آزمون میان‌ترم

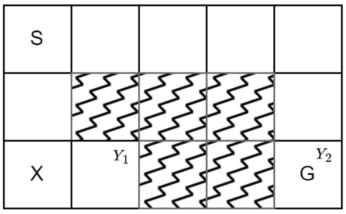
۱. (۲۰ نمره) به سوالات زیر به طور مختصر پاسخ دهید:

(آ) درست یا غلط بودن عبارت زیر را با ذکر دلیل مشخص کنید.
الگوریتم ژنتیک[۱] با اندازه‌ی جمعیت[۲] $N = ۱$ معادل Simulated Annealing با $T = \infty$ خواهد بود.

(ب) آیا k-consistency، (k-1)-consistency را نتیجه می‌دهد؟ در صورت مثبت بودن پاسخ، توضیح دهید در غیر اینصورت مثال نقضی ارائه کنید.

(ج) پیچیدگی زمانی Policy Iteration از $O(S^۲ A)$ و پیچیدگی زمانی Policy Evaluation از $O(S^۲)$ است.

(د) یک صفحه به صورت زیر در نظر بگیرید. شما در ابتدا در خانه‌ی سمت بالا چپ که با حرف S مشخص شده است قرار دارید و قصد دارید به خانه‌ی G برسید. برای رسیدن به هدف مجاز به حرکت به هر چهار طرف هستید. البته نمی‌توانید وارد خانه‌هایی که در آنها دیوار قرار دارد، شوید. عده‌ای از دوستان شما می‌خواهند تونلی حفر کنند تا شما بتوانید از زیر دیوارها عبور کنید. ورودی تونل خانه‌ی سمت چپ پایین است که با حرف X مشخص شده است. مشکل اینجاست که شما اطلاعی از اینکه حفر تونل به پایان رسیده است یا نه ندارید. در صورتی که تونل ناقص حفر شده باشد از خانه‌ی X می‌توان با یک حرکت به $Y_۱$ رفت. این در حالی است که اگر تونل به صورت کامل حفر شده باشد، از X می‌توان به $Y_۲$ که همان خانه‌ی G است، رفت. دقت داشته باشید که هزینه‌ی عبور از تونل مثل بقیه‌ی حرکات برابر با یک است.
حالا شما می‌خواهید یک نماینده از طرف خودتان را به صفحه‌ی داده شده بفرستید تا مسیر بهینه را به کمک الگوریتم A* پیدا کند. برای هدایت او لازم است تا یک تابع مکاشفه [۳] در اختیار او بگذارید. یک تابع مکاشفه‌ی admissible به جز تابع $h(x) = ۰$ ارائه دهید.



حل.

(آ) صحیح. وقتی اندازه جمعیت برابر با ۱ باشد، دو والد انتخاب شده نیز همان یک عضو خواهند بود. بنابراین نتیجه crossover نیز همان یک عضو خواهد شد. پس تنها بخشی که باعث تغییر کردن ژن می شود، mutation است که ژن را به صورت تصادفی تغییر می دهد. پس در این حالت این الگوریتم معادل walk random خواهد بود. همچنین در صورتی که در simulated annealing دما برابر بی نهایت باشد، هر حالت جدیدی پذیرفته خواهد شد. بنابراین این الگوریتم نیز معادل walk random خواهد بود.

(ب) خیر. برای مثال، مربع لاتین ۲×۲ را در نظر بگیرید. می‌دانیم این مسئله 4-consistent است زیرا با داشتن هر ۳ تایی مجاز خانه ۴ ام نیز بدست می‌آید اما با داشتن هر ۲ تایی مجاز لزوما خانه سوم مورد پذیرش نیست. حالتی را فرض کنید که قطر اصلی با ۱ و ۲ پر شده است. در این حالت دیگر امکان ادامه دادن وجود ندارد. به این نحو این مسئله 4-consistent است اما 3-consistent نیست.

(ج) نادرست. پیچیدگی زمانی Policy Iteration از $\mathcal{O}(S^۲ A + S^۳)$ است. و همینطور الگوریتم Policy Evaluation از پیچیدگی زمانی $\mathcal{O}(S^۳)$ میباشد.

