

به نام خدا

نظریه زبان‌ها و ماشین‌ها - پاییز ۱۴۰۱

تمرین شماره ۱

دستیار آموزشی این مجموعه: معین شیردل

moein.shirdel@ut.ac.ir

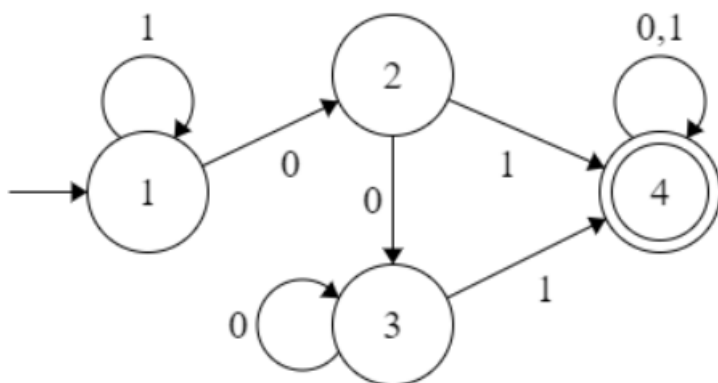
تاریخ تحویل: یکشنبه ۲۴ مهرماه ۱۴۰۱



(۱) برای زبان‌های زیر، DFA معادل را رسم کنید: (۲۰ نمره)

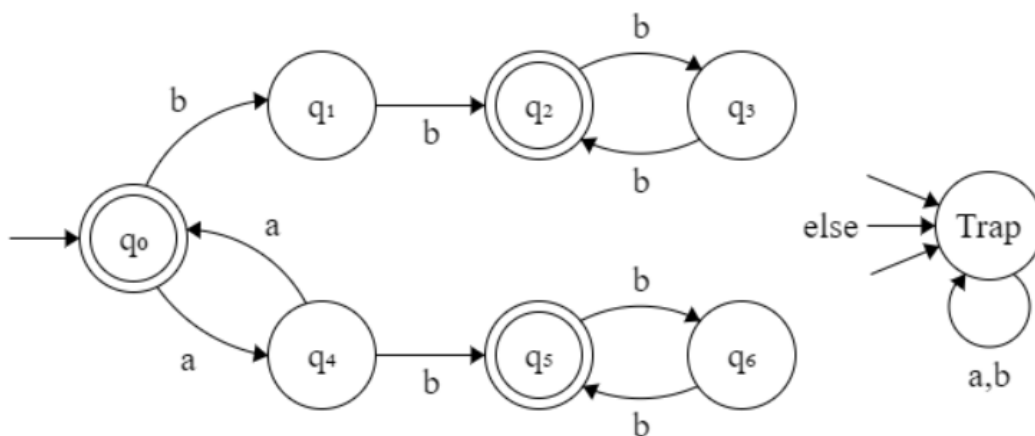
الف) $(\Sigma = \{0, 1\}) . L = \{w_1 0 1 w_2 \mid w_1, w_2 \in \{0, 1\}^*\}$

پاسخ:



ب) $(\Sigma = \{a, b\}) . L = \{a^i b^j \mid (i, j \geq 0) \text{ and } ((i+j) \bmod 2 = 0)\}$

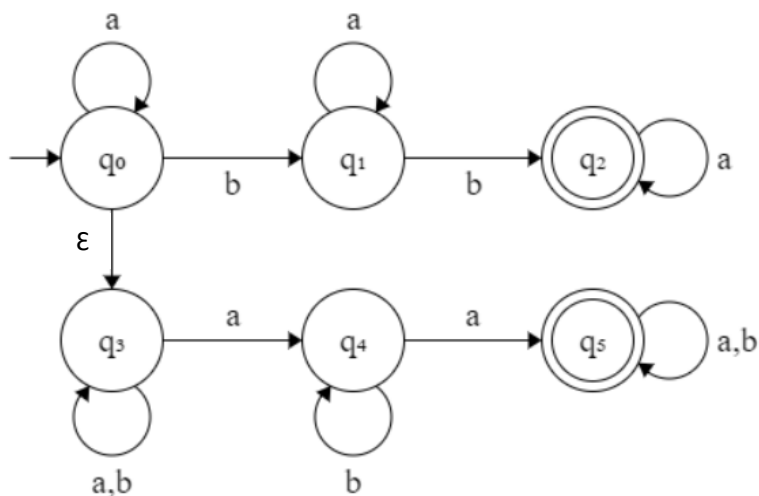
پاسخ:



۲) برای هریک از زبان های زیر، NFA مناسب را ترسیم کنید: (۲۰ نمره)

