دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

نظریه‌ی زبان‌ها و ماشین‌ها

نمونه‌سوال‌ها

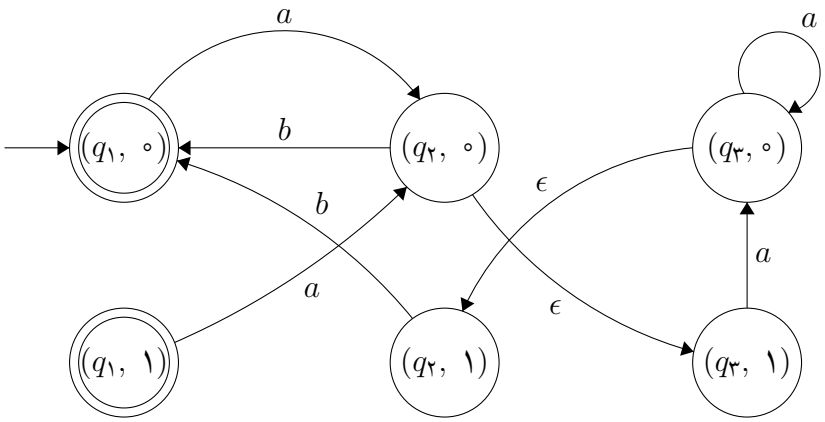
پاسخ‌نامه‌ی مجموعه‌ی ۸

همه‌ی مباحث زبان‌های منظم

نیم‌سال اول - سال تحصیلی ۰۴-۰۳

۱. **راه حل.** تفاوت این ماشین با ماشین ارائه شده در سوال، این است که این ماشین حداکثر دارای$k - ۱$ حرکت $\epsilon - transition$ متوالی است، و زبان ماشین نیز همان رشته‌های قابل قبول برای ماشین اولیه هستند که برای پذیرش آن‌ها حداکثر $k - ۱$ حرکت اپسیلون متوالی نیاز داشته باشیم. توجه کنید که شمارش اپسیلون ها با دیدن کاراکتر غیر اپسیلون، از نو شروع می شود.

همچنین توجه کنید که چون هر یال اپسیلون رو به پایین است، در نتیجه هیچ دوری وجود ندارد که تمام یال‌هایش اپسیلون باشد.



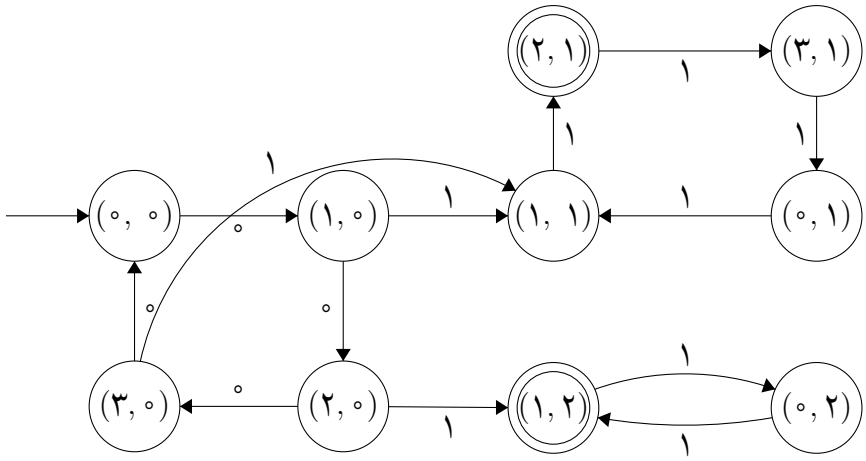
■

---

۲. الف) **راه حل.**

لازم است که یکی از n و m فرد باشد و دیگری به پیمانه ۴ با ۲ هم‌نهشت باشد. ماشین زیر، زبانی با این شرایط را تشخیص می‌دهد. لازم به ذکر است که در این ماشین تمام ۱۶ حالت ممکن برای m و n به پیمانه ۴ نوشته نشده و تنها حالت‌هایی که با استفاده از آن می‌توان به state های پذیرش رفت نوشته شده است.

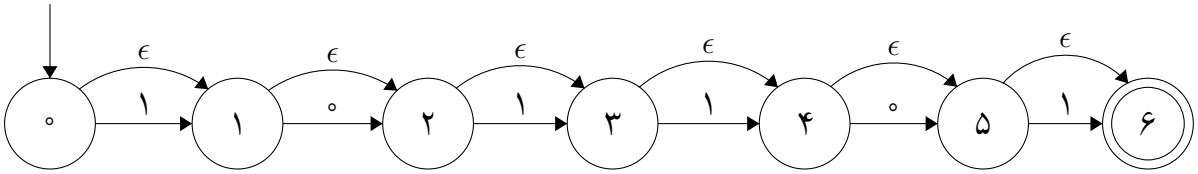
ابتدا با ورودی گرفتن ۰ بین ۴ حالت (۰،۰) و (۱،۰) و (۲،۰) و (۳،۰) حرکت می‌کنیم که تعداد ۰ها به پیمانه ۴ را ذخیره می‌کنند. اگر تعداد ۰ها به پیمانه ۴ برابر ۲ بود یعنی تعداد یک‌ها باید فرد باشد پس در این حالت تنها نیاز داریم که زوجیت تعداد یک‌ها را بررسی کنیم که دو حالت (۱،۲) و (۰،۲) را به آن اختصاص دادیم. ولی اگر تعداد صفرها فرد باشه پیمانه ۴ تعداد یک‌ها اهمیت پیدا می‌کند و ۴ حالت دیگر برای بررسی آن‌ها می‌سازیم.



■

ب) **راه حل.**

برای این زبان یک NFA رسم می‌کنیم. در این NFA با شروع از حالت ۰ در هر مرحله یا با یکی از کاراکترهای رشته ۱۰۱۱۰۱ و یا با ϵ به حالت بعدی می‌رویم. پس در نهایت رشته‌های پذیرفته شده زیررشته‌های ۱۰۱۱۰۱ خواهند بود.