(د) برای ارائه دادن یک تابع admissible می‌بایست کران پایینی برای هزینه تا حالت هدف ارائه دهیم. در مسائل مشابه می‌توان وجود دیوارها را relax کرد و فاصله‌ی منهتن را در نظر گرفت. اما نکته‌ای که در این سوال وجود دارد این است که اگر تونل به صورت کامل حفر شده باشد، این امکان وجود دارد که با یک حرکت از خانه‌ی X به خانه‌ی هدف برسیم. بنابراین تابع مکاشفه را به صورت زیر تعریف می‌کنیم.

---
[۱] Genetic Algorithm
[۲] Population
[۳] heuristic

$h(z) = min(Manhattan(z, X) + ۱, Manhattan(z, G))$

دقت کنید که از این که تونل حفر شده است یا نه اطلاعی نداریم. اما چون به دنبال کران پایین هستیم، حالت حفر شده را در نظر می‌گیریم.

۲. (۲۰ نمره) الگوریتم ژنتیک را در نظر بگیرید که در آن از کروموزوم‌هایی به فرم $x = ABCDEFGH$ با طول ثابت ۸ استفاده می‌کنیم. هر ژن می‌تواند عددی بین ۰ تا ۹ را بگیرد. تابع fitness را برای هر کروموزوم x به صورت زیر تعریف می‌کنیم:

$$f(x) = A + (2 \times B) + (3 \times C) - D - E - (2 \times F) + G - H$$

جمعیت اولیه نیز از ۴ کروموزوم به شکل زیر تشکیل شده است:

$$x1 = 24513892, x2 = 13562893, x3 = 43213205, x4 = 12903621$$

(آ) در این قسمت ابتدا fitness چهار نمونه زیر را به دست آورده و سپس احتمال انتخاب آن‌ها برای دور بعد را به دست آورید.

(ب) حال عملیات crossover را بر روی دو نمونه با بیشترین fitness انجام دهید، به صورت تک نقطه‌ای که محل crossover نقطه‌ی میانی کروموزوم باشد. همچنین عملیات crossover را به صورت دو نقطه‌ای بر روی اولین و سومین کروموزوم از لحاظ بیشتر بودن fitness انجام دهید. نقاط C و F را به عنوان دو نقطه مورد نیاز در نظر بگیرید.

(روش این نوع crossover دو نقطه‌ای به این صورت است که $CDEF$ از یک کروموزوم و بقیه ژن‌ها از یک کروموزوم دیگر بدست می‌آید)

(ج) نتایج قسمت قبل را به عنوان جمعیت جدیدمان در نظر بگیرید. حال دوباره fitness را برای این جمعیت جدید حساب کنید. آیا مجموع fitnessها افزایش پیدا کرده است؟

(د) بهترین کروموزوم مسئله (بیشترین fitness را داشته باشد) را بدست آورید. مقدار fitness آن را نیز حساب کنید.

(ه) طبق جمعیت فعلی که داریم آیا بدون انجام عملیات mutation امکان رسیدن به جواب بهینه وجود دارد؟ توضیح دهید.

حل. داریم:

$$f(x) = A + (2 \times B) + (3 \times C) - D - E - (2 \times F) + G - H$$

(آ) تابع ftiness را برای ۴ نمونه حساب می‌کنیم.

$$f(x1) = 2 + (2 \times 4) + (3 \times 5) - 1 - 3 - (2 \times 8) + 9 - 2 = 12$$
$$f(x2) = 1 + (2 \times 3) + (3 \times 5) - 6 - 2 - (2 \times 8) + 9 - 3 = 4$$
$$f(x3) = 4 + (2 \times 3) + (3 \times 2) - 1 - 3 - (2 \times 2) + 0 - 5 = 3$$
$$f(x4) = 1 + (2 \times 2) + (3 \times 9) - 0 - 3 - (2 \times 6) + 2 - 1 = 18$$

$$p(x1) = \frac{12}{37}$$
$$p(x2) = \frac{4}{37}$$
$$p(x3) = \frac{3}{37}$$
$$p(x4) = \frac{18}{37}$$

