



هوش مصنوعی

بهار ۱۴۰۳

استاد: محمدحسین رهبان

گردآورندگان: پویا نوید، امیررضا آذری، پیام تائبی

مهلت ارسال: ۲۶ اسفند

جست و جو

تمرین اول

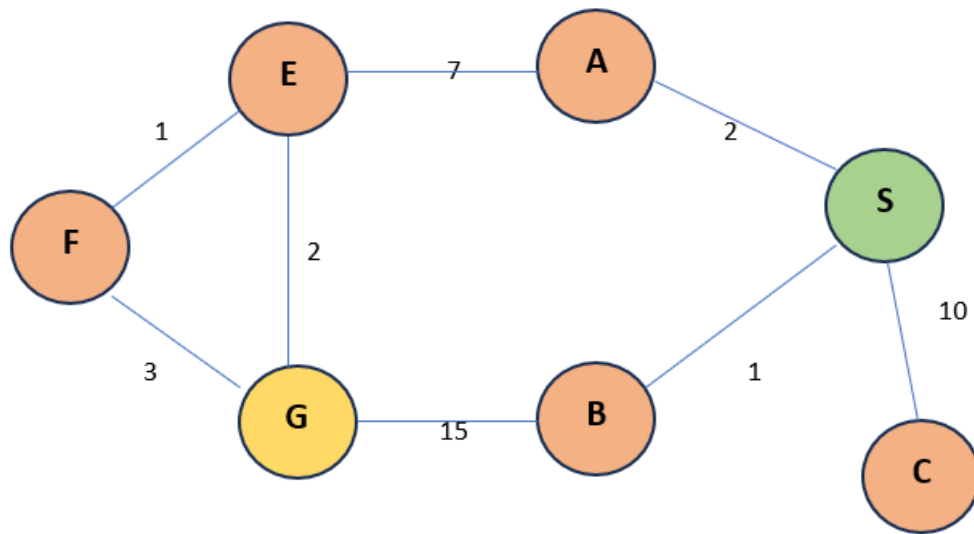
- مهلت ارسال پاسخ تا ساعت ۲۳:۵۹ روز مشخص شده است.
- در طول ترم امکان ارسال با تاخیر پاسخ هر تمرین تا سقف ۴ روز و در مجموع ۱۰ روز، وجود دارد. پس از گذشت این مدت، پاسخ‌های ارسال شده پذیرفته نخواهند بود. همچنین، به ازای هر ساعت تأخیر غیر مجاز ۰.۵ درصد از نمره تمرین به صورت ساعتی کسر خواهد شد.
- تاخیر سوالات نظری و عملی با یکدیگر محاسبه می‌شوند. به عبارتی تاخیر شما در هر تمرین معادل تاخیر بیشتر بین ارسال جواب‌های تئوری و عملی است.
- هم‌کاری و هم‌فکری شما در انجام تمرین مانعی ندارد اما پاسخ‌های ارسال شده هر کس حتماً باید توسط خود او نوشته شده باشد.
- در صورت هم‌فکری و یا استفاده از هر منابع خارج درسی، نام هم‌فکران و آدرس منابع مورد استفاده برای حل سوال مورد نظر را ذکر کنید.
- لطفاً تصویری واضح از پاسخ سوالات نظری بارگذاری کنید. در غیر این صورت پاسخ شما تصحیح نخواهد شد.
- در کنار هر سوال عددی به عنوان درجه سختی قرار گرفته است. درجه سختی برای مقایسه میزان سختی و وقت‌گیری سوالات و برنامه ریزی بهتر شما برای حل سوالات قرار گرفته است. هر درجه تقریباً معادل ۵ دقیقه وقت برای حل است. البته این اعداد به هیچ وجه دقیق نیست چرا که سرعت حل افراد متفاوت است، اما می‌توانید فرض کنید که اگر سرعت عملی مشابه با درجه سختی‌های داده شده دارید، با اطمینان بالایی در امتحانات به مشکل نخواهید خورد.

