نظریه زبان ها و ماشین ها

مهندسی کامپیوتر

نظریه زبان ها و ماشین ها

سرشناسه	:	شیرافکن، فرشید، 1351 -
عنوان و نام پدیدآور	:	مجموعه کامپیوتر(نظریه زبان‌ها و ماشین‌ها)/ گردآورنده فرشید شیرافکن.
مشخصات نشر	:	تهران: سنجش و دانش، 1390.
مشخصات ظاهری	:	118ص.؛ 22×29س‌م.
شابک	:	۹۷۸-۶۰۰-۲۳۲-۲۰۰-۵
وضعیت فهرست نویسی	:	فیپا
موضوع	:	دانشگاه‌ها و مدارس عالی -- ایران -- آزمون‌ها
موضوع	:	زبان‌های صوری-- آزمون‌ها و تمرین‌ها (عالی)
موضوع	:	زبان‌های صوری-- راهنمای آموزشی (عالی)
موضوع	:	نظریه ماشین -- راهنمای آموزشی (عالی)
موضوع	:	نظریه ماشین -- آزمون‌ها و تمرین‌ها (عالی)
موضوع	:	آزمون دوره‌های تحصیلات تکمیلی – ایران
رده بندی کنگره	:	LB2353/1390 383 ش 9827 م
رده بندی دیویی	:	1664/378
شماره کتابشناسی ملی	:	2499283

مجموعه مهندسی کامپیوتر ـ نظریه زبان ها و ماشین ها

گردآورنده: فرشید شیرافکن

ناشر: انتشارات سنجش و دانش

صفحه آرایی: مهناز تقوی

نوبت چاپ: دوم، ۱۳۹۳

تیراژ: ۵۰۰ نسخه

قیمت: ۲۳۶۰۰ تومان

شابک: ۹۷۸-۶۰۰-۲۳۲-۲۰۰-۵

نشانی: تهران، میدان انقلاب، خیابان دانشگاه، تقاطع روانمهر، پلاک۱۲۶، ساختمان سنجش و دانش

تلفن: ۶۱۲۶ ـ۰۲۱

www.sanjesh.ir

فهرست مطالب

سخن سنجش و دانش

آغاز هر سخن،زیبنده ستایش خالق یکتاست. اوست که بر هر نقشی، نگاری می بندد و بر هر نگاری، زیبایی.

بر هیچ عاقل و فرزانه ای پوشیده نیست که موتور پیشرفت و سربلندی هر کشوری بسته به علم و دانش است و طبق فرموده مقام معظم رهبری: اگر ملتی استقلال می خواهد ، اگر عزت می خواهد، اگر پیشرفت می خواهد باید دانشگاه خود را تقویت کند.

در این راستا، انتشارات سنجش و دانش(جامع ترین مرکز آموزش مکاتبه ای کشور) مفتخر است که سالهای متمادی در جهت خدمت به جامعه علمی کشور و بخصوص داوطلبان مقاطع کاردانی به کارشناسی ، کارشناسی ارشد و همچنین دکترا ، فعالیت می کند.

در این مسیر پر پیچ و خم علم آموزی و پیشرفت، یکی از مشکلاتی که همواره دانشجویان با آن مواجه هستند، از یک طرف گستردگی و پراکندگی منابع مطالعاتی و از طرف دیگر کمبود زمان آزمون کنکور ، جهت پاسخگویی به سوالات است. تا کنون جزوات و کتابهای گوناگونی در جهت رفع این مشکلات به بازار عرضه شده اما باز با این حال، پاسخ گوی نیاز خیلی از داوطلبان نیست، چرا که بعضاً یا خیلی خلاصه و گزیده است یا این که در تشریح مسائل و پاسخ گویی به سوالات از سیستمی استفاده شده که با وجود کامل و جامع بودن اکثراً با توجه به زمان محدود آزمون کنکور، عملاً سر جلسه امتحان قابل استفاده نمی باشد. به این معنا که داوطلب هر چند تسلط کامل به مطالب درسی دارد اما این مطالب در ذهن او از آنچنان انسجام و هماهنگی لازم برخوردار نیست که داوطلب بتواند در یک زمان کم به جواب سوال برسد لذا داوطلب در پاسخ گویی به سوالات دچار کمبود وقت گردیده و عملاً نمی تواند به تمام سوالات آنگونه که از خود انتظار دارد جواب دهد.

