دانشگاه صنعتی امیرکبیر

دانشکده مهندسی کامپیوتر

«مبانی و کاربردهای هوش مصنوعی ترم بهار ۱۴۰۵»

تمرین سوم

جستجوی خصمانه

در انجام تمرین‌ها به نکات زیر توجه فرمائید:

۱- مطابق قوانین دانشگاه هر نوع کپی‌برداری و اشتراک کار دانشجویان غیرمجاز است و پاسخ به تمرین‌ها باید به صورت انفرادی انجام شود.

۲- پاسخ خود را باید در قالب یک فایل PDF به صورت تایپ‌شده و یا دستنویس (مرتب و خوانا) در سامانه کورسز آپلود نمائید؛ توجه کنید که مشکل ناخوانا بودن پاسخ‌ها در هنگام تصحیح، بر عهده دانشجو خواهد بود.

۳- در صورت هر گونه سوال و یا ابهام از طریق ایمیل زیر می‌توانید در ارتباط باشید:

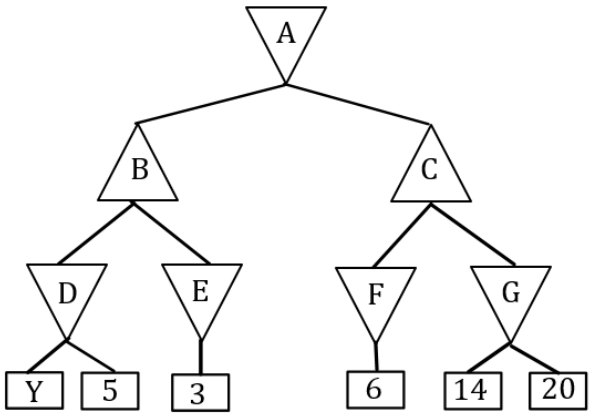
h.masroor@aut.ac.ir

۴- توجه نمائید پاسخ تمرین‌ها تنها در صورت آپلود در سامانه کورسز پذیرفته خواهد شد و ارسال پاسخ از طریق ایمیل و یا روش‌های دیگر بررسی نخواهد شد.

۵- فایل پاسخ‌نامه خود را با فرمت StudentID_HW03.pdf تا ساعت ۲۳:۵۹ روز ۱۴۰۵/۰۳/۱۲ فقط در بخش مربوطه در سایت درس آپلود نمائید.

سوال اول

درخت زیر را در نظر بگیرید. (گره‌های مثلث رو به بالا، بیشینه‌ساز (Max) و گره‌های مثلث رو به پایین، کمینه‌ساز (Min) هستند). مقدار Y یک عدد صحیح دلخواه است. بقیه مقادیر برگ‌ها در شکل مشخص شده‌اند.



الف) چه بازه‌ای از مقادیر برای Y باعث می‌شود که مقدار نهایی ریشه A (با بازی بهینه هر دو بازیکن) دقیقا برابر با خود Y باشد؟ پاسخ خود را به صورت یک نابرابری مانند $a \leq Y \leq b$ بنویسید و مراحل استدلال خود را کامل توضیح دهید.

ب) فرض کنید $Y = 4$ باشد. درخت را از چپ به راست با الگوریتم Minimax همراه با هرس آلفا-بتا پیمایش می‌کنیم (هرس در حالت تساوی مجاز است). کدام گره‌های برگ به خاطر هرس شدن بازدید نمی‌شوند؟ توضیح دهید در هر مرحله، مقدار آلفا و بتا چگونه تعیین می‌شود و چرا به آن برگ خاص می‌رسیم یا نمی‌رسیم.

ج) در این سوال، ریشه A یک گره Min است. حال می‌خواهیم همان بازی را با یک درخت جدید نمایش دهیم که ریشه آن Max باشد. چه تغییراتی باید در درخت جدید اعمال کنیم؟

سوال دوم

یک درخت بازی Minimax را در نظر بگیرید:

- گره ریشه، یک بیشینه‌ساز است.
- درخت می‌تواند هر عمقی داشته باشد و مقادیر گره‌های برگ هر عدد دلخواهی می‌تواند باشد.
- وقتی یک اقدام جدید به انتهای درخت اضافه می‌کنیم، بخش‌های قبلی درخت بدون تغییر می‌مانند.

می‌خواهیم مقدار ریشه را قبل و بعد از اضافه کردن یک اقدام جدید مقایسه کنیم. در هر بخش، شرح دهید که مقدار ریشه چه تغییری می‌کند (افزایش، کاهش، بدون تغییر، یا هر سه حالت ممکن) و چرا.

