

به نام خدا



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر

مبانی و کاربردهای هوش مصنوعی ترم بهار ۱۴۰۱

پاسخنامه تمرین اول

سوال ۱ (۱۵ نمره)

پاسخنامه سوال ۱:

۱) پهلوهای گویایی آمازون

P: رسانندگی بستری سال ۱۴۰۱ به مقصد

E: مسیر پرواز و محل گزینش یا تحویل دادن بسته

A: موقعیتهای پرواز - بازوی بلند کردن یا پایین گذاشتن بسته

S: GPS - سنسور تشخیص موقعیت - سنسور تشخیص ارتفاع

۲) ربات بازرگین تسلا

P: تعداد بردها - امتیازات مسابقه

E: زمین تسلا - توپ - ربات - دوربین

A: بازوی حرکت ربات - موقع حرکت ربات

S: سنسور تشخیص جای توپ (دوربین) - سرعت سنسور

۳) توصیف ربات محمول

P: میزان استهلاک از محمول توصیف شده

E: data collection - اطلاعات کاربر و خریدهای قبلی

A: نشان دادن محمول به کاربر

S: تحلیل ترس سابقه کاربر

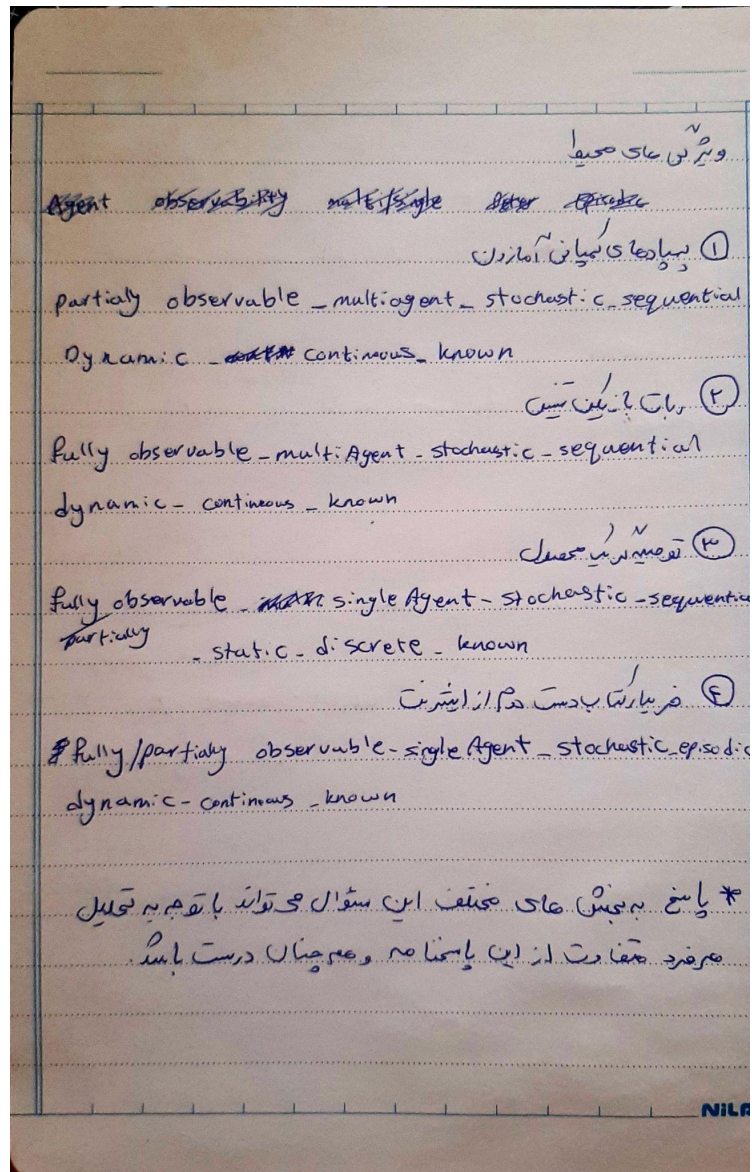
۴) فریدارک کتاب های دست دوم از اینترنت

P: پیدا کردن (خرید) کتاب با قیمت مناسب

E: بازارهای آنلاین کتاب

A: خرید (فروش) کتاب

S: تغییرات قیمت کتاب - ورودی کاربر



سوال ۲ (۱۵ نمره)

الف) برای مشخص کردن فضای حالت و توصیف حالت‌ها نیاز است تا روشی پیدا کنیم که در آن بتوانیم موقعیت مکانی n عامل Pacman و همچنین وجود یا عدم وجود k نقطه را نشان دهیم. به این منظور می‌توان لیستی از زوج مرتب‌های (x_1, y_1) تا (x_n, y_n) را برای مشخص کردن مکان عامل‌ها در هر حالت در نظر گرفت. برای وجود یا عدم وجود k نقطه نیز می‌توان یک لیست کتایی boolean (یا 0 و 1) در نظر گرفت. پس در نهایت هر حالت را می‌توان با دو لیست به شکل زیر مدل کرد:

$$(x_1, y_1, x_2, y_2, \dots, x_n, y_n), (b_1, b_2, b_3, \dots, b_k)$$

دامنه x_i و y_i ها:

$$0 \leq x_i < x$$

$$0 \leq y_i < y$$

(ب)

$$\text{Maximum state space} = (x \ y)^n \times 2^k$$

(ج) در حالتی که هر عامل برای حرکت 4 جهت برای انتخاب داشته باشد، branching factor ماکسیمم می‌شود.