(ب) کروموزوم‌های حاصل از crossover نقطه‌ای:

$$y1 = 24513621$$
$$y2 = 12903892$$

کروموزوم‌های حاصل از crossover دو نقطه‌ای:

$$y3 = 12562821$$
$$y4 = 13903693$$

(ج) تابع ftiness را برای ۴ نمونه‌ی جدید حساب می‌کنیم.

$$f(y1) = 2 + (2 \times 4) + (3 \times 5) - 1 - 3 - (2 \times 6) + 2 - 1 = 10$$
$$f(y2) = 1 + (2 \times 2) + (3 \times 9) - 0 - 3 - (2 \times 8) + 9 - 2 = 20$$
$$f(y3) = 1 + (2 \times 2) + (3 \times 5) - 6 - 2 - (2 \times 8) + 2 - 1 = -3$$
$$f(y4) = 1 + (2 \times 3) + (3 \times 9) - 0 - 3 - (2 \times 6) + 9 - 3 = 25$$

جمع fitness در حالت قبلی ۳۷ بود. الان ۵۲ است و در نتیجه مجموع fitnessها بیشتر شده است.

(د) برای بهترین کروموزوم داریم:

$$z = 99900090$$
$$f(z) = 9 + (2 \times 9) + (3 \times 9) - 0 - 0 - (2 \times 0) + 9 - 0 = 63$$

(ه) نه نمی‌توان رسید. به دلیل این که مثلا برای ژن اول (A) بدون mutation مقدار آن یکی از مقادیر ۱ یا ۲ یا ۴ می‌ماند و نمی‌توان آن را به ۹ تبدیل کرد.

۳. (۱۲ نمره) در رابطه با هر یک از مجموعه‌های زیر تعیین کنید که مجموعه موردنظر محدب می‌باشد یا خیر. توجه کنید که برای نشان دادن محدب بودن یک مجموعه بایستی اثبات کنید که مجموعه مفروض محدب است و برای نشان دادن عدم محدب بودن یک مجموعه کافی است که مثال نقضی برای آن بیاورید.

(آ) مجموعه C که برای زوج مرتب‌های شامل یک بردار x و عدد حقیقی t به شکل زیر تعریف می‌شود.

$$C = \{(x, t) \mid \ ||x|| \leq t\}$$

(ب) مجموعه C که شامل توابع احتمالاتی f می‌باشد. برای تعریف دقیق‌تر C خواهیم داشت:

$$C = \{(f \ | f \geq ۰, \int f = ۱\}$$

حل.

(آ) مجموعه مورد نظر یک مجموعه محدب می‌باشد.
برای اثبات این موضوع فرض می‌کنیم که دو زوج‌مرتب دلخواه (x, t) و (y, u) عضو مجموعه C باشند. در این صورت طبق تعریف مجموعه می‌دانیم که $t \geq ||x||$ و به شکل مشابه $u \geq ||y||$ می‌باشند. حال با استفاده از تعریف اندازه بردار خواهیم داشت:

$$\begin{cases} ||\lambda x|| = \lambda ||x|| \leq \lambda t \\ ||(۱-\lambda)y|| = (۱-\lambda)||y|| \leq (۱-\lambda)u \end{cases} \implies ||\lambda x|| + ||(۱-\lambda)y|| \leq \lambda t + (۱-\lambda)u$$

برای آن که ثابت کنیم مجموعه موردنظر محدب می‌باشد، کافی است نشان دهیم که زوج مرتب $(\lambda x + (۱-\lambda)y, \lambda t + (۱-\lambda)u)$ عضو این مجموعه است. از آن‌جا که زوج‌مرتب‌هایی که ابتدا انتخاب کردیم دلخواه بودند، بنابراین کافی است گزاره گفته شده را اثبات کنیم تا ثابت شود مجموعه C یک مجموعه محدب می‌باشد. بنابراین داریم:

$$||\lambda x + (۱-\lambda)y|| \leq ||\lambda x|| + ||(۱-\lambda)y|| \leq \lambda t + (۱-\lambda)u$$
$$\implies ||\lambda x + (۱-\lambda)y|| \leq \lambda t + (۱-\lambda)u \implies (\lambda x + (۱-\lambda)y, \lambda t + (۱-\lambda)u) \in C$$

