دانشگاه صنعتی شریف

دانشکده‌ی مهندسی کامپیوتر

درس نظریه‌ی زبان‌ها و ماشین‌ها

سوالات نمونه

پاسخ‌نامه‌ی مجموعه‌ی ۵ : زبان‌های منظم - بخش ۲

استاد: دکتر علی موقر

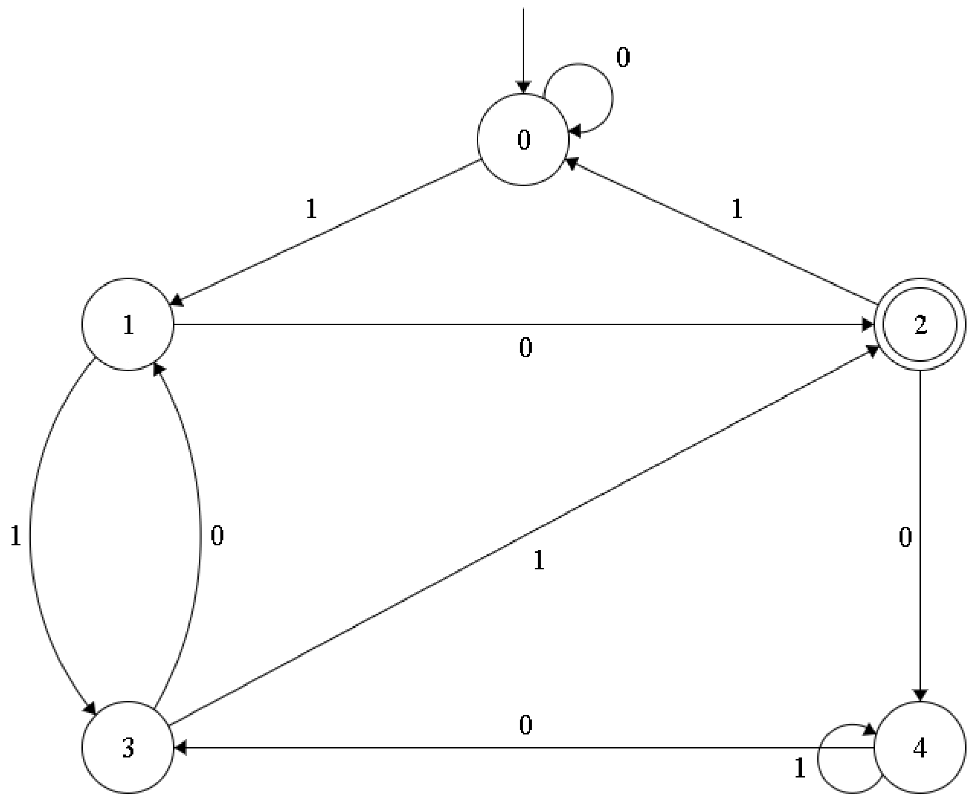
تیم دستیاران درس - نیم‌سال دوم ۰۲ - ۰۱

۱۰ اردیبهشت ۱۴۰۲

۱. مفاهیم ماشین‌های حالت متناهی

۱.۱

الف)