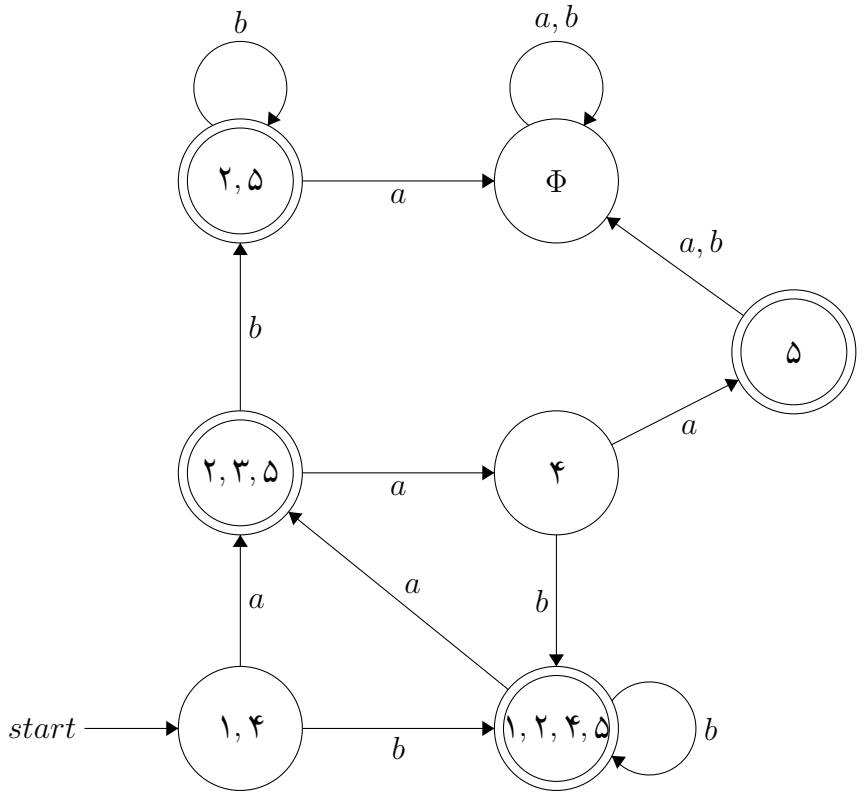


■

---

۳. **راه حل.**

با توجه به الگوریتم تبدیل *NFA* به *DFA*، این ماشین حالت متناهی غیرقطعی با *DFA* زیر معادل خواهد بود . توجه کنید که یک حالت Trap نیز تعریف شده است که به ورود به آن دیگر به حالت accept نخواهیم رفت.

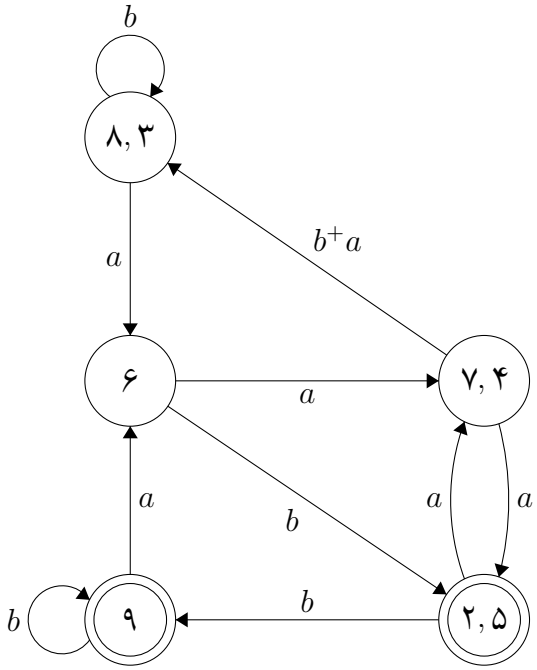


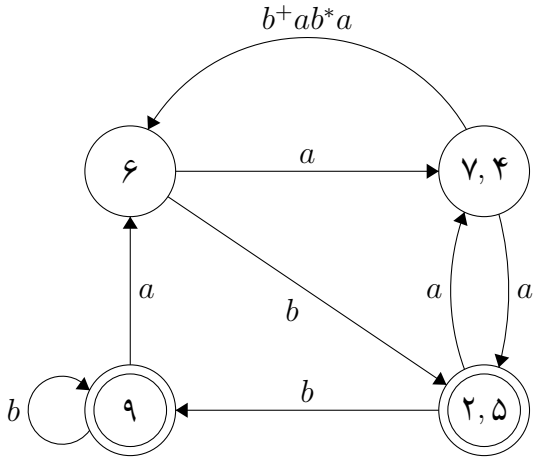
■

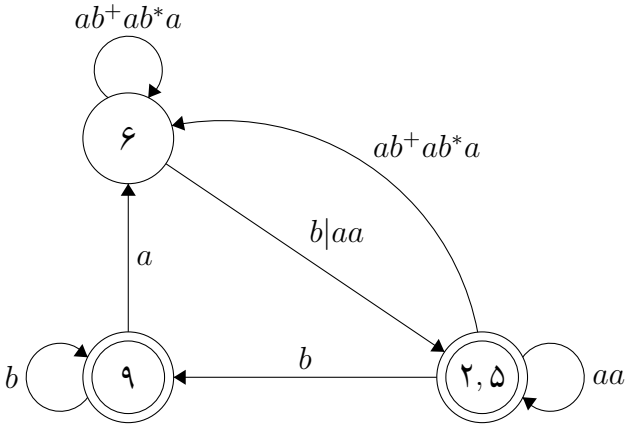
---

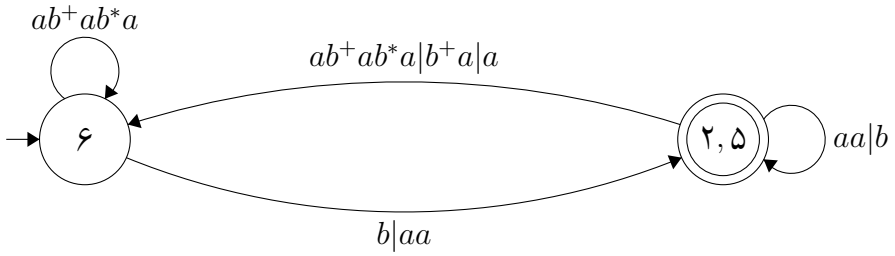
۴. راه حل.