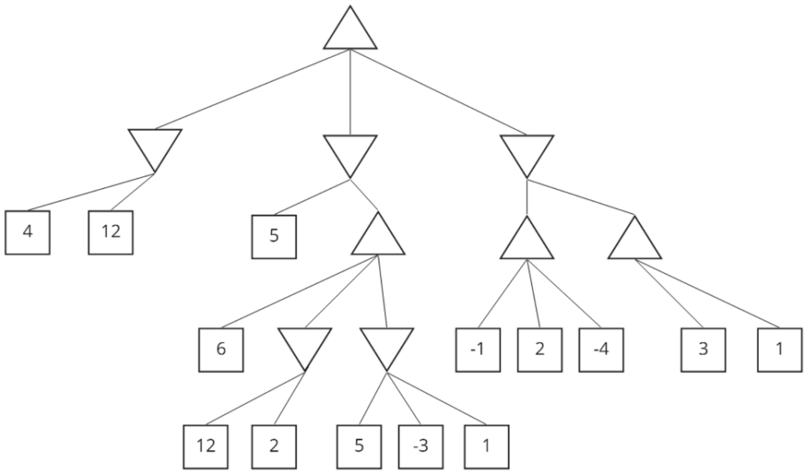
بنابراین مجموعه C یک مجموعه محدب می‌باشد.

(ب) مجموعه موردنظر یک مجموعه محدب می‌باشد.
برای اثبات این موضوع فرض می‌کنیم که دو تابع احتمالاتی دلخواه f و g عضو مجموعه C هستند. حال کافی است نشان دهیم که به ازای هر $\theta \in [۰, ۱]$ تابع $\theta f + (۱-\theta)g$ نیز عضو مجموعه C می‌باشد. بنابراین داریم:

$$\int \theta f + (۱-\theta)g = \theta \int f + (۱-\theta)\int g = \theta + (۱-\theta) = ۱$$

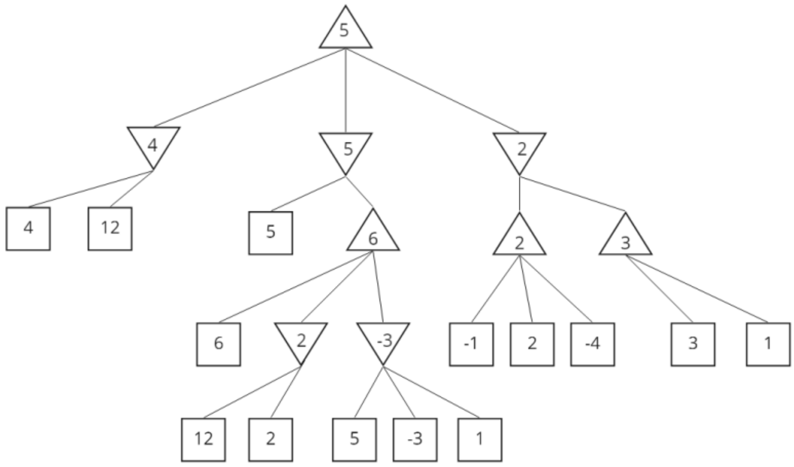
بنابراین $\theta f + (۱-\theta)g \in C$ می‌باشد و بنابراین مجموعه C یک مجموعه محدب خواهد بود.