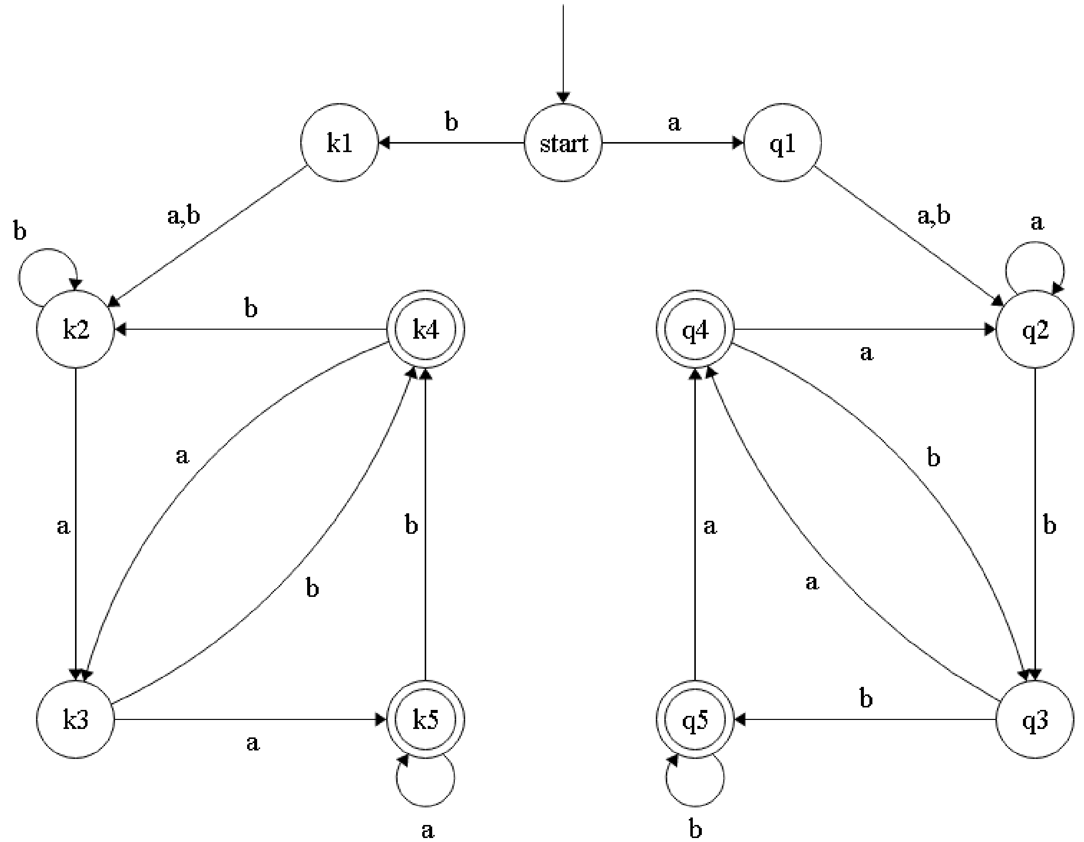


در DFA بالا هر استیت مشخص می کند که تا الان عدد ما به پیمانه 5 چند بوده است. فرض کنید در استیت i باشیم و رشته ما تا الان $X_1X_2…X_n$ باشد، اگر حرف بعدی 1 باشد، یعنی رشته بعدی ما $X_1X_2…X_n1$ خواهد بود، و به راحتی می توان دید که باقی مانده این عدد به 5 برابر است با 2i+1 پس اگر در استیت i باشیم و ورودی 1 را بگیریم به استیت متناظر با 2i+1 میرویم، به طریق مشابه اگر در استیت i باشیم و ورودی 0 را بگیریم باید به استیت 2i برویم.

دقت کنید در اینجا 2i و 2i+1 را به پیمانه 5 نگاه می کنیم.

حال فقط کافیست استیت 0 را به عنوان شروع و استیت 2 را به عنوان پایان تعریف کنیم.

ب)



اول از همه به راحتی میتوان دید که شکل بالا DFA است. در حالت شروع 2 حالت داریم، حرف اول یا a است که در این صورت به سمت راست میرویم و اگر حرف اول b باشد به سمت چپ میرویم، دقت کنید که شکل کاملا متقارن است.

مثلا فرض کنید حرف اول a باشد، پس حرف یکی مونده به آخر باید b باشد.