الف) ابتدا بخش ب سوال را حل می‌کنیم تا با توجه به ماشین ساده شده زبان ماشین را بدست آوریم.









پس رجکس نهایی می‌شود: $R_۱ = (ab^+ab^*a)^*(b|aa)$

$R_{final} = R_۱(aa|b)^*((ab^+ab^*a|b^+a|a)R_۱(aa|b)^*)^*$

ب) پس از ساده سازی به ماشین متناهی کمینه زیر می‌رسیم:

یک روش برای اینکار این است که در هر مرحله مجموعه‌ها را افراز کنیم و اینکار را تا مرحله ای تکرار کنیم که دیگر مجموعه‌های باقی‌مانده تغییری نداشته باشند، حال باید یک مجموعه برای شروع افراز و یک پروتکل برای افراز کردن داشته باشیم، مجموعه شروع را دو مجموعه F و S درنظر بگیرید که F مجموعه استیت‌های نهایی است و S دیگر استیت‌ها، در هر مرحله باید بررسی کنید اعضای مجموعه با اکشن های موجود به کدام یک از مجموعه های فعلی می‌روند، بدین صورت که برای استیت‌هایی که در پاسخ تمام کنش‌ها به یک مجموعه یکسان می‌روند یک مجموعه تشکیل بدهید، برای فهم بهتر این موضوع به حل سوال می‌پردازیم:

$F = \{۲, ۵, ۹\}, S = \{۱, ۳, ۴, ۶, ۷, ۸, ۹, ۱۰\}$ استیت ۰ به دلیل دسترس پذیر نبودن حذف شد.

از اینجا به بعد زوج مرتب به شکل $x = (y, z)$ یعنی استیت x با اکشن a به مجموعه y می‌رود و با اکشن b به مجموعه z می‌رود:

$۲ = (S, F), ۵ = (S, F), ۹ = (S, F)$

$۱ = (S, S), ۳ = (S, S), ۴ = (F, S), ۶ = (S, F), ۷ = (F, S), ۸ = (S, S), ۱۰ = (S, S)$

پس S افراز می‌شود به ۳ مجموعه که در اولی تمام استیت هایی قرار می‌گیرند که به صورت $x = (S, S)$ بودند و در دومی استیت هایی قرار می‌گیرند که به صورت $x = (F, S)$ بوده‌اند و در سومی نیز $x = (S, F)$ ها قرار می‌گیرند حالت دیگری نیز نداریم:

$S_۱ = \{۱, ۳, ۸, ۱۰\}, S_۲ = \{۴, ۷\}, S_۳ = \{۶\}$

دوباره برای تمام استیت ها چک می‌کنیم:

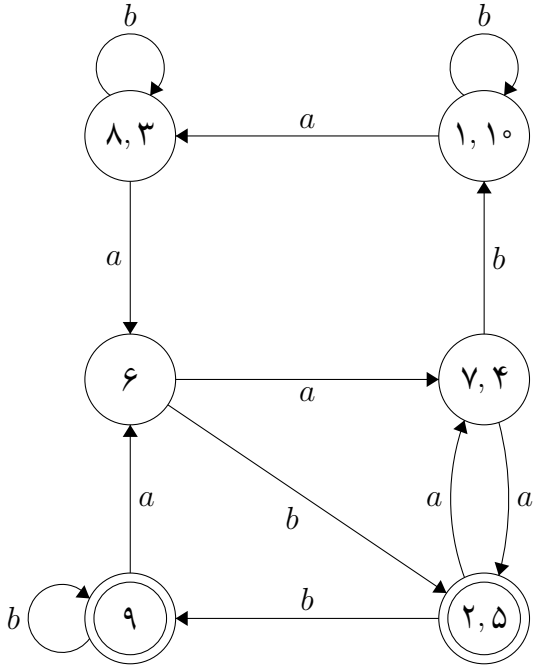
$۲ = (S_۲, F), ۵ = (S_۲, F), ۹ = (S_۳, F)$

$۱ = (S_۱, S_۱), ۳ = (S_۳, S_۱), ۴ = (F, S_۱), ۶ = (S_۲, F), ۷ = (F, S_۱), ۸ = (S_۳, S_۱), ۱۰ = (S_۱, S_۱)$

پس از افراز داریم:

$S_۴ = \{۱, ۱۰\}, S_۵ = \{۳, ۸\}, S_۶ = \{۴, ۷\}, S_۷ = \{۶\}\ F_۱ = \{۲, ۵\}, F_۲ = \{۹\}$

می‌توانید چک کنید که افراز بالا دیگر تغییری ندارد.

