



مقدمه

در این پروژه قرار است با استفاده از الگوریتم‌های جستجوی آگاهانه و ناآگاهانه که در درس هوش مصنوعی آموخته‌اید، راه حل مناسبی برای مسئله‌ای که در ادامه مطرح می‌شود بیابید و آن را پیاده‌سازی کنید.

توضیح مسئله

شرک و فیونا پس از مشکلات فراوان، بالاخره به هم رسیده‌اند و قصد برگزاری یک مراسم عروسی مجلل را دارند. آن‌ها خرِ شرک را به‌عنوان مسئول تدارکات انتخاب کرده‌اند. از آنجا که تعداد مهمانان بسیار زیاد است، خرِ شرک یک سالن بسیار بزرگ برای شام در نظر گرفته است. در این سالن، میزهای مهمانان در n ردیف و n ستون چیده شده‌اند. برای هر میز، یک لامپ جهت روشنایی در نظر گرفته شده است. برق‌کشی این لامپ‌ها به گونه‌ای است که اگر کلید هر لامپ را بزنیم، خود آن لامپ و چهار لامپ مجاور آن (بالا، پایین، چپ و راست) در صورت وجود از حالت روشن به خاموش یا از خاموش به روشن تغییر حالت می‌دهند. واضح است که اگر خرِ شرک کلیدهای درستی را انتخاب نکند، خاموش کردن همه لامپ‌ها پس از مراسم، زمان زیادی از او خواهد گرفت و برای مدیریت بخش‌های دیگر عروسی با مشکل مواجه خواهد شد.

هدف ما در این پروژه این است که به خرِ شرک کمک کنیم تا با کمترین تعداد کلید، تمام لامپ‌ها را خاموش کند.

ورودی

ورودی در قالب یک ماتریس $n \times n$ است که n نشان دهنده تعداد ردیف‌ها است (این پروژه برای $n \leq 5$ بررسی خواهد شد). در این ماتریس، وضعیت روشن یا خاموش بودن هر یک از لامپ‌های n ردیف به صورت ۰ و ۱ ارائه شده است که ۰ نشان‌دهنده خاموش بودن و ۱ نشان‌دهنده روشن بودن هر لامپ است.

نمونه ورودی
0 1 1 0
1 0 0 1
0 1 1 0
0 0 0 0

برای حل این مسئله کافیست که ابتدا کلید لامپ دوم از ردیف دوم و سپس کلید لامپ سوم از ردیف دوم زده شوند.

خروجی

خروجی هر الگوریتم شامل یک لیست با m عنصر و یک عدد (z) است که m تعداد کلیدهای لازم برای خاموش کردن تمام لامپ‌های سالن است و z برابر تعداد نودهای ویزیت شده است. هر عنصر در لیست مربوط به لامپ کلیدی است که باید زده شود و به صورت (x,y) نمایش داده می‌شود که x ردیف لامپ و y ستون لامپ است. شماره سطر و ستون‌ها از 0 در نظر گرفته می‌شود. در صورتی که مسئله جوابی نداشت در خروجی پیام مناسب نمایش داده می‌شود. مثال زیر یک نمونه از لیست مورد نظر است که پاسخ مثال بالا است. توجه کنید که چون مقدار z وابسته به الگوریتم است از نوشتن آن خودداری شده است.

نمونه خروجی
(1,1), (1,2)

پیاده‌سازی مسئله

در این پروژه، شما باید مسئله را با دو روش جستجوی ناآگاهانه‌ی BFS و IDS و همچنین روش جستجوی آگاهانه‌ی A^* حل و پیاده‌سازی کنید.

برای روش A^* ، بیش از دو heuristic معرفی و پیاده‌سازی کنید. توجه کنید که یکی از heuristic ها باید حتماً consistent باشد و جواب بهینه تولید کند. در انتها، $weighted A^*$ را با حداقل دو مقدار α پیاده‌سازی کرده و آن‌ها را با هم مقایسه کنید. همچنین، اگر برای یک مسئله چندین جواب وجود داشت، پیدا کردن یکی از جواب‌ها کافی است.