در DFA بالا داریم که هر رشته ای بخواهد قبول شود طول آن حتما حداقل 4 بوده است، همچنین حرف یکی مانده به آخر رشته هایی که قبول میشوند حتما b بوده است، چون اگر با استیت q4 یک رشته قبول شود، قبل از q4 از یکی از state های q3 یا q5 به q4 state رفته ایم. و نکته این است که q3 و q5 هردو state هایی هستند که فقط با حرف b می توان به آنها رفت، پس نتیجه میگیریم اگر به استیتq4 رفته باشیم حرف یکی مونده به آخر حتما b بوده است.

به روش مشابه می توان گفت که اگر روی استیت q5 باشیم حرف یکی مانده به آخر حتما b است.

حال باید نشان بدهم که تمام رشته هایی که حرف یکی مانده به آخر آنها b است حتما قبول میشوند، برای اثبات این گزاره فرض خلف کنید و فرض کنید رشته ای وجود دارد که طول آن حداقل 4 است و قبول نشده است، در نتیجه این رشته به یکی از استیت های q2 یا q3 رفته است.

اگر به q3 رفته باشد، قبل از q3 در یکی از استیت های q4 یا q2 بوده است. اما دقت کنید که فقط با حرف a می توان به q4 وارد شد، از طرفی هم اگر از q1 صرف نظر کنیم فقط با حرف a میتوان به q2 وارد شد که این نتیجه میدهد حرف یکی مانده به آخر a بوده است. دقت کنید که اگر قرار باشد از q1 به q2 برویم و سپس به q3 برویم طول رشته ما از 4 کمتر میشود اصن.

پس نشان دادیم که اگر با یک رشته ورودی که طولش حداقل 4 باشد به q3 رفته باشیم آن رشته حرف یکی مانده به آخرش حتما a بوده است.

به راحتی هم می توان دید که اگر یک رشته ورودی که طولش حداقل 4 باشد به q2 برود، قبل از آن روی استیت q4 بوده است، و فقط با حرف a می توان به استیت q4 وارد شد، پس حرف یکی مانده به آخرش a بوده است.

پس نشان دادیم از بین رشته هایی که طولشان حداقل 4 است، آن هایی که قبول نمی شوند حتما حرف یکی مانده به آخرشان a بوده است، در نتیجه رشته هایی که حرف یکی مانده به آخرشان b بوده حتما قبول شده اند. از طرفی هم گفتیم آن هایی که قبول میشوند حرف یکی مانده به آخرشان حتما b بوده است. که این همان چیزی بود که می خواستیم.

۲.۱

فرض کنید $D= (Q, \Sigma, \delta, q_0, F)$

تعریف می کنیم $D'=(Q, \Sigma, \delta, q_0, F')$

$$F'= \{t \in F \mid \nexists\, x \in \Sigma^+ : \delta^*(t, x) \in F\}$$

که در اینجا $\delta^* : Q \times \Sigma^* \rightarrow Q$ یک تابعی است که می گویید یک استیت بعد دریافت یک رشته به کدام استیت میرود.

در اینجا داریم F' تمام accept استیت هایی از D است که مسیر جهت داری به استیت های F نداشته باشند.

در این صورت D' اگر رشته ای را قبول کند مثل w، به راحتی می توان دید که w یک رشته از L است، و همچنین امکان ندارد رشته ناتهی x پیدا شود که wx عضوی از L باشد، چون اگر چنین x ای پیدا شود یعنی از از فاینال استیتی که که از ورودی w در D' به آن میرفتیم توانستیم بعدش به یک فاینال استیت دیگر برویم، که این طبق تعریف F' مشکل دارد.

در اصل وجود نداشتن رشته ای مثل x که wx هم عضو L باشد دقیقا معادل این است که اگر با رشته w به accept استیت f_1 رفته باشیم، بعد از f_1 دیگر نتوانیم به accept استیتی برسیم، چون اگر برسیم دقیقا دارد این را می گویید که x وجود دارد.