انتشارات سنجش و دانش با توجه به این دو مسئله مهم(یکی گستردگی منابع و دیگری زمان کم پاسخ گویی به سوالات کنکور) بر آن شد تا با استفاده از تجربه و علم اساتید مجرب و کارآزموده ، به تولید و انتشار کتبی بپردازد که عین خلاصه و موجز بودن کامل و جامع نیز باشد ، کما این که سعی گردیده در تشریح مسائل از یک سیستم جدید و راهکار میانبری استفاده شود که بدین وسیله مشکل کمبود زمان در جلسه کنکور نیز مرتفع گردد.

نیاز به استفاده از یک سیستم جدید به این دلیل است که توان پاسخ گویی به سوالات کنکور جدای از نیاز به بار علمی، نیازمند یک مهارت و شیوه خاص در تست زنی نیز می باشد. لذا در این خصوص سعی شده مطالب کتاب به گونه ای طرح ریزی و تألیف شود که ، داوطلب خود به خود علاوه بر یاد گیری مطالب به مهارت تست زنی نیز دست پیدا کند.

در آخر از مخاطبین محترم این کتاب نهایت سپاسگزاری و قدر دانی را داریم و امید داریم توانسته باشیم آنچه را که شایسته و برازنده یک دانشجوی ایرانی است ارائه کرده باشیم. از دانشجویان عزیز و اساتید محترم نیز تقاضامندیم ما را از نقطه نظرات و پیشنهادات خود بی بهره نگذارند چرا که تنها افتخار و دست آویز ما نگاه صمیمانه و رضایت بخش شماست.

به امید پیروزی و سربلندی در تمامی عرصه های زندگی

Error.azmoon@gmail.com

تلفن 021-6126

فصل اول: نگاهی کلی به زبان ، گرامر و ماشین

نظریه محاسبات، سرفصل های متنوعی از جمله نظریه ماشین ها، گرامرها و زبانهای صوری، محاسبه پذیری و پیچیدگی را شامل می شود. این موضوعات در مجموع پایه نظری علوم کامپیوتر را تشکیل می دهند. در این کتاب ماشین های مختلف را مطالعه کرده و نحوه ارتباط آنها با زبان ها و گرامرها را مورد بررسی قرار می دهیم. ایده های مطرح شده در این کتاب، کاربرد مستقیم و مهمی در زبان‌های برنامه سازی و کامپایلرها دارد.

مجموعه ها

مجموعه گروهی از اعضاء است که ساختاری غیر از عضویت ندارند. می گوییم x متعلق به مجموعه S است و می نویسیم $x \in S$، بالعکس، عبارت $x \notin s$ به این معناست که x متعلق به مجموعه S نیست.

تذکر: مجموعه نمی‌تواند دارای عضو تکراری باشد و ترتیب قرار گرفتن اعضای مجموعه مهم نمی‌باشد.

عملگرهای مجموعه

عملگرهای معمول بر روی مجموعه ها شامل اجتماع (U)، اشتراک (I)، تفاضل(-) است که به صورت زیر تعریف می شوند:

$$A \cup B = \{x \mid x \in A \quad \text{or} \quad x \in B\}$$
$$A \cap B = \{x \mid x \in A \quad \text{and} \quad x \in B\}$$
$$A - B = \{x \mid x \in A \quad \text{and} \quad x \notin B\}$$

عملگر مکمل

مکمل مجموعه Sبصورت $\bar{s}$ نشان داده شده و شامل تمام عناصر غیر موجود در S است. $\bar{S} = \{x : x \in U, x \notin S\}$

که U ، همان مجموعه جهانی است که شامل تمام اعضاء ممکنه می باشد.

روابط مقابل برقرار است: $\bar{\bar{A}} = A$, $\overline{A \cap B} = \bar{A} \cup \bar{B}$, $\overline{A \cup B} = \bar{A} \cap \bar{B}$,

مجموعه تهی

مجموعه تهی(پوچ)، مجموعه ای است که هیچ عضوی نداشته و با f نمایش داده می شود.

روابط مقابل برقرار است: $A \cup \phi = A$ $\bar{f} = U$, $\bar{\phi} \cup \phi = U$, $A \cap \phi = \phi$,

مجموعه متناهی و نامتناهی

یک مجموعه اگر حاوی تعداد متناهی از اجزاء باشد، مجموعه متناهی و در غیر اینصورت مجموعه نامتناهی نامیده می شود. اندازه یک مجموعه متناهی برابر با تعداد اعضاء موجود در آن است و بصورت |S| نمایش داده می شود.