سوالات نظری (۹۰ نمره)

۱. (۲۰ نمره، درجه سختی ۸) درستی یا نادرستی عبارتهای زیر را با ذکر دلیلی مختصر یا بیان مثال نقض مشخص کنید:
 - (آ) در یک فضای متناهی، درخت جستجو می‌تواند نامتناهی باشد.
 - (ب) اگر f و g دو تابع قابل قبول^۱ باشند، $0.7f + 0.2g$ نیز حتماً قابل قبول است.
 - (ج) در روش $local\ beam\ search$ بیشترین k فرزند، به انتخاب k فرزند تصادفی ارجحیت دارد و صرفاً برای انتخاب سریع‌تر ممکن است از روش تصادفی استفاده شود.
 - (د) فضای حالتی وجود دارد که در آن $iterative\ deepening\ search$ بدتر از روش $depth\text{-}first\ search$ عمل کند. (مثلاً پیچیدگی زمانی $O(n^2)$ به جای $O(n)$)
 - (ه) برای اینکه الگوریتم BFS کامل باشد، لازم است که درجه انشعاب متناهی باشد اما برای کامل بودن الگوریتم IDS این شرط لازم نیست.
 - (و) جستجوی عمق اول، حالتی از Best-First می‌باشد. (یعنی تابع هزینه‌ای وجود دارد که الگوریتم Best-First را معادل DFS کند.)
 - (ز) فرض کنید یک عامل^۲ هوشمند برای بازی Tetris طراحی کرده‌ایم. محیط فعالیت^۳ برای این عامل، هرکدام از ویژگی‌های ذکر شده را بررسی کنید.

^۱ Admissible^۲ Agent^۳ Task environment

۲. (۱۰ نمره، درجه سختی ۳) فرض کنید شکل زیر یک فضای جست‌وجو بوده و وضعیت شروع حالت S و وضعیت پایان G باشد. اعداد نوشته شده روی یال‌ها هزینه مسیر هستند. برای هر کدام از الگوریتم‌های زیر مشخص نمایید چه مسیری طی می‌شود.



G	F	E	C	B	A	S	States
۰	۱	۱	۷	۱۰	۷	۹	$h(s)$

- DFS •
- greedy •
- UCS •
- A* •
- BFS •

۳. (۱۵ نمره، درجه سختی ۴) الگوریتم ژنتیک را در نظر بگیرید که در آن از کروموزوم‌هایی با طول ثابت ۸ و به فرم $x = \overline{ABCDEFGH}$ استفاده می‌کنیم. هر ژن می‌تواند عددی بین ۰ تا ۹ را بگیرد. تابع fitness را برای هر کروموزوم x به صورت زیر تعریف می‌کنیم:

$$f(x) = 3 * A + 2 * B + C - D - E + F + 2 * G + 3 * H$$

جمعیت اولیه از ۴ کروموزوم زیر تشکیل شده است:

$$x_1 = 21832965$$

$$x_2 = 19476320$$

$$x_3 = 89234548$$

$$x_4 = 37910977$$

آ) مقدار fitness نمونه‌های داده‌شده را به دست آورید.

ب) حال عملیات crossover را بر روی دو نمونه با بیشترین fitness انجام دهید، به صورت تک نقطه‌ای که محل crossover نقطه میانی کروموزوم باشد. همچنین عملیات crossover را به صورت دو نقطه‌ای روی دومین و سومین کروموزوم از لحاظ بیشتر بودن fitness انجام دهید. (روش این نوع crossover به این صورت است که CDEF از یک کروموزوم و بقیه ژن‌ها از یک کروموزوم دیگر بدست می‌آیند.)

ج) فرض کنید دستگاهی داریم که می‌تواند یک عدد طبیعی دو رقمی از ۱۰ تا ۸۹ را به صورت تصادفی خروجی بدهد. چهار بار از دستگاه استفاده کردیم و به اعداد زیر رسیدیم:

۷۲, ۴۷, ۱۵, ۷۵

حال می‌خواهیم با کمک این اعداد عملیات mutation را انجام دهیم. اگر \overline{xy} عدد دو رقمی تولید شده باشد، مقدار x امین حرف کروموزوم از سمت چپ را y واحد در پیمانه ۱۰ اضافه می‌کنیم. عملیات mutation را برای جمعیت بدست آمده در قسمت ب با توجه به اعداد دستگاه، به ترتیب از چپ به راست اعمال کنید.

د) آیا fitness جمعیت جدید نسبت به جمعیت اولیه بهتر شده است؟

ه) طبق جمعیت اولیه، آیا بدون انجام عملیات mutation امکان رسیدن به جواب بهینه (جواب با بیشترین fitness ممکن) وجود دارد؟ اگر بله، مسیر رسیدن به آن را بگویید یا ثابت کنید ممکن نیست.