۳.۱

الف)

فرض کنید $q_{0,3}$ استیت شروع D_3 باشد

طبق تعریف تابع morphism کافیست 2 گزاره زیر را به ازای هر $q \in Q_1$ نشان دهیم:

$$g(f(\delta_1(q, a))= \delta_3(g(f(q)), a)$$

$$g(f(q_{0,1})) = q_{0,3}$$

برای گزاره اول داریم که:

$$f(\delta_1(q, a)) = \delta_2(f(q), a)$$

حال دقت کنید که 2 طرف تساوی بالا دارند یک استیت از Q_2 را نشان میدهند، در نتیجه می توان از 2 طرف g گرفت:

$$g(f(\delta_1(q, a))) =g(\delta_2(f(q), a))$$

دقت کنید که طبق تعریف، به ازای هر $q'\in Q_2$ داریم که $g(\delta_2(q', a))= \delta3(g(q'), a)$، حال دقت کنید که $f(q) \in Q_2$ در نتیجه می توان در این عبارت بجای q' مقدار $f(q)$ را قرار دهیم و به تساوی $g(\delta_2(f(q), a))= \delta3(g(f(q)), a)$ برسیم، در نتیجه:

$$g(f(\delta_1(q, a))) =g(\delta_2(f(q), a)) = \delta_3(g(f(q)), a)$$

برای گزاره دوم هم داریم که:

$$f(q_{0,1})=q_{0,2} \Rightarrow g(f(q_{0,1}))=g(q_{0,2})=q_{0,3}$$

ب)

برای اثبات این گزاره روی طول w استقرا میزنیم.

برای پایه استقرا حالت های $|w|=0$ و $|w|=1$ را باید بررسی کنیم.

حالت $|w|=0$ بدیهی است و حالت $|w|=1$ همان فرض تابع morphism است.

برای گام استقرا فرض کنید حکم برای w درست باشد، نشان میدهیم برای هر $a\in \Sigma$ حکم برای wa هم درست است.

میدانیم که:

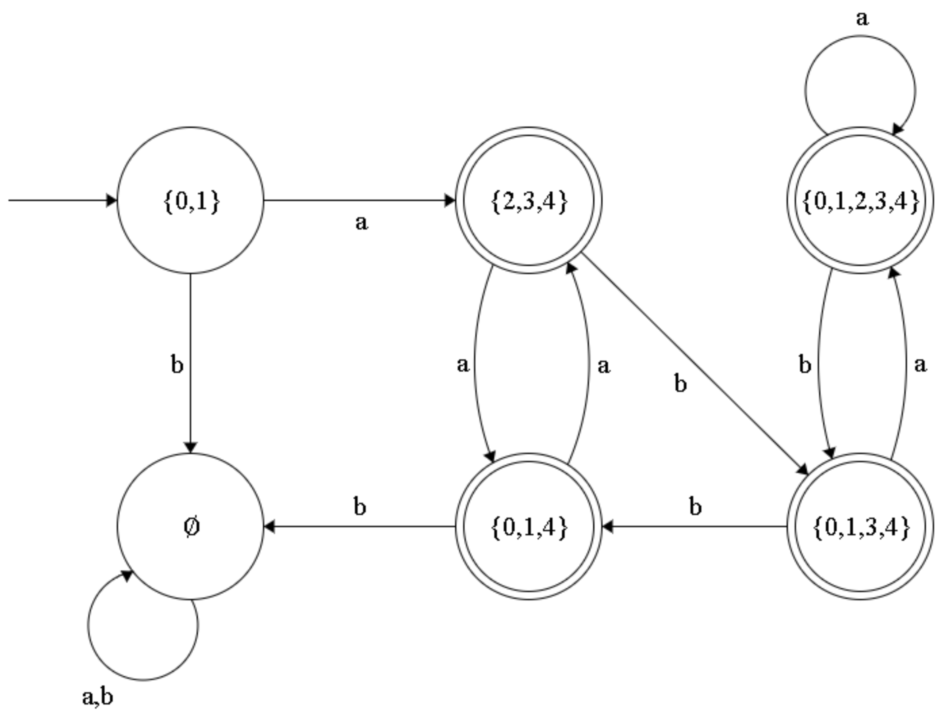
$$\delta^*_1(q, wa)= \delta^*_1 (\delta^*_1(q, w),a)$$