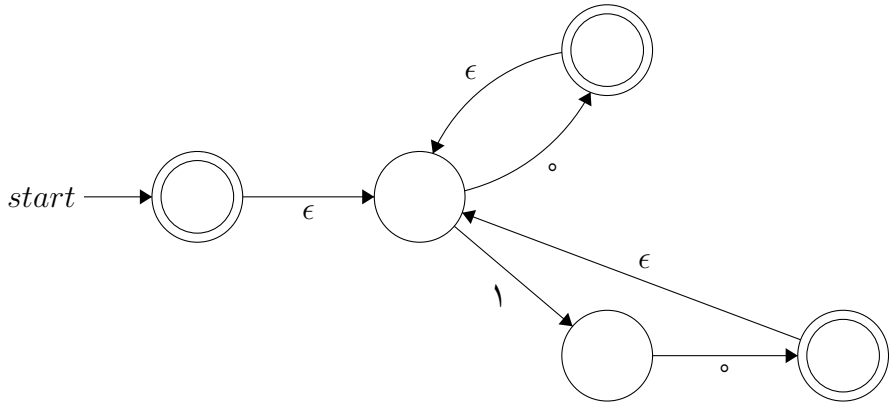


■

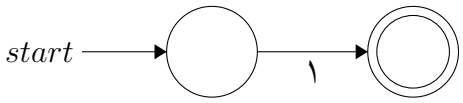
---

۵. **راه حل.**

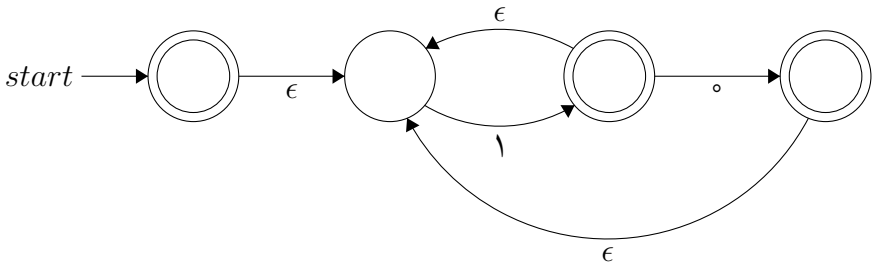
ماشین زیر، با عبارت $(۰ \cup ۱۰)^*$ معادل است



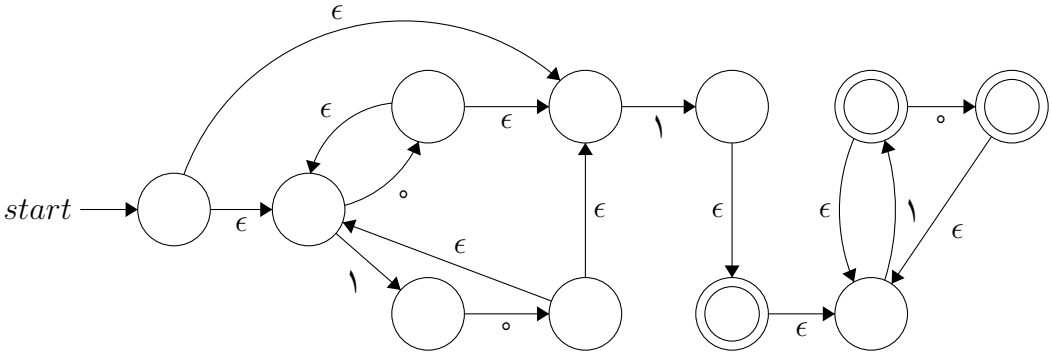
ماشین زیر، با عبارت ۱ معادل است



ماشین زیر، با عبارت *(۱۰ ∪ ۱) معادل است



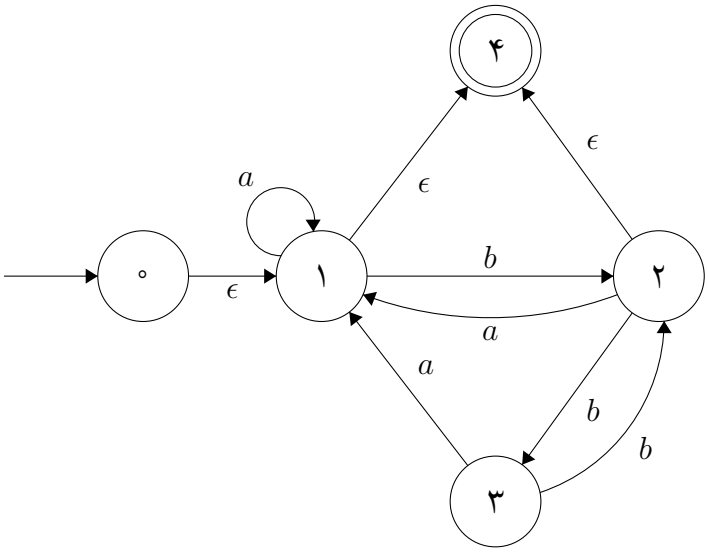
حالا ماشین معادل این عبارت منظم، از الحاق این سه ماشین به صورت زیر به دست می‌آید