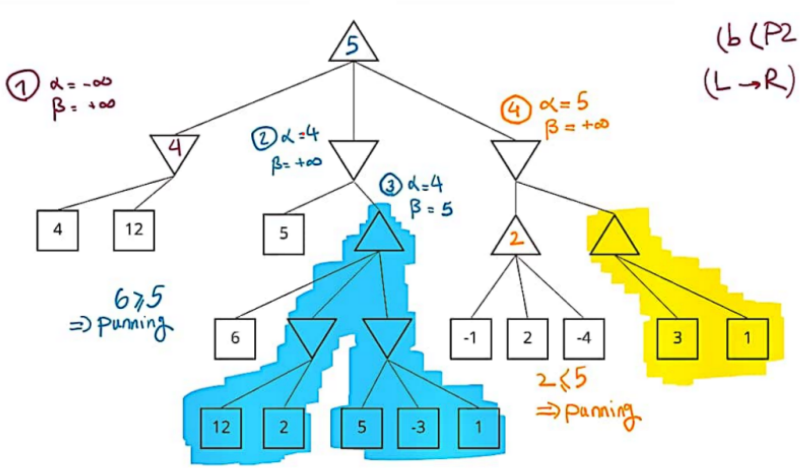
۴. (۱۰ نمره) درخت minimax زیر را در نظر بگیرید و با توجه به آن به دو بخش زیر پاسخ دهید.



- مقادیر تمام گره‌های min و max را پر کنید. مثلث‌های رو به بالا گره‌های بیشینه هستند، در حالی که مثلث‌های رو به پایین گره‌های کمینه هستند.
- هرس آلفا-بتا را با فرض گسترش از چپ به راست انجام دهید.

حل.

- 

- 

۵. (۱۲ نمره) مسئله رمزگذاری زیر را در نظر بگیرید:

```
  T W O
+ T W O
-------
F O U R
```

هر حرف انگلیسی نمایانگر یک رقم متمایز است. می‌خواهیم مقداردهی به این حروف را به گونه‌ای انجام دهیم تا رابطه بالا برقرار باشد. به عبارتی حاصل جمعی که در بالا با جایگذاری اعداد بجای حروف بدست می‌آید، جایگذاری صحیحی باشد. همچنین استفاده از رقم ۰ در شرایطی که تعداد ارقام اعداد را کاهش دهد مجاز نیست. (F و T نمی‌توانند ۰ باشند.)

(آ) مسئله را به صورت یک مسئله CSP مدل کنید و گراف قیود آن را رسم کنید.

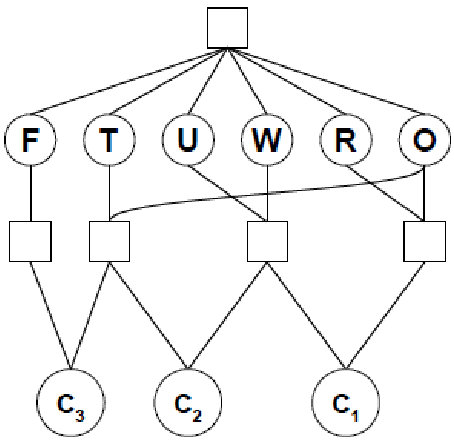
(ب) با استفاده از هیوریستیک‌های MRV و LCV این مسئله را حل کنید و درخت نهایی را رسم کنید.

حل.

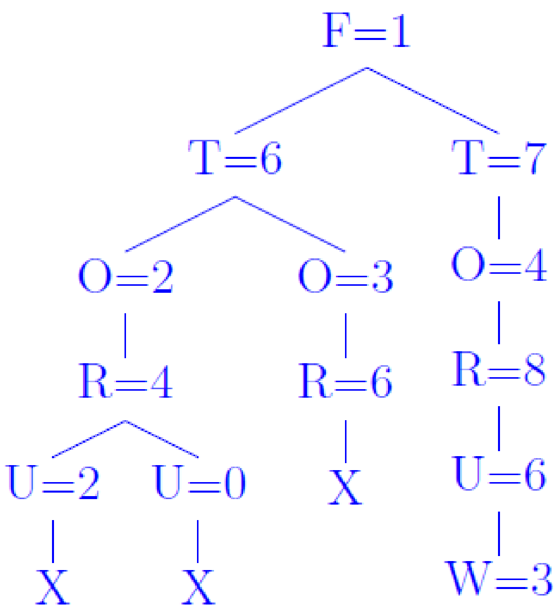
(آ) مسئله به صورت زیر مدل شده است:

Variables: {F, T, U, W, R, O, C1, C2, C3}
Domain of (U, W, R, O): {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
Domain of (F, T): {1, 2, 3, 4, 5, 6, 7, 8, 9}
Domain of (C1, C2, C3): {0, 1}
Constraints:

- O + O = R + 10 * C1
- C1 + W + W = U + 10 * C2
- C2 + T + T = O + 10 * C3
- F = C3
- F, T, U, W, R, O are unique and not equal to eachother.