در نتیجه داریم که:

$$h(\delta^*_1(q, wa))= h(\delta^*_1 (\delta^*_1(q, w),a))= \delta^*_2(h(\delta^*_1(q, w)),a)= \delta^*_2(\delta^*_2 (h(q),w),a)= \delta^*_2(h(q),wa)$$

۲. هم‌ارزی و کمینه‌سازی

۱.۲



طبق الگوریتم پیش میریم، اول از همه میدانیم استیت شروع 0 است که چون با اپسیلون به 1 راه دارد استیت شروع جدیدمان {0,1} میشود.

استیت های {0,1} با b به هیچ جا نمیروند، پس آنها را به استیت تهی میبریم.

استیت تهی هم می دانیم کلا با هر ورودی ای به خودش میرود.

استیت های {0,1} با ورودی a به {2,3,4} می توانند بروند، پس این استیت را اضافه می کنیم.

استیت های {2,3,4} با ورودی a به {0,1,4} می توانند بروند و با ورودی b به {0,1,3,4} می توانند بروند. پس این دو تا استیت جدید را اضافه می کنیم.

استیت های {0,1,4} با ورودی b به هیچی نمی توانند بروند پس آنها را به تهی میبریم. با ورودی a هم به {2,3,4} ممکن است بروند.

استیت های {0,1,3,4} با ورودی a به تمامی استیت ها می توانند بروند، پس استیت {0,1,2,3,4} را اضافه می کنیم.

استیت های {0,1,3,4} با ورودی b به {0,1,4} می توانند بروند.

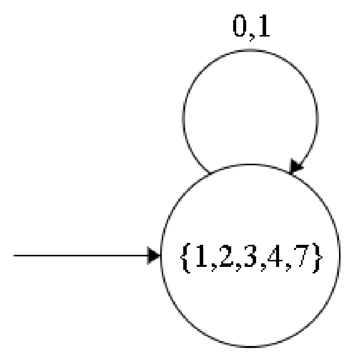
استیت های {0,1,2,3,4} با ورودی a به همین مجموعه برمیگردد، و با ورودی b به {0,1,3,4} میروند.

دقت کنید که دیگر استیت جدیدی نیاز نیست تولید کنیم و dfa ما کامل است.

۲.۲

دقت کنید که میتوانیم استیت هایی که به آنها هیچ وقت وارد نمیشویم را همان اول حذف کنیم. الان استیت های 5 و 6 هر دو اینگونه هستند که هیچ transition ای به سمتشان وجود ندارد و استیت شروع هم نیستند، پس 2 تا استیتی هستند که هیچ وقت به آنها وارد نخواهیم شد.

دقت کنید که این 2 استیتی که گفتیم میتوانیم حذف کنیم تنها استیت های final ما بودند، پس بعد از حذف، dfa ما دیگر حالت فاینال نخواهد داشت، پس با یک dfa رو برو هستیم که هیچ رشته ای را نمی تواند accept کند، و به وضوح چنین dfa ای با dfa زیر هم ارز است.



۳. خواص بستاری زبان‌های منظم

۱.۳

فرض کنید که $x=a_1a_2...a_m$ و همچنین فرض کنید که $D= (Q, \Sigma, \delta, q_0, F)$ برابر باشد با DFA ای که زبان L را میپذیرد.

فرض کنید استیت های این DFA، استیت های q_0 تا q_n باشند.

یک NFA تولید می کنیم به صورت $D'= (Q', \Sigma, \delta', k_{0,0}, F')$.

فرض کنید که این ماشین (n+1)(m+1) استیت دارد، که اسم های آن به صورت $k_{i,j}$ هستند که در آن i از 0 تا n می تواند باشد و j از 0 تا mمی تواند باشد.