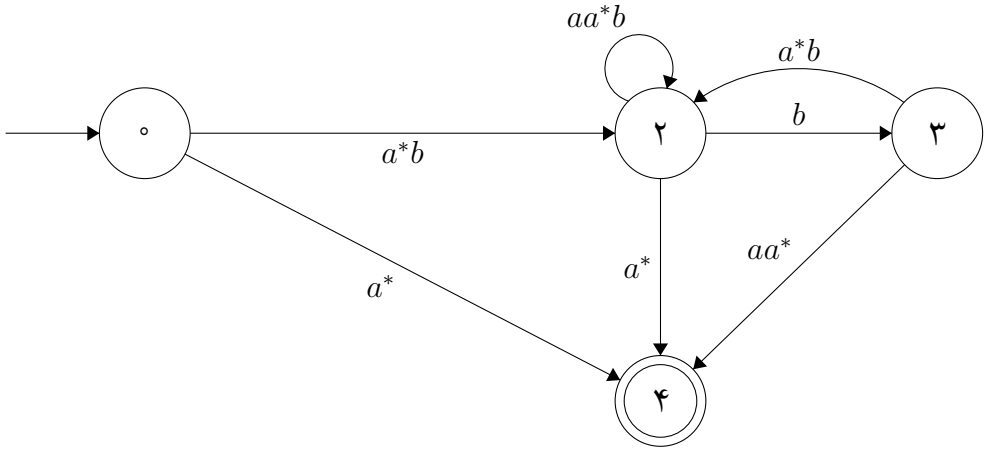
■

---

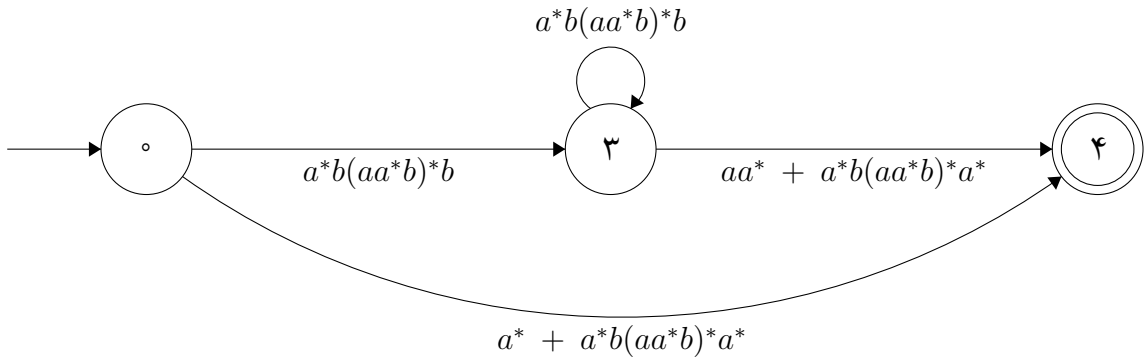
۶. **راه حل.** ابتدا استیت شروع و پایان را اضافه می‌کنیم.



سپس یکی یکی شروع به حذف استیت‌های میانی می‌کنیم. با حذف استیت ۱ به ماشین زیر می‌رسیم:



سپس استیت ۲ را حذف می‌کنیم:



و در نهایت با حذف استیت ۳ به عبارت منظم زیر می‌رسیم:

$$a^* + a^*b(aa^*b)^*a^* + a^*b(aa^*b)^*b(a^*b(aa^*b)^*b)^*(aa^* + a^*b(aa^*b)^*a^*)$$

■

---

۷. **راه حل.**

الف) این عملگر روی زبان‌های منظم بسته نیست. برای مثال، زبان منظم زیر را در نظر بگیرید:

$$L = (a \cup b)^*$$

با اعمال این عملگر روی این زبان، به زبان زیر می‌رسیم:

$$copythenreverse(L) = \{ww^R | w \in \Sigma^*\}$$

همانطور که می‌دانیم، این زبان نامنظم است. به عنوان مثال کافیست لم پمپاژ را روی آن با رشته‌ی a^pbba^p اعمال کنیم. از آن‌جا که داریم $|xy| \leq p$ ، همواره رشته‌ای به فرم $a^n (n \leq p)$ پمپاژ می‌شود، و کافیست با قرار دادن $i = 0$ و خارج کردن عبارت $y = a^n$ از رشته، کاری کنیم که رشته‌ی حاصل دیگر به فرم ww^R نباشد.

ب) این زبان لزوما منظم نیست. زبان a^*bc^* را در نظر بگیرید. رشته‌ای از این زبان به فرمت a^ibc^j به طوری که $i+j+۱=۳n$ باشد. حال سه حال زیر ممکن است:

۱) $x=a^n$ و $z=c^n$: که در این صورت بدیهتا xz عضور زبان L' خواهد بود

۲) در حالتی که i کوچکتر از n باشد داریم $x=a^ibc^{n-i-۱}$ و $z=c^n$: در این صورت xz به صورت $a^ibc^{۲n-i-۱}$ خواهد بود که $i+(۲n-i-۱)$ فرد است و این رشته عضو زبان L' است.

۳) در حالتی که j کوچکتر از n باشد نیز همانند حالت دو داریم: $a^{۲n-j-۱}bc^j$ عضو زبان L' خواهد بود

بنابراین زبان L' به صورت زیر خواهد بود:

$$L'=\{a^nb^n|n>۰\}\cup\{a^mbc^n|n+m\,٪\,۲=۱\}$$

که این زبان منظم نیست. برای اثبات نامنظم بودن این زبان کافی است که از لم پمپاژ استفاده کنیم و رشته a^pb^p را در نظر بگیریم. چون داریم که $|xy|<=p$ پس $y=a^m$ پس بعد از پمپ کردن رشته حاصل به شکل a^tb^p خواهد بود که عضو زبان نیست.

■

---

۸. **راه حل.**

اگر p(x)=cx+d باشد در آن صورت L منظم است: اگر c،d هم علامت باشند، $|p(n)|=|c|n+|d|$ و عبارت منظم آن $a^{|d|}(a^{|c|})^*$ می‌باشد. اگر هم‌علامت نباشند در واقع چند جمله ای باید در یک ناحیه محدود یک علامت داشته باشد و در یک ناحیه نامحدود علامت دیگری داشته باشد.

تمام رشته‌های قابل پذیرش در زبان L را که در آن ناحیه محدود تولید می‌شوند می‌توانیم بنویسیم(تعدادشان محدود است) حال کمترین $n=n_۰$ که از آن به بعد چند جمله‌ای تغییر علامت می‌دهد را در نظر بگیرید. عبارت منظم آن عبارت است از: $a^{|p(n_۰)|}(a^{|c|})^*$ که با اجتماع آن با رشته‌های با تعداد محدود که در ناحیه دیگر ساختیم عبارت منظم آن شکل می‌گیرد. در نتیجه در هر حالت از c و d زبان L دارای یک عبارت منظم است در نتیجه منظم است.
طبق لم پمپاژ به ازای p داریم:

$$a^n\in L; n>p=>xy^iz=a^{n+(i-۱)|y|}\in L$$

در نتیجه اختلاف تعداد a بین هر دو عدد متوالی $n_۱,n_۲$ برای $p(n_۱),p(n_۲)$ حداکثر می‌تواند |y| باشد. حال می‌دانیم به ازای هر چندجمله‌ای درجه $m>۱$ مثل $p(m)=a_mx^m+...+a_۰$ داریم:

$$lim_{n\Longrightarrow\infty}\Delta p(n)=|p(n+۱)|-|p(n)|=\infty>|y|$$

و به تناقض می‌رسیم. در نتیجه فرض خلف باطل و چندجمله‌ای حتما به صورت p(x)=cx+d می‌باشد. ■

