

به نام خالق زیبایی ها

پروژه دوم
درس مبانی هوش مصنوعی

امیرفاضل کوزه گر کالجی

9931099

بخش اول:

پاسخ 1-1: ابتدا متغیری به نام score با مقدار اولیه 0 تعریف می کنیم. سپس، حاصل تفریق امتیاز استیت successor با امتیاز سطح فعلی بازی را، با امتیازمان جمع می کنیم.

اگر پکمن حرکت نمی کرد از آن 10 امتیاز کم می کنیم.

به ازای قرص های قدرت، نیز یک میزان امتیازی به عامل می افزاییم.

حال اگر میزان غذای باقیمانده در استیت successor کمتر از غذای باقیمانده فعلی بود، به امتیازمان می افزاییم.

حال دو حال داریم. اینکه عامل روح تا به حال ترسیده باشد یا نه؟

اگر ترسیده باشد:

اگر فاصله با روح ترسیده کمتر از فاصله با دیگر روح ها باشد، به امتیاز می افزاییم در غیر این صورت آن را کم می کنیم.

اگر نترسیده باشد:

اگر فاصله با روح نترسیده کمتر از فاصله با باقی روح ها باشد، کجازات می شود در غیر این صورت، به امتیاز افزوده می شود.

2-1: از عکس آن مقدار استفاده می توان کرد همانند افزودن عکس جمع فاصله تا غذا ها را به امتیاز اصلی.

3-1: در 1-1 توضیح داده شده.

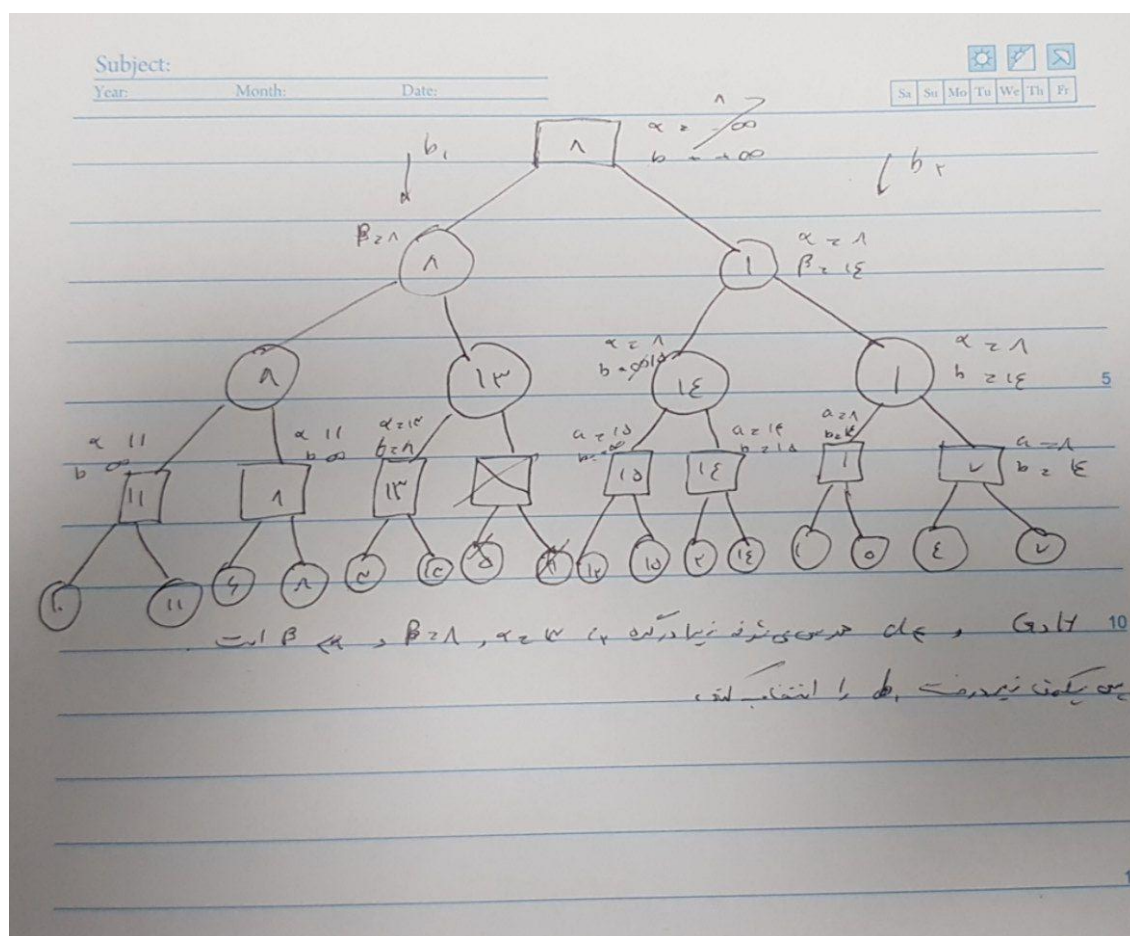
بخش دوم:

2-1: یکمن آگاه است که نمی تواند به هدف برسد و همچنین می داند با هدر دادن زمان، امتیازش کمتر است پس تلاش می کند تا مینیمم ضرر را داشته باشد پس به سمت روح می رود تا کمترین زمان را از دست داده و مینیمم ضرر را کسب کند.

2-2:

بخش سوم:

1-3:



2-3: هرس آلفا بتا ممکن است در گره های میانی مقادیر متفاوتی با مقادیر بدست آمده بدون هرس تولید کند. دو پارامتر آلفا و بتا داریم که یکی برای حد بالا و دیگری برای حد پایین است. حال اگر حد پایین (آلفا) بزرگتر از حد بالا (بتا) شود دیگر لازم نیست جست و جو را ادامه دهیم و هرس می کنیم. حال از آنجایی که یک شاخه را هرس کردیم، ممکن است مقدار

خروجی زیر درختی که هرس شده است کمتر از مقداری باشد که عامل ما انتخاب کرده است (عامل مینیمم) اما از آنجایی که می دانیم در سطح بالاتر مقدار این عامل انتخاب نمی شود، در نتیجه برایمان مهم نیست که مقدار آن کمتر باشد. اما اگر از α بتا استفاده نکنیم، چون از پایین شروع به مقدار دهی می کنیم، تمام مقادیر درست انتخاب می شوند که ممکن است با مقادیر الگوریتم α بتا متفاوت باشند. اما این تفاوت فقط در گره های میانی است و امکان ندارد در ریشه رخ دهد. زیرا مقادیر α بتا بر اساس عامل ریشه (مینیمم یا ماکسیمم) تغییر می کنند و در نهایت جواب یکسانی دارند. در واقع ریشه چون بالاتر از بقیه گره ها قرار دارد مقدار آن قطعا در هر دو الگوریتم یکسان می شود.

3-3: برخی اوقات این امکان وجود دارد که هرس α بتا در درخت minimax، اشتباه شود:

- وقتی ترتیب نود ها بهینه نباشد
- وقتی درخت جست و جو نا متقارن باشد
- وقتی که تابع هیوریستیک admissible نباشد.

بخش چهارم:

4-1: minimax تصور می کند که بازی zero-sum است یعنی به گونه ای که یک عامل می برد و دیگری می بازد. اما expectiminimax، مناسب بازی ای است که عامل هایی با تصمیم های رندوم و شانسی در آن وجود دارد. یعنی نتیجه یک عمل، آنچنان مشخص نیست.

4-2:

برای استفاده از رولت ویل در حالت و با توجه به مقدار آن یک احتمال از پیش تعیین شده برای آن در نظر می گیریم و در \expValue انجام دهیم در تابع نهایت به صورت رندوم یکی را انتخاب می کنیم.

حال اگر نیاز بود تا بیشتر از یک حالت انتخاب شود، به ازای هر حرکت مجاز یکبار تابع \expValue صدا شود و در نهایت بین مقادیر آن ها ماکسیمم گرفته می شود.