برای 'δ این گونه تعریف می کنیم:

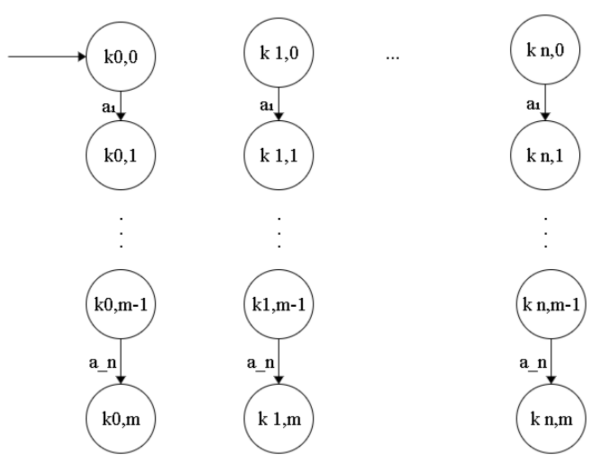
$$\delta'(k_{i,j}, \mathrm{r}) = \begin{cases} \{k_{t,j}\} & if\ j = 0\ and\ r \neq a_1\ and\ \delta(q_i, \mathrm{r}) = q_t \\ \{k_{t,0}, k_{i,1}\} & if\ j = 0\ and\ r = a_1\ and\ \delta(q_i, \mathrm{r}) = q_t \\ \{k_{i,j+1}\} & if\ \ m-1 \geq j \geq 1\ and\ r = a_{j+1} \\ \{k_{t,j}\} & if\ j = m\ \ and\ \delta(q_i, \mathrm{r}) = q_t \end{cases}$$

برای 'F هم تعریف می کنیم:

$F'=\{k_{i,m} \mid q_i \in F\}$

حال اثبات می کنم که 'D بیانگر زبان put(L,x) است.

اتفاقی که دارد اینجا می افتد این است که اگر از استیت های فاینال صرف نظر کنیم، استیت های $k_{0,0}$ تا $k_{n,0}$ دقیقا کپی q_0 تا q_n هستند، اما هر کدام از آنها مثل $k_{i,0}$ اگر ورودی a_1 بگیرد می توانند به $k_{i,1}$ برود که این در اصل مشخص می کند که رشته x از کجا شروع میوشد. همچنین $k_{0,m}$ تا $k_{n,m}$ هم دقیقا کپی q_0 تا q_n هستند و استیت های فاینالشان متناظر است.



در اصل اگر طبق شکل بخواهیم بگیم، یعنی از $k_{0,0}$ شروع می کنیم و رشتمان را تا جایی جلو میبریم تا مثلا به استیت $k_{i,0}$ برسیم و یک طبقه پایین برویم، یعنی در اصل اینجا شروع کردیم به اضافه کردن رشته x، و آنقدر پایین میرویم تا به طبقه آخر برسیم و رشته x به طور کامل اضافه شده باشد. حالا در خانه $k_{i,m}$ خواهیم بود، و گفتیم $k_{0,m}$ تا $k_{n,m}$ هم دقیقا کپی q_0 تا q_n هستند، پس سپس می توانیم در طبقه آخر ادامه دهیم تا به استیت فاینال برسیم.

۲.۳ (امتیازی)

الف)

لم 1: برای هر زبان منظمی مثل L گذاره زیر درست است:

$$\exists\ p,t\epsilon N: \forall\ i\geq p : (\exists\ x\epsilon L: |x|=i \Leftrightarrow \exists\ x\epsilon L: |x|=i+t)$$

به بیانی دیگر در این لم ادعا می کنم که مجموعه اعداد صحیحی مثل i، که در زبان منظم L رشته ای وجود داشته باشد که طولش برابر با i باشد، از جایی به بعد متناوب است.