الف) اگر به همه گره‌های کمینه‌ساز، یک اقدام جدید اضافه کنیم، مقدار ریشه چه تغییری می‌کند؟ اگر به همه گره‌های بیشینه‌ساز یک اقدام جدید اضافه کنیم چطور؟ با مثال توضیح دهید.

حال فرض کنید درخت Expectimax و فقط شامل بیشینه‌ساز، شانس (Chance) و برگ است و هیچ گره کمینه‌سازی وجود ندارد.

ب) اگر به همه گره‌های شانس، یک اقدام جدید اضافه کنیم، مقدار ریشه چه تغییری می‌کند؟ برای هر حالت یک مثال بزنید.

ج) اگر به همه گره‌های بیشینه‌ساز یک اقدام جدید اضافه کنیم، آیا نتیجه با حالت Minimax برای بیشینه‌سازها تفاوت دارد؟ توضیح دهید.

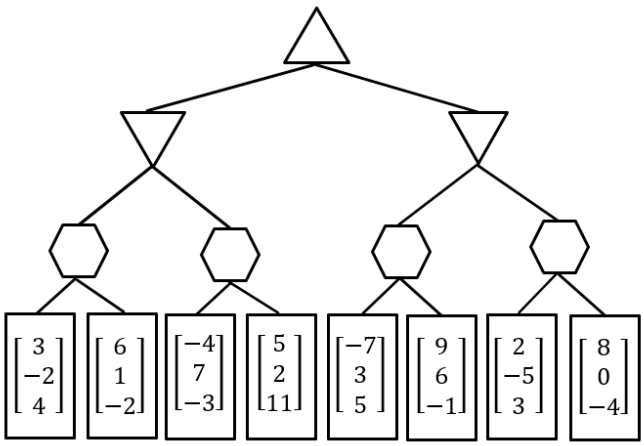
سوال سوم

یک درخت بازی چندعامله (Multi-agent Game Tree) را با ســاختار زیر در نظر بگیرید که در آن سه عامل مختلف با اهداف متفاوت در حال تصمیم‌گیری هستند:

گره ریشــه (گره max): عامل اول اســت و مقدار اول (بالا) از بردار مطلوبیت را بیشــینه (Maximize) می‌کند.

گره‌های لایه میانی (گره min): عامل دوم هســـتند و مقدار دوم (وســـط) از بردار مطلوبیت را کمینه (Minimize) می‌کنند.

گره‌های لایه پایین (شش‌ضلعی): عامل سوم هستند و قدر مطلق مقدار سوم (پایین) از بردار مطلوبیت را کمینه (Minimize) می‌کنند. (در صورت برابری قدر مطلق‌ها، گره سمت چپ انتخاب می‌شود).



الف) بردار مطلوبیتی که در نهایت به گره ریشـــه می‌رســـد را محاســـبه کنید. گره ریشـــه چه مقداری (Utility) را برای خود دریافت می‌کند؟ پاسخ خود را گام‌به‌گام توضیح دهید.

ب) آیا در این درخت بازی چندعامله امکان هرس کردن (Pruning) گره‌ها وجود دارد؟ دلیل خود را به طور کامل بیان کنید.

سوال چهارم

در این مسئله، یک درخت بازی با دو عامل به نام‌های تام و جری را در نظر بگیرید:

تام به عنوان یک عامل بیشینه‌ساز (Maximizer) استاندارد عمل می‌کند و گره‌های او با مثلث رو به بالا نشان داده می‌شوند.

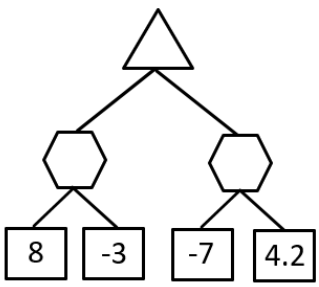
جری رفتاری متفاوت دارد. گره‌های او با شش‌ضلعی نشان داده می‌شوند. او به جای کمینه کردن مستقیم مقادیر، یک تابع خاص به نام $f(x)$ را بر روی مقادیر فرزندانش اعمال کرده و سپس فرزندی را انتخاب می‌کند که مقدار $f(x)$ برای آن کمینه باشد. (در صورت برابری مقادیر $f(x)$ برای چند فرزند، جری گره سمت چپ را انتخاب می‌کند).