---

۹. **راه حل.**

الف) این زبان منظم نیست. اثبات با برهان خلف. فرض می‌کنیم این زبان منظم است. از لم پمپاژ استفاده می‌کنیم. به ازای یک n داده شده، رشته زیر در زبان است:

$$(۲^n, ۲^{n+۱} - ۱, ۲^{۲n+۱} - ۲^n)$$

این رشته به شکل زیر است:

$$\begin{pmatrix} ۰ \\ ۰ \\ ۱ \end{pmatrix}^n \begin{pmatrix} ۱ \\ ۱ \\ ۱ \end{pmatrix} \begin{pmatrix} ۰ \\ ۱ \\ ۰ \end{pmatrix}^n \tag{۱}$$

از لم پمپاژ استفاده می‌کنیم. اگر قرار دهیم n=p طبق لم پمپاژ نتیجه می‌گیریم که y جزوی از قسمت

$$\begin{pmatrix} ۰ \\ ۰ \\ ۱ \end{pmatrix} \tag{۲}$$

است. پس رشته $xy^۲z$ می‌شود:

$$\begin{pmatrix} ۰ \\ ۰ \\ ۱ \end{pmatrix}^{n+|y|} \begin{pmatrix} ۱ \\ ۱ \\ ۱ \end{pmatrix} \begin{pmatrix} ۰ \\ ۱ \\ ۰ \end{pmatrix}^n \in L \longrightarrow ۱^{n+|y|+۱}۰^{n+۱} = ۰^{n+|y|}۱۰^{n+۱} * ۰^{n+|y|+۱}۱^{n+۱} \tag{۳}$$

اما می‌دانیم که:

$$۱^{n+|y|+۱}۰^{n+۱} > ۰^y۱^{n+۱}۰^{n+۱} = ۲^{n+۱}(۲^{n+۱} - ۱) = ۰^{n+|y|}۱۰^{n+۱} * ۰^{n+|y|+۱}۱^{n+۱} \tag{۴}$$

که با تساوی قبلی در تناقض است. تناقض حاصله نشان می‌دهد که فرضی که کردیم غلط بود و این زبان منظم نیست.

ب) با استفاده از برهان خلف، نشان می دهیم که L یک زبان منظم نمی باشد. فرض کنید L منظم است. در این صورت یک ماشین حالت متناهی قطعی وجود دارد که این زبان را بپذیرد. فرض کنید این ماشین حالت متناهی قطعی، دارای n حالت باشد. حال رشته s را به شکل زیر در نظر بگیرید:

$$\begin{pmatrix} ۱ \\ ۰ \\ ۰ \end{pmatrix}^n \begin{pmatrix} ۱ \\ ۱ \\ ۰ \end{pmatrix} \begin{pmatrix} ۰ \\ ۰ \\ ۱ \end{pmatrix}^{n+۱} \in L \tag{۵}$$

$$\frac{۱^{n+۱}۰^{n+۱}}{۰^{n}۱^{n+۱}} = \frac{۲^{n+۱}(۲^{n+۱}-۱)}{۲^{n+۱}} = ۲^{n+۱}-۱ = ۰^{n+۱}۱^{n+۱} \tag{۶}$$

مشابه قسمت قبل اگر از لم پمپاژ استفاده کنیم نتیجه می‌گیریم که y جزوی از

$$\begin{pmatrix} ۱ \\ ۰ \\ ۰ \end{pmatrix} \tag{۷}$$

است. پس بعد از یکبار پمپاژ رشته ما می‌شود:

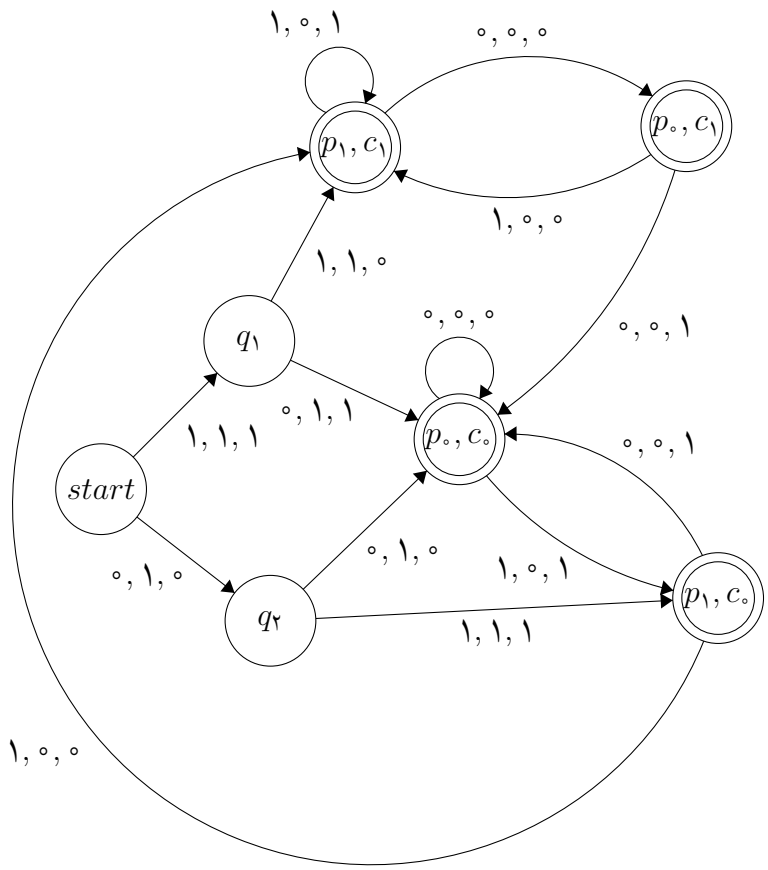
$$\begin{pmatrix} ۱ \\ ۰ \\ ۰ \end{pmatrix}^{n+|y|} \begin{pmatrix} ۱ \\ ۱ \\ ۰ \end{pmatrix} \begin{pmatrix} ۰ \\ ۰ \\ ۱ \end{pmatrix}^{n+۱} \in L \longrightarrow \frac{۱^{n+|y|+۱}۰^{n+۱}}{۰^{n+|y|}۱۰^{n+۱}} = ۰^{n+|y|+۱}۱^{n+۱} \tag{۸}$$

