دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

هوش مصنوعی

بهار ۱۴۰۵

استاد: ماری‌اوریاد
گردآورندگان: رادین جریره، سید محمدرضا جوادی، کیارش شیخی، امیرپارسا صفری، کیمیا علوی، علی مقدسی

تمرین اول — **جستجوی ناآگاهانه، آگاهانه و محلی** — مهلت ارسال: ۱۱ اردیبهشت

- مهلت ارسال پاسخ تا ساعت ۲۳:۵۹ روز مشخص‌شده است.
- در طول ترم امکان ارسال با تاخیر پاسخ همه‌ی تمارین تا سقف ۴ روز و در مجموع ۱۰ روز، وجود دارد. پس از گذشت این مدت، پاسخ‌های ارسال‌شده پذیرفته نخواهند بود. همچنین، به ازای هر ساعت تأخیر غیر مجاز ۰.۵ درصد از نمره تمرین به صورت ساعتی کسر خواهد شد.
- هم‌کاری و هم‌فکری شما در انجام تمرین مانعی ندارد اما پاسخ ارسالی هر کس حتما 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.
- لطفا تصویری واضح از پاسخ سوالات نظری بارگذاری کنید. در غیر این صورت پاسخ شما تصحیح نخواهد شد.
- پاسخنامه سوالات را در سامانه کوئرا و به فرمت AI_HW1_<STUDENT_ID> بارگزاری کنید.

سوالات نظری (۱۰۰ نمره)

۱. (۲۰ نمره) درستی و نادرستی عبارات زیر را با ذکر دلیل مشخص کنید.

(آ) پیچیدگی فضایی الگوریتم DFS در جستجوی گرافی، همچنان در اردر خطی $O(bm)$ است، زیرا در هر لحظه فقط گره‌های مسیر فعلی در حافظه قرار دارند.

(ب) اگر h_1 و h_2 دو تابع اکتشافی [1] یک‌نوا [2] باشند آنگاه $\frac{h_1+h_2}{2}$ نیز یک‌نوا است.

(ج) اگر h_1 یک تابع یک‌نوا باشد و c یک عدد ثابت بیشتر از ۱ باشد آنگاه $h_2 = c \times h_1$ نیز یک‌نوا است.

(د) جستجوی A* در حالت درختی با یک تابع اکتشافی قابل قبول، همواره بهینه است.

(ه) با کاهش دما، عملکرد الگوریتم Simulated Annealing به عملکرد الگوریتم Local Beam Search با $k = 1$ میل می‌کند.

(و) الگوریتم IDS در بدترین حالت از نظر تعداد گسترش‌ها دقیقاً به اندازه BFS گره باز می‌کند.

(ز) اگر کوچک‌ترین هزینه یال برابر صفر باشد (یعنی یال‌هایی با هزینه ۰ وجود داشته باشد)، الگوریتم UCS ممکن است حتی در صورت وجود پاسخ با هزینه متناهی، خاتمه نیابد.

(ح) اگر h_1 و h_2 یک‌نوا باشند، آنگاه $min(h_1, h_2)$ نیز یک‌نوا است.

(ط) در Local Beam Search با فرض محدود بودن زمان و حافظه، افزایش K لزوما منجر به افزایش احتمال یافتن استیت هدف می‌شود.

(ی) اگر با شروع از وضعیت S، الگوریتم Hill Climbing به وضعیت هدف G برسد، آنگاه الگوریتم Local Beam Search با $K > 1$ که S جزئی از جمعیت اولیه‌اش می‌باشد نیز لزوما به وضعیت G می‌رسید.

[1] heuristic
[2] consistent

۲. (۱۰ نمره) درخت کاملی با ضریب انشعاب b و عمق d را در نظر بگیرید. هدف در عمق m قرار دارد.

(آ) در بهترین و بدترین حالت، تعداد گره‌های بسط داده شده توسط DFS و BFS را محاسبه کنید.