مقادیری که در برگ‌های درخت (Terminal Nodes) نوشته شده‌اند، صرفا نشان‌دهنده‌ی مطلوبیت (Utility) تام هستند، نه جری.

فرض کنید تابع جری به صورت زیر تعریف شده است:

$$f(x) = \begin{cases} 5 & x \geq 0 \\ -2 & x < 0 \end{cases}$$

با توجه به درخت زیر، تام چه مقدار مطلوبیتی دریافت خواهد کرد؟ توضیح دهید.



سوال پنجم

الف) آیا اگر تابع ارزیابی بتواند برای مقدار واقعی یک حالت کران بالا یا کران پایین تضمین شده ارائه دهد، می‌توان هرس آلفا-بتا را بهبود داد؟ چگونگی بهبود را با یک مثال توضیح دهید.

ب) در یک درخت expectimax اگر بدانید مقدار گره‌ها در یک بازه $[L, U]$ قرار دارد، ترتیب بررسی فرزندهای یک گره شانس چگونه باشد تا احتمال هرس کردن افزایش پیدا کند ؟ راه حل خود را توضیح دهید.

در یک الگوریتم هرس جدید به نام b_pruning بجای ذخیره‌سازی آلفا و بتا در الگوریتم هرس آلفا-بتا، در هر سطح از مسیر فعلی ریشه تا گره بهترین مقدار پیدا شده از نظر بازیکن همان سطح را ذخیره می‌کنیم. این مقدار در هر سطح iام را با b_i نشان می‌دهیم. فرض کنید ریشه درخت min است.

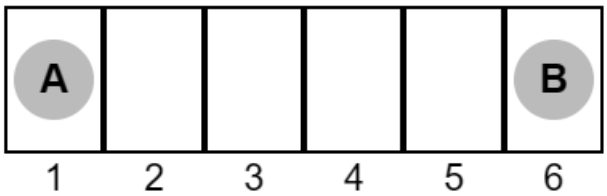
ج) در حالت جدید، اگر گره‌های min و max بخواهند درخت را هرس کنند، چگونه می‌توانند انجام دهند؟ به طور کامل توضیح دهید.

د) شبه کد الگوریتم با minimax با استفاده از b_pruning را بنویسید.

ه) توضیح دهید در b_pruning چرا تعداد شاخه‌های هرس شده فرقی با هرس آلفا-بتا ندارد.

سوال ششم

در بازی زیر بازیکن A باید به جای بازیکن B برود و بازیکن B هم به جای بازیکن A برود. اگر بازیکن A به زودتر از B به نقطه مورد نظر خود برسد، برنده می‌شود اما اگر B زودتر از A برسد، بازیکن A بازی را می‌بازد. هر بازیکن فقط می‌تواند به چپ یا راست(در صورت نبودن در خانه اول یا خانه آخر) حرکت کند. اگر دو بازیکن به هم رسیده باشند، بازیکنی که نوبت حرکت آن است می‌تواند از روی بازیکن دیگر بپرد و به خانه مجاور بازیکن دیگر بیفتد. فرض کنید شروع کننده بازی A است. بازیکن‌ها به بهترین شکل بازی می‌کنند. برد B 100 امتیاز و برد A 100- امتیاز دارد.



الف) اشکال الگوریتم minimax استاندارد در این بازی چیست؟

ب) فرض کنید برای رفع مشکل قسمت قبل از جست‌وجوی عمق محدود استفاده می‌کنیم. تابع ارزیابی مناسبی را برای این روش از دید بازیکن A پیشنهاد دهید و سپس الگوریتم minimax با محدودیت عمق ۶ را اجرا کنید. با توجه به درختی که پس از اجرا ساخته‌اید، در هر حالت‌هایی که بازیکن A تصمیم گیرنده است، بازیکن A چه حرکتی را انتخاب می‌کند؟ (هر حالت به صورت دوتایی (موقعیت B، موقعیت A) نمایش داده شود.)

فرض کنید اگر B بتواند بفهمد که بازنده است، الگوریتم minimax را اجرا نمی‌کند و از بازی کنار می‌کشد.

ج) تعداد خانه‌ها چقدر باشد تا B بداند قطعا بازنده است؟ پاسخ خود را توضیح دهید.

د) اگر A نیز بتواند همان تصمیم B را بگیرد، تعداد خانه‌ها چقدر باشد تا A بداند قطعا بازنده است؟ پاسخ خود را توضیح دهید.