اما می‌دانیم که

$$\frac{۱^{n+|y|+۱}۰^{n+۱}}{۰^{n+|y|}۱۰^{n+۱}} >= \frac{۰^{|y|-۱}۱^{n+۲}۰^{n+۱}}{۰^{n+|y|}۱۰^{n+۱}} = \frac{۲^{n+۱}(۲^{n+۲}-۱)}{۲^{n+۱}} = ۲^{n+۲}-۱ > ۲^{n+۱}-۱ = ۰^{n+|y|+۱}۱^{n+۱} \tag{۹}$$

که با تساوی قبل در تناقض است. تناقض حاصل نشان می‌دهد که فرض خلف باطل بوده و زبان منظم نیست.

پ) این زبان منظم است و برای اثبات این موضوع از اثبات ساختاری با ساخت یک NFA استفاده می‌کنیم. تحلیل زبان به این صورت است: اگر عدد باینری n را در ۳ ضرب کنیم انگار مقدار $N + ۲N =$ $N + shiftLeft(N, ۱)$ را محاسبه می‌کنیم. در نتیجه اگر رشته‌ای که داریم را reverse کنیم، برای بدست آوردن s[i] یا بیت i ام ضرب، به مقادیر C[i] و $N[i-۱]$ و N[i] نیاز داریم که در اینجا C[i] همان بیت carry i ام است. ماشین متناهی M را می‌سازیم تا reverse(L) را بسازد. واضح است که با تغییر این ماشین می‌توان به ماشینی رسید که خود L را بشناسد و L منظم خواهد بود. با توجه به این توضیحات، پس از دو کاراکتر اول که در مقدار دوم tuple شان مقدار ۱ دارند به ۴ استیت می‌رسیم که آن‌ها را با مقدار carry و previousBit مشخص کرده‌ایم. بین این استیت‌ها با tuple های $(۱, ۰, X)$ و $(۰, ۰, X)$ حرکت می‌کنیم که مقدار X از مقدار c و p استیت قبلی و مقدار اول tuple بدست می‌اید. هر فلشی هم که در این NFA کشیده نشده یعنی نمی‌تواند به استیت accept برسد.



■

---

۱۰. **راه حل.**

الف) یک صف و یک مجموعه خالی M را در نظر می‌گیریم. ابتدا $q_۰$ را در صف قرار می‌دهیم. به این صورت تا زمانی که صف خالی نشده است، اولین عضو صف را خارج می‌کنیم و به مجموعه M اضافه می‌کنیم. سپس به ترتیب حروف الفبا به ازای هر حرف اگر حالت $\delta(q, \alpha) = s$ موجود باشد و در مجموعه M موجود نباشد، آن حالت را به انتهای صف اضافه می‌کنیم. از آنجا که تعداد حالات متناهی است و به واسطه مجموعه M مارک می‌شوند، الگوریتم پایان پذیر است. نهایتا اگر $M \cap F$ تهی نبود نتیجه می شود که زبان ماشین تهی نیست.

ب) طبق تعریف، رشته‌ی w یک رشته عمومی برای استیت‌های p و q است اگر و فقط اگر استیت r ای وجود داشته باشد که داشته باشیم که $\delta(p, w) = \delta(q, w) = r$. برای هر استیت r عضو Q ماشین‌های حالت متناهی زیر را تعریف می‌کنیم:

$$A_{p,r} = (Q, \Sigma, \delta, p, r), A_{q,r} = (Q, \Sigma, \delta, q, r) \qquad (۱۰)$$

با این تعریف رشته w یک رشته عمومی برای p, q است اگر و فقط اگر w عضو اشتراک زبان‌های $A_{p,r}$ و $A_{q,r}$ برای حداقل یک r عضو Q باشد. برای هر r فرض می‌کنیم که $A_{p,q,r}$ ماشین حالت متناهی سازنده اشتراک دو زبان $A_{p,r}$ و $A_{q,r}$ باشد.(می‌دانیم که اشتراک دو زبان منظم منظم است و پیشتر با نحوه ساخت ماشین حالت متناهی سازنده اشتراک دو ماشین حالت متناهی دیگر آشنا شدید.)
حال کافی است که برای هر $r \in Q$ بررسی کنیم که زبان $A_{p,q,r}$ تهی است یا خیر و برای اینکار می‌توانیم

از الگوریتم قسمت الف استفاده کنیم. اگر حتی یکی از این زبان‌ها تهی نبود یعنی رشته عمومی برای مجموعه $\{p, q\}$ وجود دارد.

پ) طرف راست عبارت یعنی وجود یک رشته عمومی به ازای هر زوج حالت، در صورت وجود رشته عمومی برای Q بدیهی است.

حال طرف دوم را ثابت می‌کنیم.

فرض کنید برای هر مجموعه دو عضوی از استیت‌ها مثل $\{p, q\}$ رشته عمومی $w_{p,q}$ وجود داشته باشد.

حال با استقرا بر روی سایز مجموعه S ثابت می‌کنیم که برای هر $S \in Q$ یک رشته عمومی وجود دارد.

واضح است که در صورت اثبات این ادعا می‌توان نتیجه گرفت که مجموعه Q نیز رشته عمومی خواهد داشت.

برای پایه استقرا در حالت $|S| = ۱$ رشته ϵ رشته عمومی است. حال برای گام استقرا فرض کنید که برای هر مجموعه S با تعداد اعضای کمتر مساوی i حکم را ثابت کرده‌ایم. می‌خواهیم آن را برای یک مجموعه $i + ۱$ عضوی مثل $S = \{p_۱, p_۲, ..., p_{i+۱}\}$ ثابت کنیم.

