



ترم اول سال
تحصیلی ۹۴ - ۹۳

پروژه درس هوش مصنوعی

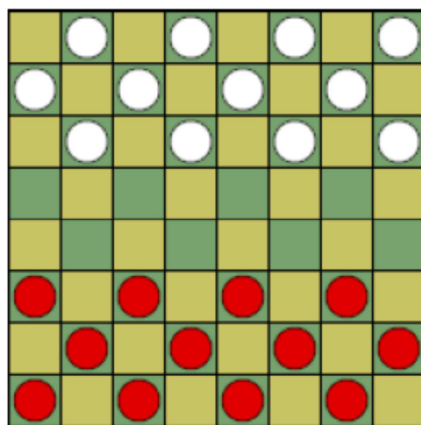
در این پروژه دانشجویان باید یکی از سیستم‌های هوشمند زیر را به دلخواه پیاده‌سازی کنند.

۱. الگوریتم ژنتیکی مرکب (hybrid) با جستجوی تپهنوردی: یک نمایش بیتی (دودویی) برای افراد (individuals) جمعیت در نظر گرفته می‌شود. جمعیت اولیه به صورت تصادفی با یک توزیع یکنواخت روی مقادیر هر یک از ژن‌ها تولید می‌شود. در هر نسل از الگوریتم، تمام افراد جمعیت بر اساس برازندگی مرتب شده و درصد مشخصی (#ParentPercent) از افراد بهتر برای بازتولید (reproduction) انتخاب می‌شوند. سپس به تعداد نصف فرزندان مورد نیاز زوج در نظر گرفته می‌شود که برای تشکیل هر زوج از بین افراد انتخاب شده دو فرد انتخاب می‌شوند. یک فرد می‌تواند بر اساس برازندگی خود در زوج‌های متعددی شرکت کند اما شانس شرکت در زوج‌های جدید باید هر بار کمتر شود. پس از زوج‌بندی، در هر زوج ابتدا عملگر تقطیع دو نقطه‌ای اعمال شده و دو فرزند تولید می‌شود و سپس به هر یک از فرزندان به صورت جداگانه عملگر جهش اعمال می‌شود. نقاط تقطیع و مکان جهش در هر زوج به صورت تصادفی انتخاب می‌شوند. پس از آن هر یک از فرزندان با استفاده از جستجوی تپهنوردی در صورت امکان بهبود داده می‌شود. برای اینکار جستجوی تپهنوردی با شروع از یک فرد در همسایگی آن به دنبال افراد بهتر می‌گردد و اینکار را تا جایی ادامه می‌دهد که فرد بهتری پیدا نکند یا به تعداد دفعات مشخصی (#MaxImprove) فرد مورد نظر بهبود داده شده باشد. جستجوی تپهنوردی امکان حرکت به کنار (sideway moves) با بکارگیری یک لیست Tabu با اندازه ۱۰ را دارد. حداکثر حرکت‌های به کنار ۳۰ است. در پایان این نسل از الگوریتم، فرزندان بهبود داده شده، با افراد موجود در جمعیت مقایسه شده و جایگزین افراد بدتر می‌شوند. این فرآیند تا برآورده شدن شرط توقف که همگرا شدن جمعیت یا رسیدن به حداکثر نسل‌های ممکن (#MaxGen) است، ادامه می‌یابد. یک فایل ورودی بنام "params.txt" حاوی پارامترهای تعیین کننده عملکرد الگوریتم به عنوان ورودی به برنامه پیاده‌سازی شده داده می‌شود. سطرهای این فایل به ترتیب شامل موارد زیر است:

- IndivSize xxx
- PopSize xxx
- ParentPercent xxx
- OffspringPercent xxx
- MaxGen xxx
- CrossoverProb xxx
- MutationProb xxx
- MaxImprove xxx

برای تعیین برابری افراد، الگوریتم پیاده‌سازی شده باید از یک تابع به صورت $y=f(x)$ استفاده کند که در آن x نمایش بیتی یک فرد و y میزان برابری آن است. تعریف این تابع که باید بیشینه شود متعاقباً در اختیار دانشجویان قرار می‌گیرد (ممکن است توابع متعددی برای آزمایش الگوریتم پیاده‌سازی شده داده شود). الگوریتم پیاده‌سازی شده باید یک فایل ثبتي (log) ایجاد کند که در آن برای هر نسل، بهترین فرد جمعیت، میزان برابری بهترین فرد، میانگین برابری افراد جمعیت، تعداد افراد متفاوت جمعیت و تعداد دفعات فراخوانی تابع برابری تا آن لحظه را نشان دهد. زبان برنامه‌نویسی برای پیاده‌سازی الگوریتم به دلخواه است.