الگوریتم A^* Weighted

این الگوریتم نسخه‌ای از الگوریتم A^* است که در آن از یک وزن (w) برای تسریع فرایند استفاده می‌کنیم. تفاوت این الگوریتم با A^* در این است که تابع heuristic را در یک ضریب ثابت ضرب می‌کنیم. با این کار تفاوت بین دو مقدار مختلف در heuristic بیشتر شده و فرایند جستجو سریع‌تر خواهد شد. در این الگوریتم تابع هزینه به شکل زیر تعریف می‌شود:

$$f(n) = w * h(n) + g(n)$$

محدودیت زمانی اجرا

	3×3	4×4	5×5
BFS	1 ثانیه	5 ثانیه	-
IDS	1 ثانیه	5 ثانیه	-
A^*	1 ثانیه	1 ثانیه	2 دقیقه
Weighted A^*_1	1 ثانیه	1 ثانیه	2 دقیقه
Weighted A^*_2	1 ثانیه	1 ثانیه	2 دقیقه

سوالات

1. نحوه مدل کردن مسئله شامل تعریف initial state، goal state، action و ... را به طور دقیق توضیح دهید.
2. هر یک از الگوریتم‌های پیاده‌سازی شده را توضیح دهید و تفاوت‌ها و مزیت‌های هر یک نسبت به دیگری را قید کرده و عنوان کنید که کدام الگوریتم‌ها جواب بهینه تولید می‌کنند.
3. heuristic‌های معرفی شده در بخش جستجوی آگاهانه را توضیح داده و هر یک را از نظر admissible بودن و consistent بودن بررسی کنید.
4. سپس، الگوریتم را با استفاده از تمام heuristic‌هایی که معرفی کردید اجرا کرده و نتایج آن‌ها را با یکدیگر مقایسه کنید.
5. به ازای هر الگوریتم، تست کیس‌ها را اجرا کرده و میانگین زمان اجرای هر الگوریتم را ثبت کنید. برای هر تست کیس جدول زیر را کامل کنید (برای نمایش جدول زیر می‌توانید از کد موجود در نوت‌بوکی که به شما داده شده‌است استفاده کنید و نیازی به تغییر آن نیست).

	پاسخ مسئله (خروجی الگوریتم‌ها)	زمان اجرا
BFS		
IDS		
A*		
Weighted A* ₁		
Weighted A* ₂		

* جواب این سوالات را به صورت کامل در گزارش خود بنویسید.

نکات پایانی

- دقت کنید که کد شما باید به نحوی زده شده باشد که نتایج قابلیت بازتولید داشته باشند.
- توضیحات مربوط به هر بخش از پروژه را بطور خلاصه و در عین حال مفید در گزارش خود ذکر کنید. از ابزارهای تحلیل داده مانند نمودارها استفاده کنید. حجم توضیحات گزارش شما هیچ گونه تاثیری در نمره نخواهد داشت و تحلیل و نمودارهای شما بیشترین ارزش را دارد.
- سعی کنید از پاسخ‌های روشن در گزارش خود استفاده کنید و اگر پیش‌فرضی در حل سوال در ذهن خود دارید، حتما در گزارش خود آن را ذکر نمایید.
- پس از مطالعه کامل و دقیق صورت پروژه، در صورت وجود هرگونه ابهام یا سوال با طراحان پروژه در ارتباط باشید.
- نتایج، گزارش و کدهای خود را در قالب یک فایل فشرده با فرمت AI_CA1_[stdNumber].zip در سامانه ایلرن بارگذاری کنید. به طور مثال AI_CA1_810101999.zip
- محتویات پوشه باید شامل فایل پاسخ‌های شما به سوالات کتبی، فایل jupyter-notebook، خروجی html و فایل‌های مورد نیاز برای اجرای آن باشد. از نمایش درست خروجی‌های مورد نیاز در فایل html مطمئن شوید.
- توجه کنید این تمرین باید به صورت تک‌نفره انجام شود و پاسخ‌های ارائه شده باید نتیجه فعالیت فرد نویسنده باشد. در صورت مشاهده تقلب به همه افراد مشارکت‌کننده، نمره تمرین 100- و به استاد نیز گزارش می‌گردد. همچنین نوشته نشدن کدها توسط هوش مصنوعی نیز بررسی می‌شود!

موفق باشید