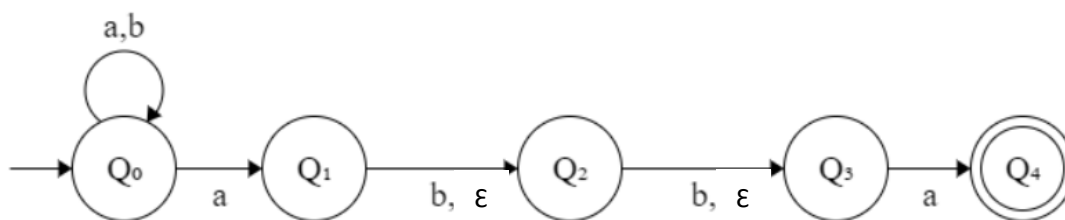
الف) $L = \{ w \in \{a,b\}^* \mid n_a(w) \geq 2 \text{ or } n_b(w) = 2 \}$ (با حداکثر ۶ استیت)

پاسخ:



ب) $L = \{ w \in \{a,b\}^* \mid w \text{ ends with "abba", "aba" or "aa"} \}$ (با حداکثر ۵ استیت)

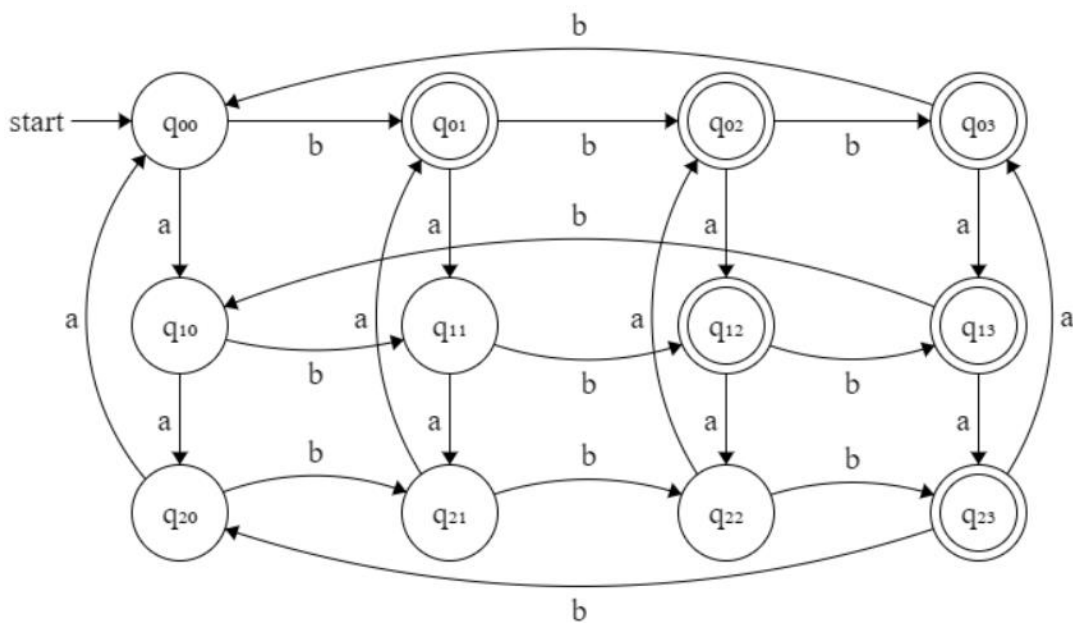
پاسخ:



۳) زبان L با الفبای $\Sigma = \{a, b\}$ را در نظر بگیرید. این زبان شامل رشته هایی است که باقی مانده ی تعداد a های رشته بر ۳، از باقی مانده ی تعداد b های آن رشته بر ۴ کمتر باشد. DFA معادل زبان L را رسم کنید.

(۱۵ نمره)