۴. (۱۵ نمره، درجه سختی ۴) فرض کنید Π مهره در یک صفحه $n * n$ داریم. شما می‌توانید تمام Π مهره را همزمان کنترل کنید. چند مهره می‌تواند در یک خانه قرار بگیرند و هر مهره در هر لحظه از زمان می‌تواند به جهت شمال، جنوب، غرب یا شرق حرکت کند و یا این که سر جای خود ثابت باقی بماند. جدول حاوی تعدادی خانه مانع است که هیچ‌کدام از مهره‌ها نمی‌توانند در این خانه‌ها قرار بگیرند. هدف شما این است که تمام مهره‌ها با کمترین مرحله ممکن در یک خانه قرار بگیرند. (در یک مرحله‌ی زمانی تمام مهره‌ها می‌توانند همزمان با هم حرکت کنند.) این مسئله را به عنوان یک مسئله جستجو در نظر بگیرید:

الف) فرض کنید از الگوریتم A^* با کمک تابع اکتشافی^۴ برای حل این مسئله جستجو استفاده کنید. در هر مرحله زمانی $p_i = (x_i, y_i)$ موقعیت مهره i ام در صفحه است. برای هر کدام از توابع اکتشافی زیر، قابل قبول^۵ بودن و یک‌نوا^۶ بودن آن را مشخص کنید:

• تعداد جفت مهره‌هایی که در یک مکان یکسان قرار ندارند.

• اگر نزدیک‌ترین مهره‌ها فاصله حداقل ۴ داشته باشند، ۲ و در غیر اینصورت ۰

$$\frac{1}{4} \max_{i,j} |x_i - x_j| + \frac{1}{4} \max_{i,j} |y_i - y_j|$$

$$\frac{1}{4} \max(\max_{i,j} |x_i - x_j|, \max_{i,j} |y_i - y_j|)$$

ب) فرض کنید طبق توضیحات بخش قبل، یک تابع قابل قبول و یک‌نوا h_a برای حل مسئله داریم. حال مسئله تغییر می‌کند و شما $m \leq n$ مهره دارید که از سایر مهره‌ها سنگین‌ترند و بعد از هر مرحله‌ای که حرکت کنند، نیاز به ۱ واحد زمانی استراحت دارند (استراحت به این صورت است که سر جای خود می‌مانند). آیا می‌توان گفت که برای مسئله جدید همچنان h_a قابل قبول است؟ یک‌نواپی چگونه؟

^۴Heuristic

^۵Admissible

^۶Monotonic

۵. (۱۵ نمره، درجه سختی ۵) یک جدول $n * n$ داریم که هر خانه‌ی آن سیاه یا سفید است. در هر مرحله می‌توانیم یک زیرجدول از آن را انتخاب کنیم و رنگ خانه‌های درون آن را معکوس کنیم (از سیاه به سفید و بالعکس). می‌خواهیم با کمترین تعداد مرحله کاری کنیم که جدول کاملاً سفید شود.

الف) فضای مساله را گونه‌ای پیکربندی کنید به نحوی که حالت‌ها، کنش‌ها، ضریب انشعاب و حالت اولیه و نهایی مساله واضح باشد و آن‌ها را مشخص کنید.

ب) اندازه فضای مساله را به فرمت O بزرگ و بر حسب n بدست آورید.

ج) یک تابع اکتشافی نابدی‌ی برای حل این مساله ارائه دهید و این تابع را از لحاظ یک‌نوایی و قابل قبول بودن بررسی کنید.

۶. (۱۵ نمره، درجه سختی ۶) به سوالات زیر درباره الگوریتم A^* پاسخ دهید:

• ۱. یک مثال از یک گراف جهت‌دار و یک تابع heuristic نه لزوماً قابل قبول بزنید که در آن الگوریتم A^* مسیر بهینه را پیدا نکند. هزینه تمامی یال‌ها باید مثبت باشد و گراف حداکثر ۶ گره داشته باشد. گره‌های شروع و پایان را مشخص کنید و برای گره‌های باقی‌مانده مقدار تابع heuristic را بنویسید.

- آ) مسیر بهینه را مشخص نمایید.

- ب) مسیری که A^* پیدا می‌کند را مشخص نمایید.

• ۲. سوال بالا را برای گراف جهت‌دار بدون دور و یک تابع heuristic قابل قبول و نه لزوماً یکنوا با حداکثر ۸ گره حل نمایید.