(ب) سه حالت مختلف را برای گره هدف (که در عمق m قرار دارد) در نظر بگیرید: هدف چپ‌ترین گره در این عمق باشد، هدف راست‌ترین گره در این عمق باشد و یا هدف به احتمال برابر در یکی از گره‌های موجود در این عمق قرار داشته باشد. در هرکدام از این سه حالت عملکرد الگوریتم‌های BFS و DFS را مقایسه کرده و بگویید استفاده از کدام یک بهتر است.

(ج) برای چه مقادیری از m و d ، استفاده از BFS منطقی‌تر است؟

۳. (۲۰ نمره) مسئله‌ی مرتب‌سازی n عدد صحیح متمایز را در نظر بگیرید ($n \geq ۳$). فرض کنید در هر مرحله می‌توانیم دو عضو همسایه را با هم جابه‌جا کنیم. می‌خواهیم دنباله را به صورت صعودی مرتب کنیم.

(آ) فضای مسئله را به گونه‌ای مدل‌سازی کنید که حالت‌ها، کنش‌ها، حالت اولیه و حالت نهایی مسئله مشخص باشد.

(ب) دو تابع fitness مختلف ارائه کنید: تابع اول که نقطه بهینه محلی غیرسراسری داشته باشد و تابع دوم که دارای چنین نقطه‌ای نباشد.

(ج) با در نظر گرفتن تابع fintess دومی که در بخش قبل ارائه کردید، الگوریتم Hill Climbing را مرحله به مرحله تا اتمام الگوریتم روی دنباله‌ی (۱۵, ۲۴, ۹, ۸) اجرا کنید.

(د) با در نظر گرفتن تابع دوم و جدول دمای زیر، ۸ مرحله اول الگوریتم Simulated Annealing را روی دنباله‌ی بخش (ج) اجرا کنید:

t	۱	۲	۳	۴	۵	$\geq$ ۶
T	۵	۴	۳	۲	۱	۰

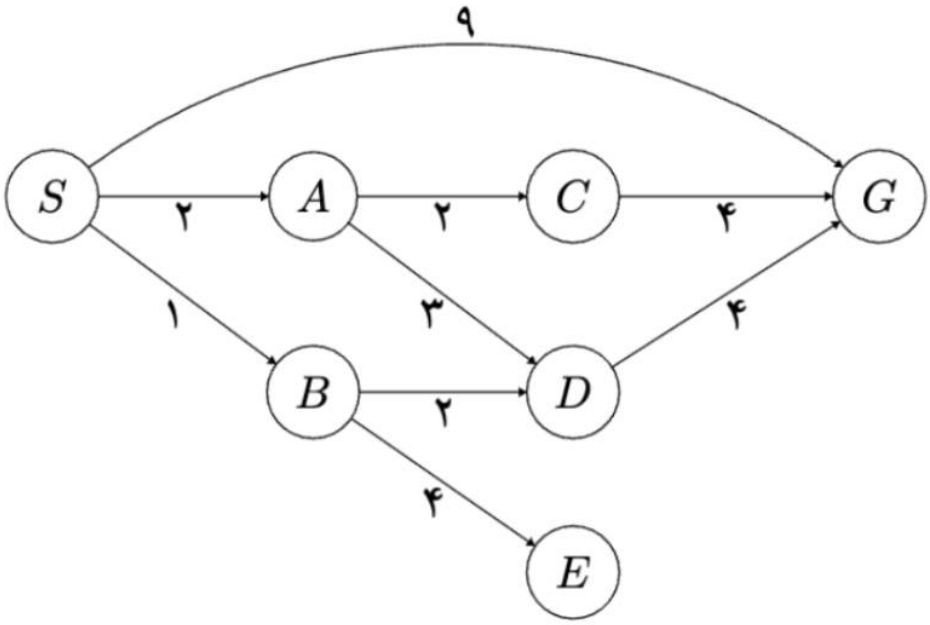
در اجرای این الگوریتم، هنگام نمونه‌برداری کردن همسایه‌های یک حالت، به ترتیب جابه‌جایی‌ها را از چپ به راست انجام دهید. این یعنی اولین همسایه نمونه‌برداری شده از یک حالت، با جابه‌جا شدن دو عضو اول دنباله آن حالت به دست می‌آید، و دومین همسایه نیز با جابه‌جا شدن اعضای دوم و سوم آن دنباله ساخته می‌شود. برای نمونه‌برداری از هر حالت، ابتدا الگوریتم را روی اولین همسایه آن (مطابق ترتیب تعریف شده) اعمال کنید، و اگر این همسایه به عنوان حالت بعدی انتخاب نشد به همسایه دوم و سوم رفته و اگر هیچ‌کدام انتخاب نشدند الگوریتم را به پایان برسانید. برای تولید اعداد تصادفی جهت انتخاب یا عدم انتخاب همسایه نیز می‌توانید از اعداد زیر استفاده کنید. (ممکن است تمامی اعداد استفاده نشوند.)