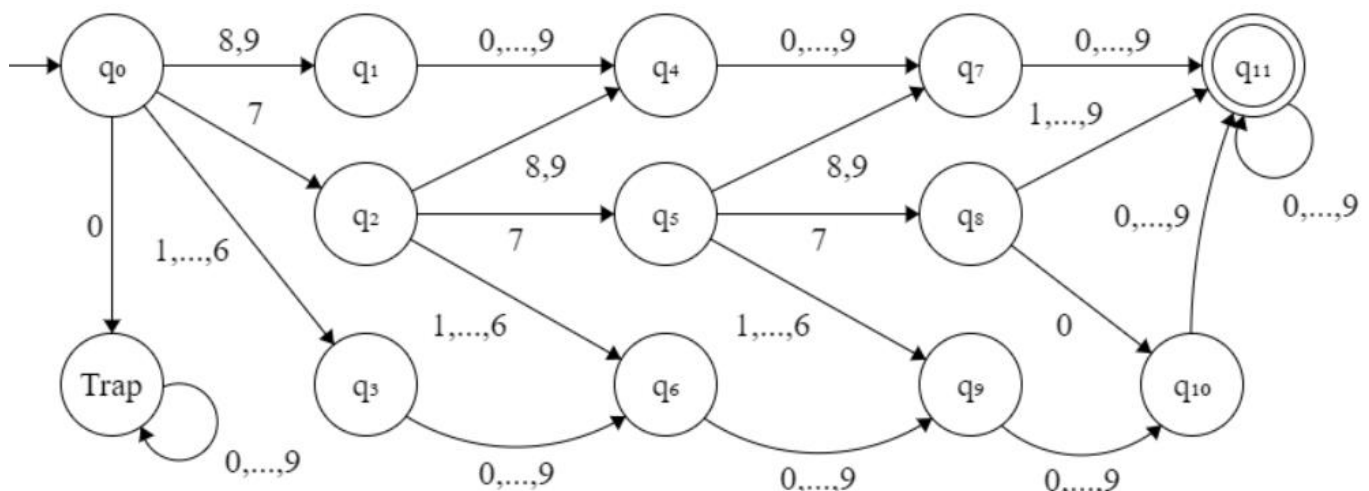
پاسخ:



* توجه کنید که ساختار فوق قابل تعمیم برای سوالاتی با اعداد متفاوت نیز هست. در هر استیت، اندیس آن استیت بیان گر باقی مانده‌ی تعداد a ها بر ۳ و b ها بر ۴ است. در هر ردیف از استیت‌ها، باقی مانده تعداد b ها بر ۴ محاسبه می‌شود و در هر ستون، باقی مانده‌ی تعداد a ها بر ۳. همچنین، accepting state ها نیز آن‌هایی هستند که اندیس اول آن‌ها از اندیس دومشان کمتر باشد.

۴) زبان **NUMBERS** با الفبای $\Sigma = \{0, 1, \dots, 9\}$ را در نظر بگیرید. این زبان شامل تمام رشته‌های نشانگر اعداد بزرگ‌تر از ۷۷۷۰ (و نه خود آن) است. برای این زبان، DFA مناسب را طراحی کنید. توجه داشته باشید که زبان قابلیت پذیرش اعداد با هر تعداد رقم را دارد، ولی رشته‌هایی که با رقم صفر در سمت چپ شروع می‌شوند باید در ابتدا **reject** شوند. (۱۵ نمره)

پاسخ:



۵) درستی یا نادرستی جملات زیر را با ارائه‌ی اثبات یا مثال نقض مناسب مشخص کنید. (۱۰ نمره)

الف) در یک NFA که زبان L را می‌پذیرد، با تبدیل $\text{non-accepting state}$ ها به accepting state و برعکس، به NFA پذیرنده‌ی زبان \bar{L} می‌رسیم.

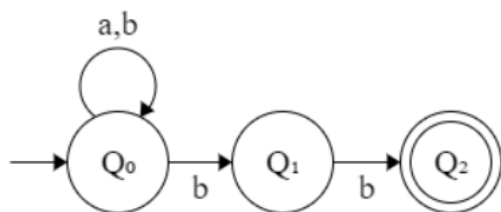
پاسخ:

نادرست. این گزاره برای DFA صادق است و نه برای NFA.

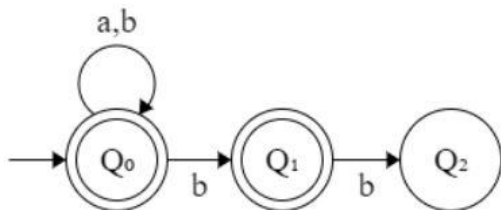
مثال نقض: زبان L را بدین صورت تعریف می‌کنیم.

$$L = \{ w \in \{a,b\}^* \mid w \text{ ends with "bb"} \}$$

بدین ترتیب، NFA زیر برای آن قابل تصور است:



که پس از تبدیل $\text{non-accepting state}$ ها به accepting state و برعکس، به NFA زیر می‌رسیم.



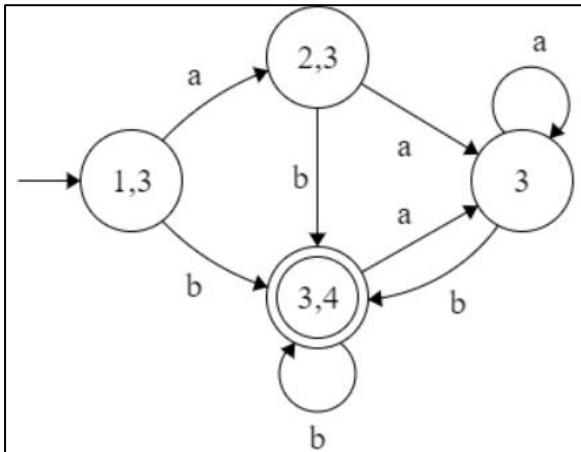
حال در این NFA تبدیل یافته، رشته abb قبول می‌شود در حالی که abb عضو \bar{L} نیست.