اثبات لم:

فرض کنید که D، DFA زبان L باشد، مجموعه X_i را تعریف می کنیم تمام استیت هایی از D که بعد i حرف می توان به آنها رسید. در این صورت اگرحداقل یک استیت فاینال در X_i وجود داشته باشد، می توان گفت که رشته ای به طول i وجود دارد که زبان L آن را می پذیرد و در غیر این صورت زبان L هیچ رشته ای به طول i نمی پذیرد.

چون تعداد استیت ها محدود است به راحتی طبق لانه کبوتری می توان دید که اعداد P و t پیدا میشوند که $X_p=X_{p+t}$. حال دقت کنید که طبق تعریف مجموعه X ها به راحتی می توان نتیجه گرفت $X_{p+1}=X_{p+t+1}$ چون X_{p+1} شامل استیت هایی است که از استیت های X_p می توان با یک حرف به آنها رفت، و چون $X_p=X_{p+t}$ می توان دید که $X_{p+1}=X_{p+t+1}$.

حال همینجوری و با روند استقرایی می توان نشان داد که برای هر i≥p داریم که:

$$X_i=X_{i+t}$$

پس وجود داشتن رشته ای به طول i در زبان L با وجود داشتن رشته با طول i+t در این زبان معادل است، که این همان چیزی بود که می خواستیم اثبات کنیم.

لم 2: برای هر زبان منظمی مثل L گذاره زیر درست است:

$$\exists\ p,t\epsilon N: \forall\ i\geq p : (\exists\ x\epsilon L: |x|=i^2 \Leftrightarrow \exists\ x\epsilon L: |x|=(i+t)^2)$$

به بیانی دیگر در این لم ادعا می کنم که مجموعه اعداد صحیحی مثل i، که در زبان منظم L رشته ای وجود داشته باشد که طولش برابر با i^2 باشد، از جایی به بعد متناوب است.

اثبات لم:

$$i\geq p \Rightarrow i^2\geq p \Rightarrow$$

$$\exists\ x\epsilon L: |x|=i^2 \Leftrightarrow \exists\ x\epsilon L: |x|=i^2+t \Leftrightarrow \ldots \Leftrightarrow \exists\ x\epsilon L: |x|=i^2+2it+t^2$$

$$\Leftrightarrow \exists\ x\epsilon L: |x|=(i+t)^2$$

حال به سمت حل خود سوال میرویم، طبق لم 2، عدد t و p وجود دارند که برای هر $i\geq p$ داشته باشیم که:

$$\exists\ x\epsilon B: |x|=i^2 \Leftrightarrow \exists\ x\epsilon B: |x|=(i+t)^2$$

فرض کنید که در dfa زبان A، استیت ها q_0 تا q_n باشند و فرض کنید q_0 استیت شروع باشد.

یک nfa تعریف می کنیم که اسم استیت های آن $q_{i,j}$ باشد، به طوری که i از 0 تا n می تواند باشد و j از 0 تا p+t-1 می تواند باشد. یعنی یک nfa تعریف کردیم که $(n+1)(p+t)$ تا استیت دارد.

این nfa اگر $D= (Q, \Sigma, \delta, q_{0,0}, F)$ باشد باید F و δ را در آن تعریف کنیم.

فرض کنید که δ_A تابع transition در dfa زبان A باشد.

تابع δ را به اینگونه تعریف می کنیم:

$$\delta(q_{i,j}, \mathrm{r}) = \begin{cases} q_{k,j+1} & \delta_A(\ q_i, r) = q_k\ and\ j < p+t-1 \\ q_{k,j+1-t} & \delta_A(\ q_i, r) = q_k\ and\ j = p+t-1 \\ \emptyset & \text{otherwise} \end{cases}$$

همچنین F را به این صورت تعریف می کنیم:

$$F = \{q_{i,j}\ |\ q_i\epsilon\ F_A \text{ and } \exists\ y\epsilon B:\ |y|=j^2\}$$

که در اینجا F_A برابر است با مجموعه فاینال استیت های dfa زبان A.