$$\{۰/۲۷, ۰/۶۶, ۰/۹۱, ۰/۱۳, ۰/۶۲, ۰/۳۶, ۰/۰۴\}$$

۴. (۲۰ نمره) گراف جستجوی زیر را در نظر بگیرید. S راس شروع و G هدف است. هزینه هر یال بر روی آن نوشته شده است و یال‌ها دو طرفه اند. برای هر یک از الگوریتم‌های زیر مشخص کنید کدام راس در هر مرحله باز می‌شود و مقدار هر گره چگونه بروزرسانی می‌شوند. مسیری که هر استراتژی خروجی می‌دهد را مشخص کنید (اولویت باز کردن رئوس طبق ترتیب حروف انگلیسی است.)

- جستجوی BFS
- جستجوی DFS
- جستجوی UCS
- جستجوی Greedy
- جستجوی درختی A*

- جستجوی گرافی *A



هیوریستیک

G	E	D	C	B	A	S
۰	۱۰	۱	۴	۶	۰	۶

۵. (۱۵ نمره) یک الگوریتم ژنتیک را در نظر بگیرید که هر کروموزوم به صورت زیر نمایش داده می‌شود:

$$x = ABCDEF$$

هر ژن مقداری بین ۰ تا ۹ می‌گیرد. تابع fitness به صورت زیر تعریف شده است:

$$f(x) = ۲A + B + ۳C - D + ۲E + F$$

جمعیت اولیه:

$$x_۱ = ۵۲۳۶۴۱, \quad x_۲ = ۸۱۷۲۰۳, \quad x_۳ = ۴۶۹۱۲۵, \quad x_۴ = ۹۰۲۵۷۸$$

(آ) fitness هر کروموزوم را محاسبه کرده و بهترین آن را مشخص کنید.

(ب) تقاطع را به صورت زیر اعمال کنید:

- روی دو کروموزوم با بالاترین fitness، عملگر تقاطع تک‌نقطه‌ای (بعد از ژن سوم) اعمال کنید.
- روی دومین و سومین کروموزوم برتر، تقاطع دومحوره انجام دهید به‌گونه‌ای که ژن‌های CD از یکی از والدین و بقیه ژن‌ها از والد دیگر بیایند.