$$\text{Maximum branching factor} = 4^n$$

(د)

زمانی که تمام نقاط خورده شوند هدف ما است. پس با توجه به توصیف حالتی که در بخش الف ارائه دادیم، حالت هدف به شکل زیر خواهد بود:

$$(x_1, y_1, x_2, y_2, \dots, x_n, y_n), (\text{False}, \text{False}, \text{False}, \dots, \text{False})$$

و آزمون هدف به صورت چک کردن لیست boolean و False بودن تمام مقادیر آن تعریف می‌شود.

(ه)

چون همه‌ی عامل‌ها همزمان action انجام می‌دهند، یک هیوریستیک قابل قبول می‌تواند به صورت زیر تعریف شود:

$$h = \frac{\text{تعداد نقاط باقی مانده}}{n}$$

توجه: شیوه توصیف حالت این مساله می‌تواند راه‌حل‌های متفاوتی داشته باشد که با توجه به آن ممکن است جواب برخی قسمت‌ها نیز متفاوت شود. در صورتی که مدل‌سازی و تحلیل راه حل شما قابل قبول باشد نمره بخش‌های مربوطه به شما داده می‌شود.

سوال ۳ (۱۵ نمره)

الف) درستی و نادرستی عبارات زیر را مشخص کنید و در صورت نادرستی هر عبارت، دلیل آن را به طور کامل توضیح دهید.

1. ماکزیمم دو هیوریستیک قابل قبول همواره قابل قبول است.
2. جست‌وجوی درختی A^* با هر تابع هیوریستیک دلخواه، کامل است.
3. محیط کاری وجود دارد که در آن هیچ عامل کاملاً واکنش‌گری نمی‌تواند منطقی (عقلانی) رفتار کند.

4. ورودی برنامه‌ی یک عامل همان ورودی تابع عامل است.
 5. فرض کنید یک عامل، عمل خود را به طور تصادفی و یکنواخت از میان مجموعه اعمال ممکن انتخاب می‌کند. در این صورت محیط کار قطعی ای وجود دارد که این عامل در آن منطقی باشد.
 6. عاملی وجود دارد که در دو محیط کار مجزا، کاملاً منطقی باشد.
 7. در یک محیط غیر قابل مشاهده، هر عاملی منطقی است.
 8. یک عامل بازیکن poker منطقی، هرگز نمی‌بازد.
- ب) به سوالات زیر پاسخ کوتاه دهید.

1. آیا ممکن است بیش از یک برنامه‌ی عامل برای پیاده‌سازی یک تابع عامل وجود داشته باشد؟
2. بهترین عامل در شرایطی که اهداف متناقض وجود دارد و یا چندین هدف وجود دارد که با قطعیت قابل حصول نیستند کدام است؟

سؤال ۳۰ :

الف) از درست ~~درست~~ ۲ درست ۳ درست

۴. نادرست ۵. ورودی تابع عامل دنباله ای از State طاسست

ورودی برنامه ی عامل توسط Sensor ها دریافت می شود

۵. درست ۶. درست ۷. نادرست ۸. نادرست

عامل بدون دریافت اطلاعات از محیط جا
نیز باید رفتاری منطقی داشته باشد
↓
بازی شاسی است

ب)

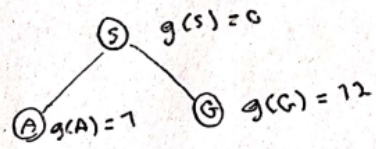
۱- بله، پیاده سازی های مختلفی برای هدف یک تابع عامل می تواند
وجود داشته باشد.