ب) می‌دانیم که M (که یک NFA است) زبان L را می‌پذیرد. در این صورت، حتماً یک DFA وجود دارد که \bar{L} را بپذیرد.

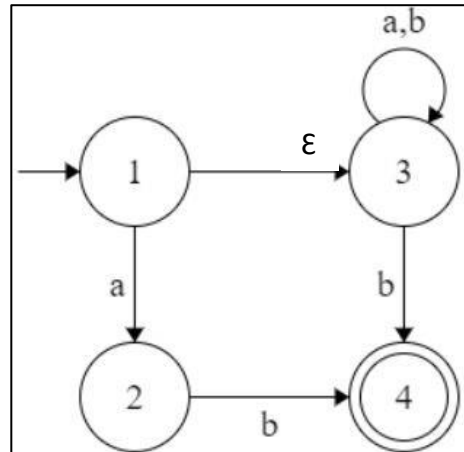
پاسخ:

درست. اگر یک NFA وجود داشته باشد که زبان L را بپذیرد، یعنی این زبان منظم است و حتماً یک DFA معادل وجود دارد که L را بپذیرد. زبان‌های منظم تحت عملگر متمم بسته هستند، زیرا با تبدیل accepting state ها به $\text{non-accepting state}$ و برعکس در DFA آن‌ها می‌توان به DFA مربوط به زبان متمم‌شان دست یافت. پس اگر زبان L منظم باشد برای آن یک DFA موجود است که از روی آن یک DFA برای \bar{L} قابل طراحی خواهد بود (و \bar{L} نیز منظم خواهد بود).

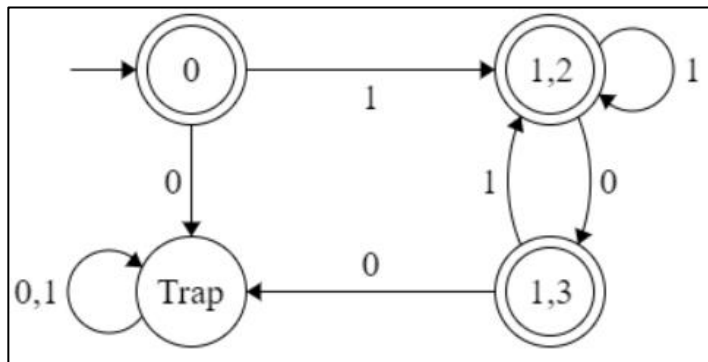
۶) برای هر یک از NFA های زیر، DFA معادل را ترسیم کنید. (۲۰ نمره)



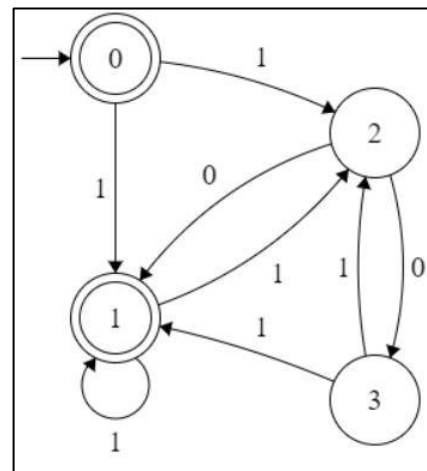
پاسخ:



(الف)



پاسخ:



(ب)

۷) L^R ، نشان دهنده‌ی زبانی است که با وارونه کردن تمامی رشته‌های زبان L به دست می‌آید.

ثابت کنید اگر L یک زبان منظم باشد، L^R نیز منظم است. (۱۰ نمره)

پاسخ:

اگر زبان L یک زبان منظم باشد، یک DFA وجود دارد که رشته‌های آن را بپذیرد. اگر در این DFA، جهت تمامی فلش‌ها را معکوس کنیم، initial state را به final state تبدیل کنیم و final state را به initial state، DFA حاصل زبان L^R را خواهد پذیرفت. در صورتی که در DFA ابتدایی چند تا final state وجود داشته باشد، می‌توان ابتدا یک final state جدید تعریف کرد و با تعداد ϵ -transition، final state ها را به آن متصل کرد و بعد عمل معکوس کردن initial state و final state ها و جهت فلش‌ها را انجام داد (برای جلوگیری از ایجاد چند initial state در استیت ماشین جدید). در چنین حالتی، یک NFA به وجود خواهد آمد. در هر دو حالت، یک استیت ماشین برای L^R یافته‌ایم. پس ثابت می‌شود که L^R نیز منظم است.