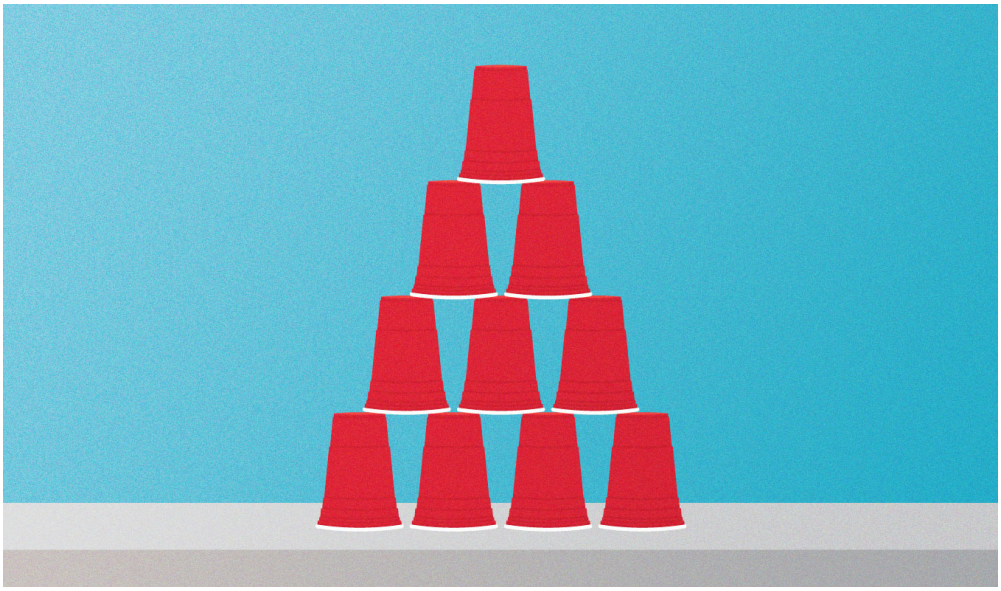
(ج) فرض کنید عملگر جهش همیشه و با احتمال یک روی تمامی جمعیت اعمال می‌شود، و طریقه اعمال آن با دو مقدار تصادفی x و y به صورت زیر است:

- ژن در موقعیت x انتخاب می‌شود
- مقدار y به صورت پیمانه ۱۰ به آن اضافه می‌شود

با مقادیر تصادفی $(x = ۳, y = ۴), (x = ۷, y = ۲), (x = ۱, y = ۵), (x = ۶, y = ۰)$ عمل جهش را به ترتیب روی فرزندان اعمال کنید.

(د) میانگین fitness جمعیت جدید را با جمعیت اولیه مقایسه کنید.

۶. (۱۵ نمره) یک برج ساخته‌شده از لیوان‌ها داریم که به صورت لایه‌به‌لایه روی هم قرار گرفته‌اند. لیوان‌هایی که در پایین‌ترین ردیف قرار دارند مستقیماً روی میز هستند، و هر لیوان در ردیف‌های بالاتر دقیقاً روی دو لیوان زیرِ خود قرار دارد. در هر مرحله می‌توانیم لیوانی که حداکثر یک لیوان بالای آن است را انتخاب کرده و با ضربه توپ آن لیوان و تمامی لیوان‌هایی که مستقیم و غیرمستقیم بالایش هستند را فرو بریزیم. هدف بازی این است که با انجام کمترین تعداد ضربه، تعداد لیوان‌های فروریخته بیشتر از نیمی از کل لیوان‌ها شود.



(آ) فضای مسئله را طوری تعریف کنید که شامل حالت‌ها، کنش‌ها، هزینهٔ کنش‌ها و حالت(های) هدف باشد.

(ب) وابستگان یک لیوان خاص را تعریف می‌کنیم: تمامی لیوان‌هایی که در صورت هدف قرار گرفتن آن لیوان فرو می‌ریزند. اگر تعداد وابستگان لیوان c را با $f(c)$ نمایش دهیم، تابع اکتشافی زیر را تعریف می‌کنیم:

$$h(S) = \frac{\text{cups}_{\text{needed}}}{\max\limits_{c \in \text{targetable cups}(S)} f(c)}$$

که در آن صورت کسر برابر با کمینهٔ تعداد لیوان‌هایی است که نیاز داریم سرنگون کنیم تا برنده شویم و مخرج آن بیشینه تعداد لیوان‌هایی است که می‌توانیم با یک ضربه توپ بیاندازیم. بررسی کنید که آیا این تابع قابل قبول و یک‌نوا هست یا نه و دلایل خود را توضیح دهید.

(ج) یک تابع اکتشافی دیگر ارائه دهید که قابل قبول باشد اما یک‌نوا نباشد. توضیح دهید چرا قابل قبول است ولی یک‌نوا نیست. (برای این بخش می‌توانید فرض کنید ارتفاع برج حداقل ۱۵ است.)