ادعا می کنم NFA ای که تعریف کردم زبان C(A,B) را میسازد.

در اصل اتفاق که دارد اینجا می افتد این است که اگر در استیت $q_{i,j}$ باشیم، انگار داریم فرض می کنیم که در استیت q_i در dfa زبان A هستیم و تابع transition ما با اندیس اول دقیقا مشابه تابع transition زبان A رفتار می کند.

اندیس دوم در $q_{i,j}$ دارد می گوید که اگر رشته ما تا به اینجا m تا حرف شده باشد و $m\geq p+t$ در این صورت:

$$m\equiv j (\text{mod } t)$$

که این یعنی وجود داشتن رشته ای به طول m^2 در زبان B معادل وجود داشتن رشته ای به طول j^2 در زبان B است.

برای همین است که $\delta(q_{i,j}, \mathrm{r})$ همیشه اندیس دومش به پیمانه t دارد یدونه زیاد میشود.

حالا کدام استیت ها باید accept شوند؟ آن استیت هایی که اولا اندیس اولشان در dfa زبان A فاینال باشد، چون رشته ما باید حتما رشته ای باشد که مال زبان A است. دوما اگر اندیس دوم استیتی که در آن هستیم j باشد، پس باید y عضو زبان B وجود داشته باشد که طولش برابر با j^2 باشد. این در اصل دارد می گوید که اگر رشته ما تا الان m حرف شده باشد، حتما در زبان B رشته به طول m^2 وجود دارد، چون اگر $m<p+t$ پس یعنی تا الان j حرفی بوده رشتمون، اگر هم $m\geq p+t$ که یعنی m و j به پیمانه t یکی بودند، پس وجود داشتن رشته به طول m^2 در زبان B معادل وجود داشتن رشته به طول j^2 در B بوده است.

پس این nfa ما زبان C(A,B) را توصیف می کند.

ب)

فرض کنید که dfa زبان L، برابر باشد با $D= (Q, \Sigma, \delta, q_0, F)$ و استیت هایش q_0 تا q_p باشد. زبان R_i را این گونه تعریف می کنیم، $R_i=(Q, \Sigma, \delta, q_i, F)$، یعنی ماشین R_i دقیقا مانند مایشین D است با این فرق که استیت شروعش q_i است.

همچنین فرض کنید که $T_i=(Q, \Sigma, \delta, q_0, \{T_i\})$، یعنی T_i شبیه همان ماشین D است با این اختلاف که استیت پایانش فقط q_i است.

حال طبق قسمت الف سوال میدانیم که زبان زیر منظم است:

$$C(T_i ,R_i)= \{x \in T_i \mid (\exists y) [|y| = |x|^2 , y \in R_i] \}$$

اگر $C(R_i , T_i)$، را بخواهیم تحلیل کنیم، این زبان دارد تمام رشته هایی مثل x را قبول می کند، که اولا x عضو T_i است، که یعنی x رشته ای بوده که در dfa زبان L به استیت q_i رسیده است، همچنین رشته y عضو R_i باید وجود داشته باشد که

$|y| = |x|^2$، که دقت کنید چون استیت شروع R_i، استیت q_i بوده، و استیت های فاینالش همان استیت فاینال های D بوده می توان گفت که xy عضو زبان L بوده است، پس انگار$C(T_i ,R_i)$ ، بیانگر رشته هایی مثل x است که برای آنها شرط

$(\exists y) [|y| = |x|^2, xy \in L]$ برقرار باشد، و داشته باشیم که رشته x در dfa زبان L به استیت q_i رفته باشد، حال واضح است اگر برای تمام i ها زبان $C(T_i ,R_i)$ را اجتماع بگیریم زبان sqrt(L) را تشکیل میدهد، و چون اشتراک چند تا زبان منظم، منظم است، نتیجه میگیریم که sqrt(L)، منظم است.