طبق فرض می‌دانیم که دو استیت $p_۱$ و $p_۲$ یک رشته عمومی مثل $w_{p۱,p۲}$ دارند. حال مجموعه زیر را در نظر بگیرید:

$$S' = \{\delta(p_۱, w_{p۱,p۲}), \delta(p_۲, w_{p۱,p۲}), ..., p_{i+۱}, w_{p۱,p۲})\} \tag{۱۱}$$

می‌دانیم که $\delta(p_۱, w_{p۱,p۲}) = \delta(p_۲, w_{p۱,p۲})$ پس مجموعه S' یک مجموعه i عضوی از استیت‌های ماشین حالت متناهی است. پس طبق فرض استقرا یک رشته عمومی مثل $w_{S'}$ دارد. پس به راحتی می‌توان دید که رشته $w_{p۱,p۲}w_{S'}$ یک رشته عمومی برای S خواهد بود.

■

---

۱۱. راه حل.

بخش a) برای اثبات باید ابتدا ثابت کنیم که برای هر X که عضوی از $A(BA)^*$ باشد، حتما X عضوی از $(AB)^*A$ خواهد بود و سپس برعکس آن را اثبات کنیم. برای مورد اول داریم:

$$\exists \quad \omega \in A, \omega_1 \in BA, \omega_2 \in BA, ..., \omega_n \in BA \quad s.t. \quad X = \omega(\omega_1\omega_2...\omega_n)$$

$$= \omega(\omega_{1B}\omega_{1A}\omega_{2B}\omega_{2A}...\omega_{nB}\omega_{nA})$$

حال اگر ترتیب ترکیب‌بندی‌ها کلمات را عوض کنیم خواهیم داشت:

$$X = (\omega\omega_{1B})(\omega_{1A}\omega_{2B})...(\omega_{(n-1)A}\omega_{nB})\omega_{nA}$$

از آنجایی که داریم:

$$\omega\omega_{1B} \in AB, \omega_{1A}\omega_{2B} \in AB, ..., \omega_{(n-1)A}\omega_{nB} \in AB, \omega_{nA} \in A$$

لذا می‌توانیم نتیجه بگیریم که X عضو $(AB)^*A$ می‌باشد.

حال برای مورد دوم داریم:

$$\exists \quad \omega_1 \in AB, \omega_2 \in AB, ..., \omega_n \in AB, \omega \in A \quad s.t. \quad X = (\omega_1\omega_2...\omega_n)\omega$$

$$= (\omega_{1A}\omega_{1B}\omega_{2A}\omega_{2B}...\omega_{nA}\omega_{nB})\omega$$

مثل ایده مورد قبلی داریم:

$$X = \omega_{1A}(\omega_{1B}\omega_{2A})(\omega_{2B}\omega_{3A})...(\omega_{nB}\omega)$$

از آنجایی که داریم:

$$\omega_{1A} \in A, \omega_{1B}\omega_{2A} \in BA, \omega_{2B}\omega_{3A} \in BA, ..., \omega_{nB}\omega \in BA$$

لذا نتیجه گرفته می‌شود که X عضو $A(BA)^*$ می‌باشد.

بخش b) از قواعد حاکم بر رشته‌ها می‌دانیم که اگر a و b دو رشته بر روی یک الفبای مشخص باشند، خواهیم داشت:

$$(ab)^R = b^R a^R$$

حال فرض می‌کنیم که X عضوی از زبان $(AB)^R$ باشد، در این صورت می‌توانیم آن را به صورت $X = (ab)^R$ در نظر بگیریم که $a \in A, b \in B$ می‌باشند. طبق قاعده‌ای که بالاتر مطرح کردیم می‌توانیم نتیجه بگیریم که $X = b^R a^R$ می‌باشد و از آنجایی که $b \in B, a \in A$ می‌باشد، نتیجه می‌گیریم که X عضوی از $B^R A^R$ نیز می‌باشد. حال باید ثابت کنیم که هر X عضو $B^R A^R$ عضوی از $(AB)^R$ نیز می‌باشد که با توجه به اینکه قاعده‌ای که در مورد رشته‌ها گفتیم دو طرفه است، این رابطه هم به راحتی همانند حالت اول ثابت می‌شود.

لذا نشان دادیم که دو زبان طرح‌شده در سوال زیرمجموعه‌ی یک دیگر هستند و طبق نظریه مجموعه‌ها نتیجه می‌گیریم که دو زبان با یک دیگر مساوی هستند.

بخش c) برای اثبات همانند قبل عمل می‌کنیم. برای رسیدن از سمت چپ به راست اگر نشان دهیم که

$$\forall n \forall X \quad X \in (A^n)^R \Rightarrow X \in (A^R)^n$$

برقرار است، کار تمام است. برای این کار از استقراء استفاده می‌کنیم. برای گام شروع استقراء، یعنی $n = 0$ داریم:

$$X = (\epsilon)^R = \epsilon \Rightarrow X \in (A^R)^0 = \epsilon$$

حال برای اثبات در حالت کلی داریم:

$$X \in (A^n)^R \Rightarrow X \in (A^{n-1}A)^R \xrightarrow{(AB)^R = B^R A^R} X \in A^R (A^{n-1})^R$$

طبق فرض استقراء خواهیم داشت:

$$X \in (A)^R (A^R)^{n-1} = (A^R)^n$$

حال برای اینکه از سمت راست به سمت چپ برسیم مجددا از استقراء استفاده می‌کنیم. برای گام شروع استقراء، یعنی $n = 0$ همانند حالت قبلی به وضوح می‌بینیم که حکم برقرار است و به سراغ اثبات در حالت کلی می‌رویم:

$$X \in (A^R)^n \Rightarrow X \in (A^R)^{n-1} (A^R)^1$$

طبق فرض استقراء خواهیم داشت:

$$X \in (A^{n-1})^R (A^1)^R \xrightarrow{(AB)^R = B^R A^R} X \in (A^1 A^{n-1})^R = (A^n)^R$$