(ب) درخت نهایی در زیر آورده شده است:



۶. (۱۵ نمره) فرض کنید حسین MDP ساده $M = (S, A, R, P, \gamma)$ با $|S| < \infty$ و $|A| < \infty$ و $0 < \gamma < 1$ را دارد.
حال یک MDP جدید $\hat{M} = (S, A, \hat{R}, P, \gamma)$ به طوریکه $\hat{R} - R = \epsilon$ را فرض کنید.
درواقع در این سوال می خواهیم بررسی کنیم آیا اگر به تمامی پاداش ها مقدار ϵ اضافه شود تابع ارزش و سیاست بهینه چه تغییری می کنند؟ ثابت کنید که $\hat{V}^* - V^* = \frac{\epsilon}{(1-\gamma)}$
آیا سیاست بهینه تغییر میکند؟
حل.

$$V^*(s) = max_a\{R(s,a) + \gamma \sum_{s'} P(s'|s,a)V^*(s')\}$$

$$V^*(s) + \frac{\epsilon}{1-\gamma} = max_a\{R(s,a) + \epsilon + \gamma \sum_{s'} P(s'|s,a)(V^*(s') + \frac{\epsilon}{1-\gamma})\}$$

$$V^*(s) + \frac{\epsilon}{1-\gamma} = max_a\{\hat{R}(s,a) + \gamma \sum_{s'} P(s'|s,a)(V^*(s') + \frac{\epsilon}{1-\gamma})\}$$

$$\rightarrow \hat{V}^*(s) = V^*(s) + \frac{\epsilon}{1-\gamma}$$

سیاست بهینه نیز تغییری نمی کند زیرا ترتیب (اردرینگ) تابع پاداش حفظ می شود. می توان این را به صورت تئوری نوشت. کافی است تعریف سیاست بهینه را بنویسید و گام های اثبات کاملا مشابه است که هدف سوال نیست و همین توضیح نیز کافی است.

۷. (۱۶ نمره) حسین از شاگردان خوب در رشته خودش است اما در تصمیم‌گیری برای آینده تحصیلی‌اش دچار سردرگمی شده و مرتبا درحال عوض کردن تصمیم‌هایش است. او با پریدن بین فیلدها به دنبال فیلد مورد علاقه خودش می‌گردد. فرض کنید حسین بین ۵ فیلد Hardware, RL, Information Theory, Robotics, Quantum در حال جابجایی است. با توجه به شناختی که از حسین داریم می‌دانیم که بعد از ورود به هر فیلد میزان جایزه‌ای که می‌گیرد (معلوم نیست از چه کسی) چقدر است. شما می‌توانید مقدار این جایزه‌ها را به ازای فیلدهای مختلف در جدول زیر مشاهده کنید:
از آنجایی که حسین کمی هم حواس پرت است، بعضی وقت‌ها فیلدها را با هم قاطی می‌کند. جدول زیر نشان می‌دهد اگر حسین قصد تغییر فیلد بدهد با چه احتمالی به کدام فیلدها می‌پرد.:
حال فرض کنید حسین در یک سال اخیر با سیاست زیر قصد داشته بین فیلدها جابجا شود:

Field	Reward
Hardware	-1
RL	+1
Robotics	+1
Quantum	+2

Start State	Action	Probability	End State
Hardware	ToML	0.6	Robotics
Hardware	ToML	0.4	RL
Hardware	ToPhysics	1	Quantum
RL	ToPhysics	0.2	Hardware
RL	ToPhysics	0.8	Robotics
Robotics	ToHardware	0.5	Hardware
Robotics	ToHardware	0.5	Robotics
Robotics	ToPhysics	0.8	Quantum
Robotics	ToPhysics	0.2	RL
Quantum	ToML	1	RL
Quantum	ToHardware	1	Hardware

$$\pi_0(\text{Hardware}) = \text{ToPhysics}$$
$$\pi_0(\text{Robotics}) = \text{ToHardware}$$
$$\pi_0(\text{RL}) = \text{ToPhysics}$$
$$\pi_0(\text{Quantum}) = \text{ToML}$$

(آ) تابع ارزش سیاست اخیر حسین را بدست آورید. لازم نیست مقادیر نهایی را بدست آورید و صرف نوشتن معادله‌ها کافی است. (بر حسب γ)

(ب) با استفاده از یک گام اجرای الگوریتم Policy Iteration با فرض

$$V_Q > V_{RL} > V_{Ro} > V_{Hw}$$

به حسین کمک کنید تا بتواند راحت‌تر فیلد خودش را پیدا کند.

حل.

(آ)

$$V^{\pi_\cdot}(HW) = \gamma V^{\pi_\cdot}(Q) + ۲$$
$$V^{\pi_\cdot}(RL) = ۰/۲\gamma V^{\pi_\cdot}(HW) + ۰/۸\gamma V^{\pi_\cdot}(RO) + ۰/۶$$
$$V^{\pi_\cdot}(RO) = ۰/۵\gamma V^{\pi_\cdot}(HW) + ۰/۵\gamma V^{\pi_\cdot}(RO)$$
$$V^{\pi_\cdot}(Q) = \gamma V^{\pi_\cdot}(RL) + ۱$$

(ب) حال با فرض داده شده در هر حالت به ازای تمامی حرکت ها مقدار تابع ارزش را حساب میکنیم و به ازای آنها بزرگترین را برای هر حالت انتخاب میکنیم.

$$Hw \quad : \quad \begin{cases} \gamma(۰/۶V_{RO} + ۰/۴V_{RL}) + ۱ & \text{if} \quad a = \text{ToML} \\ \gamma(V_Q) + ۲ & \text{if} \quad a = \text{ToPhysics} \end{cases}$$

در این مورد با توجه به فرض صورت سوال داریم

$$\gamma(V_Q) \geq \gamma(۰/۶V_{RO} + ۰/۴V_{RL})$$

پس در این قسمت آپدیت به صورت $\pi(Hw) = ToPhysics$ تغییر میکند.

$$RL \quad : \quad \begin{cases} \gamma(۰/۲V_{Hw} + ۰/۸V_{RO}) + ۰/۶ & \text{if} \quad a = \text{ToPhysics} \end{cases}$$

پس در این قسمت آپدیت به صورت $\pi(RL) = ToPhysics$ تغییر میکند.

$$RO \quad : \quad \begin{cases} \gamma(۰/۵V_{RO} + ۰/۵V_{HW}) & \text{if} \quad a = \text{ToHardware} \\ \gamma(۰/۸V_Q + ۰/۲V_{RL}) + ۱/۸ & \text{if} \quad a = \text{ToPhysics} \end{cases}$$

در این مورد با توجه به فرض صورت سوال داریم

$$\gamma(۰/۸V_Q + ۰/۲V_{RL}) + ۱/۸ \geq \gamma(۰/۵V_{RO} + ۰/۵V_{HW})$$

پس در این قسمت آپدیت به صورت $\pi(RO) = ToPhysics$ تغییر میکند.

$$Q \quad : \quad \begin{cases} \gamma(V_{RL}) + ۱ & \text{if} \quad a = \text{ToML} \\ \gamma(V_{Hw}) - ۱ & \text{if} \quad a = \text{ToHW} \end{cases}$$

در این مورد با توجه به فرض صورت سوال داریم

$$\gamma(V_{RL}) + ۱ \geq \gamma(V_{Hw}) - ۱$$

پس در این قسمت آپدیت به صورت $\pi(RL) = ToML$ تغییر میکند.