۲- عامل utility-based - زیرا هدف مفروضه ایست در هر صورت
utility تعریف می شود

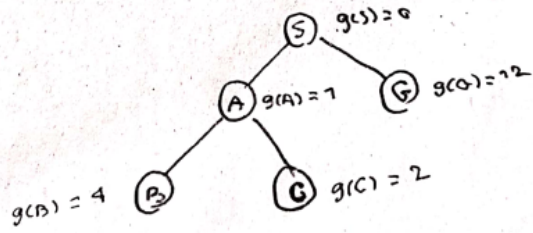
S

$$g(S) = 0$$

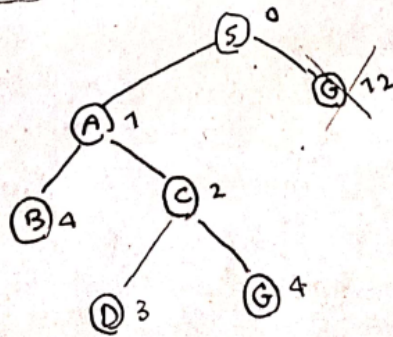
$S \rightarrow A$



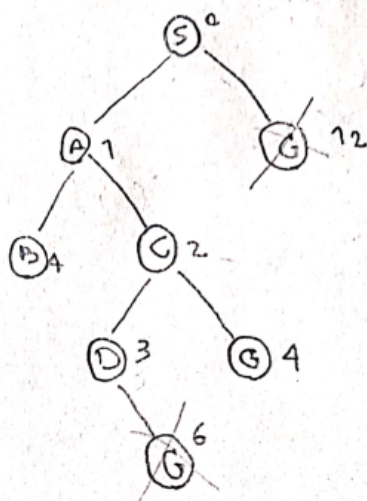
$S \rightarrow A \rightarrow C$



$S \rightarrow A \rightarrow C \rightarrow D$

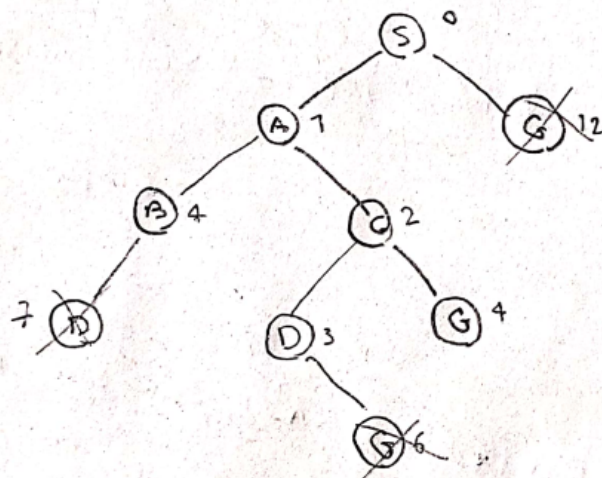


$S \rightarrow A \rightarrow B$



$S \rightarrow A \rightarrow C \rightarrow G$

هزینه 4



(د)

(د) چونکه مسئله غیر متناهی است، \bar{L}_n admissible و consistent است، \bar{L}_n قابل قبول است.

در مورد A^* جواب بهینه را برای هر دانه جواب صحیح $S \rightarrow A \rightarrow C \rightarrow G$ ضلع بود، چون کمترین

هزینه را دارد.

(ه)

خیر h_1 قابل پذیرش نیست دلیل: $h(S) > h^*(S)$

خیر h_1 قابل سازگاری نیست دلیل: $h(S) > c(S, A) + h(A)$

بله h_2 قابل پذیرش هست چون هزینه h برای هر گره کمتر از هزینه واقعی تا گره هدف است.

خیر $2h$ قابل سازگاری نیست دلیل: $h(S) > c(S, A) + h(A)$

سوال ۵ (۱۰ نمره)

N	State	g(N)	h(N)	f(N)	#exp
1	S	0	10	10	1
2	A	3	5	8	2
3	B	7	8	15	-
4	C	4	3	7	3
5	D	9	2	11	5
6	D	8	2	10	4
7	S	6	10	16	-
8	B	11	8	19	-
9	G1	14	0	14	6
10	B	12	8	20	-
11	G1	15	0	15	-

سوال ۶ (۱۰ نمره)

الف)

برای قابل قبول بودن، نباید هزینه مسیر نود D تا هدف، بیشتر از مقدار واقعی آن تخمین زده شود. هزینه واقعی رسیدن از نود D به هدف برابر با 3 است، پس برای قابل قبول بودن باید $0 \leq h(D) \leq 3$ باشد.
برای سازگار بودن باید روابط زیر برقرار باشد:

$$1) h(D) \leq c(D, a, E) + h(E) \rightarrow h(D) \leq 1 + h(E)$$

$$2) h(E) \leq c(E, a, F) + h(F) \rightarrow h(E) \leq 1 + h(F) \rightarrow h(E) \leq 1.5$$

به طرفین عبارت 2، یک واحد اضافه می‌کنیم:

$$3) 1 + h(E) \leq 1 + 1.5$$

با روابط 1 و 3 به نتیجه زیر می‌رسیم:

$$h(D) \leq 1 + 1.5 \rightarrow h(D) \leq 2.5$$

پس برای سازگار و قابل قبول بودن باید رابطه زیر برقرار باشد:

$$0 \leq h(D) \leq 2.5$$

ب)

برای اینکه گره E زودتر از گره B گسترش یابد، باید رابطه زیر برقرار باشد:

$$f(E) \leq f(B) \rightarrow 2 + h(E) \leq 2 + h(B) \rightarrow 2 + 1.1 \leq 2 + h(B) \rightarrow 1.1 \leq h(B)$$

برای این که گره B زودتر از گره F گسترش یابد، باید رابطه زیر برقرار باشد:

$$f(B) \leq f(F) \rightarrow 2 + h(B) \leq 3 + h(F) \rightarrow h(B) \leq 1 + h(F) \rightarrow h(B) \leq 1.5$$

با اجتماع دو رابطه بالا به رابطه زیر می‌رسیم:

$$1.1 \leq h(B) \leq 1.5$$