابزارهای هوش مصنوعی در تمرینات عملی بلامانع است.
- هر فرد یا گروه برای مجموعه تمرین‌های عملی تا سقف ۱۵ روز تاخیر بدون کسر نمره مجاز است. در صورت اتمام این مهلت مجاز، برای هر روز تاخیر در تحويل تمرین‌ها 30 درصد از نمره آن تمرین کسر خواهد شد.
- در صورت وجود هرگونه ابهام یا سوال از تمرینات، از طریق تلگرام به دستیاران آموزشی درس مراجعه کنید. ([محمد جلال نعمت‌بخش](#))

این تکلیف شامل مراحل اجرای الگوریتم Q-Learning در محیط **Frozen Lake** با استفاده از کتابخانه Gymnasium است و با هدف درک بهتر تصمیم‌گیری تحت عدم قطعیت طراحی شده است.

## شرح کلی تکلیف (۱۰۵ امتیاز)

این تکلیف طراحی شده تا دانشجویان، مفاهیم یادگیری تقویتی (Reinforcement Learning) را با استفاده از الگوریتم Q-Learning در محیط Frozen Lake پیاده‌سازی کنند. شما باید با نوشتن کد پایتون، عاملی را آموزش دهید تا در یک شبکه‌ی  $n^*$  از نقطه شروع به نقطه هدف برسد، بدون آنکه در چاله‌ها بیفتد. همچنین در پایان باید عملکرد عامل را تحلیل و بررسی کنید.

---

## مراحل انجام تکلیف

### 1. نصب کتابخانه Gymnasium

با استفاده از دستور زیر در محیط پایتون، کتابخانه را نصب کنید:

```
pip install gymnasium
```

### 2. ایجاد محیط و مقداردهی اولیه Q-Table

محیط Frozen Lake را ایجاد کنید.

یک Q-Table با ابعاد (تعداد حالت‌ها × تعداد اعمال) ایجاد و مقداردهی اولیه کنید.

### 3. پیاده‌سازی الگوریتم Q-Learning

الگوریتم Q-Learning را با استفاده از سیاست **epsilon-greedy** پیاده‌سازی کنید تا عامل بتواند از طریق آزمون و خطا سیاست بهینه را بیاموزد.

### 4. ارزیابی عملکرد عامل

پس از آموزش عامل، عملکرد آن را بر اساس میزان موفقیت (درصد رسیدن به خانه‌ی هدف) بررسی و محاسبه کنید.

## ۵. تحلیل نتایج

به پرسش‌هایی درباره سیاست بهینه، تأثیر پارامترها (مانند نرخ بادگیری، تخفیف، اپسیلون) و تفاوت بین محیط‌های قطعی و تصادفی پاسخ دهید.

---

## خروجی مورد انتظار

- کد کامل که قابلیت اجرای مجدد و بازتولید نتایج را داشته باشد. در این بخش امکان استفاده از دستیارهای هوشمند را دارید.<sup>(۳۰)</sup>
  - یک فایل گزارش از روند پیشرفت پروژه به همراه کتابخانه استفاده شده gym و محیط آن<sup>(۴۵)</sup>
  - یک گیف نهایی از انیمیشن ساخته شده با render mode روی human یا ansi، انجام هر دو ۱۰ امتیاز مثبت<sup>(۲۵)</sup>
- 

## اطلاعات پس زمینه

### محیط :Frozen Lake

- ابعاد شبکه:  $n * n$

- فضای اعمال (Action Space):

Left(): چپ

Down(): پایین

Right(): راست

Up(): بالا

## بخش‌های تکمیلی: (امتیاز اضافه هر تسک ۱۰ امتیاز)

- مقادیر مختلفی از پاداش رو قرار داده و نتایج را با هم از نظر زمان اتمام و امتیاز مقایسه کنید. به همراه فیلم(GIF)

- برای کد truncate در نظر بگیرید. (اتمام با رسیدن به شرطی معین) به همراه فیلم(GIF)  
- محیط را به صورت تصادفی در آورده و این‌بار نیز نتایج را با حالت اولیه مقایسه کنید. به همراه فیلم(GIF)