۲. **بازیکن برای بازی Checkers:** این بازی دونفره است و بازیکنان بصورت نوبتی مهره‌های خود را حرکت می‌دهند. بازی در یک صفحه 8×8 انجام می‌شود و مهره‌ها فقط می‌توانند در خانه‌های مشکی رنگ قرار گرفته و حرکت کنند. حالت اولیه مهره‌ها به صورت زیر است:



مهره‌ها به صورت قطری (diagonal) حرکت می‌کنند و اگر در حرکت خود از روی مهره حریف رد شوند، مهره حریف زده خواهد شد (و از صفحه بازی خارج می‌شود). مهره‌ها فقط می‌توانند به سمت جلو حرکت کنند، مگر اینکه به آخرین سطر مقابل خود برسند که در اینصورت به یک مهره ویژه تبدیل خواهند شد و قابلیت حرکت به عقب را نیز پیدا می‌کنند. هر بازیکنی که نتواند هیچ حرکتی انجام دهد یا تمام مهره‌های خود را از دست بدهد، بازنده می‌شود. سطرهاى صفحه بازی از بالا به پایین با اعداد ۱ تا ۸ و از چپ به راست با حروف A تا H شماره‌گذاری می‌شوند. صفحه بازی همیشه به شکلی قرار دارد که خانه سمت چپ بالا (1A) سفید رنگ باشد. برای اطلاع بیشتر از قوانین بازی به مطالب تکمیلی موجود در اینترنت مراجعه کنید.

هدف پیاده‌سازی یک برنامه بازیکن برای این بازی با استفاده از یکی از روش‌های مبتنی بر درخت بازی (جستجوی آلفا - بتا، قطع و ارزیابی، ...) است و در غیر اینصورت پیاده‌سازی فاقد ارزش است. استفاده از مراجعه (lookup) در کنار جستجو بلامانع است. برنامه پیاده‌سازی شده باید دارای یک بخش مستقل (تابع، شیء، فرآیند ...) برای بازیکن باشد که با دریافت یک حالت از صفحه بازی (محل قرارگیری مهره‌ها در صفحه)، یک حرکت معتبر را انتخاب کند. بدنه اصلی برنامه با شروع از حالت اولیه بازی در هر نوبت حرکت انتخاب شده توسط بازیکنان را در بازی اعمال می‌کند و حالت جدید صفحه (در صورت نیاز حذف برخی مهره‌ها) را بدست می‌آورد. بازیکن شروع کننده بازی به صورت تصادفی انتخاب می‌شود. این تکرار تا جایی

ادامه پیدا می‌کند که یکی از بازیکنان نتواند هیچ حرکتی را برای حالت ورودی داده شده انتخاب کند. در اینصورت برنامه اصلی بازیکن دیگر را به عنوان برنده مشخص می‌کند. بدنه اصلی همچنین باید بتواند یک فایل ورودی شامل یک حالت از بازی را دریافت و بازی را از آن حالت به بعد ادامه دهد. فایل مشخص کننده حالت ورودی شامل ۸ سطر خواهد بود که متناظر با سطرهاى صفحه بازی هستند و خانه‌های مشکی رنگ هر سطر با ویرگول از هم جدا شده‌اند (در هر سطر ۳ ویرگول وجود دارد). هر خانه مشکی رنگ می‌تواند خالی، دارای مقدار 1 برای یک مهره سفید، مقدار 11 برای یک مهره سفید ویژه، مقدار 2 برای یک مهره سیاه و مقدار 22 برای یک مهره سیاه ویژه باشد. به عنوان مثال فایل حاوی حالت اولیه بازی به شکل زیر قابل بیان است:

1,1,1,1

1,1,1,1

1,1,1,1

'''

'''

2,2,2,2

2,2,2,2

2,2,2,2

برنامه اصلی باید یک فایل ثبتي (log) ایجاد کند که در آن با ذکر حالت شروع بازی، حرکت‌های هر یک از بازیکنان در هر نوبت و تأثیر آن در صفحه بازی را نشان دهد و در نهایت بازیکن برنده را اعلام کند. زبان برنامه‌نویسی به دلخواه است.

۳. موضوع سوم متعاقباً اعلام خواهد شد.

دانشجویان نکات زیر را برای پیاده‌سازی پروژه‌ها در نظر بگیرند:

- دانشجویان می‌توانند پروژه‌ها را در گروه‌های حداکثر ۳ نفری انجام دهند.
- پروژه‌ها به صورت حضوری (زمان ارائه متعاقباً اعلام خواهد شد) باید با حضور تمام اعضاء گروه ارائه شود.
- دانشجویان باید تمام نرم‌افزارهای مورد نیاز برای اجرا شدن کامل برنامه خود را پیش از تحویل پروژه تهیه و در کامپیوتر مورد استفاده برای تحویل پروژه نصب کرده باشند.
- در هنگام تحویل پروژه از تمام اعضاء گروه سوال خواهد شد. تمام اعضاء باید با نحوه عملکرد برنامه آشنا بوده و بتوانند به سوالات مطرح شده پیرامون برنامه پاسخ دهند.
- مسئولیت هر گونه عملکرد پیش‌بینی نشده و نقص در کارکرد مورد نظر برنامه در هنگام تحویل بر عهده دانشجو است.

● دانشجویان باید تا قبل از تحویل پروژه فایل‌های کد و اجرایی برنامه را به صورت فشرده شده در قالب یک فایل zip از طریق ایمیل hkarshenas@chmail.ir ارسال کنند. علاوه بر این، فایل zip ارسالی باید حاوی یک فایل توضیحی حاوی اطلاعات زیر باشد:

- موضوع انتخاب شده برای پیاده‌سازی
- نام اعضای گروه همراه با شماره دانشجویی
- نام و توضیح مختصر هر یک از فایل‌های کد برنامه
- شرح برنامه‌های مورد نیاز برای اجرا و چگونگی نصب، راه‌اندازی و اجرای برنامه

موفق باشید

کارشناس