سه مفهوم اساسی

در درس نظریه زبانها و ماشین ها، سه مفهوم زیر بررسی می شوند:

زبان	زبان، مجموعه ای از رشته ها روی یک الفبا می باشد.
گرامر	گرامر، ابزاری برای تولید زبان می باشد.
ماشین	ماشین(اتوماتا)، ابزاری برای پذیرش زبان می باشند. توسط ماشین می توان تشخیص داد که آیا یک رشته مربوط به زبان هست یا نه.

زبان

یک زبان در اغلب موارد بعنوان زیر مجموعه ای از $\sum^*$ تعریف می شود. هر رشته در زبان، جمله ای از زبان خوانده می شود. می توان هر مجموعه ای از رشته های روی یک الفبای Σ را یک زبان تلقی کرد.

چند تعریف بر روی رشته ها در زیر آورده شده است:

۱- طول : **طول رشته برابر تعداد سمبل های موجود در رشته است. (طول رشته w با $|w|$ نشان داده می شود.)**

۲- الحاق: **الحاق دو رشته v,w، یعنی(wv) رشته ای است که با اتصال سمبل های v به گوشه سمت راست w بدست می آید.**

۳- معکوس : **معکوس رشته با نوشتن سمبل ها در جهت عکس بدست می آید.**

۴- زیر رشته : **هر دنباله متوالی از سمبل ها در w، زیر رشته w خوانده می شوند.**

5- **پیشوند و پسوند** : اگر w=vu ، آنگاه زیر رشته v پیشوند و زیر رشته u ، پسوند رشته w خوانده می شوند.

مثال: اگر w=abbab ، آنگاه مجموعه تمام پیشوندهای رشته w برابر است با:

$\{l, a, ab, abb, abba, abbab\}$

و برخی از پسوندهای این رشته برابر است با: $\{bab, ab, b\}$

طول الحاق رشته ها، برابر با مجموع طول هر یک از آنها می باشد، یعنی: $|uv| = |u| + |v|$

رشته تهی، رشته ای است که هیچ سمبولی ندارد و بوسیله l نمایش داده می شود. برای هر رشته w داریم: $l w = w l = w$ و $|\lambda| = 0$.

مثال: در الفبای $\sum = \{a, b, c\}$ با فرض w=abbc آنگاه w^R برابر است با cbba.

چنانچه Σ یک الفبا باشد، آنگاه از $\sum^*$ برای نمایش مجموعه رشته های بدست آمده از الحاق صفر یا بیشتر سمبل دیگر استفاده می کنیم. مجموعه $\sum^*$ همواره حاوی l خواهد بود. $\sum^*$ بدون l را با $\sum^+$ نمایش می‌دهند:

$\Sigma^+ = \Sigma^* - \{l\}$

مثال: با فرض $\sum = \{a\}$ داریم: $\sum^+ = \{a, aa, aaa, ...\}$

مثال: با فرض $\sum = \{a, b\}$ داریم: $\sum^* = \{\lambda, a, b, bb, ab, aa, aab, ...\}$

Σ^+ **و $\sum^*$ همواره نامتناهی هستند.**

مثال: با فرض $L = \{ab, aa, baa\}$ کدامیک از رشته های زیر در L^* موجود است؟

الف- abaabaaabaa

ب- baaaaabaaaab

حل: رشته الف در L^* موجود است، چون می توان آن را به رشته های ab,aa,baa,ab,aa تجزیه کرد که همگی در L قرار دارند. اما هیچ تجزیه ای برای رشته ب ممکن نیست و بنابراین رشته ب در L^* قرار ندارد.

■

منظور از $x^{(R^n)}$، n بار معکوس کردن رشته x می باشد. اگر n زوج باشد، حاصل برابر x و اگر n فرد باشد، برابر معکوس رشته x می باشد.

مثال: اگر x=abb باشد، حاصل 4 بار معکوس کردن رشته x برابر خود رشته یعنی abb می باشد. همچنین حاصل 7 بار معکوس کردن رشته x برابر معکوس رشته یعنی bba می باشد.■

مثال: با فرض $u = aba^2$ داریم: $u^2 = aba^2 . aba^2 = aba^3ba^2$

به ازای تمام رشته‌های u و تمام n ها داریم: $|u^n| = n|u|$

اگر Σ شامل n عنصر باشد، آنگاه Σ^* دارای n^k رشته به طول k است.

مثال: مجموعه $L = \{a^n b^n : n \ge 0\}$، یک زبان بر روی الفبا $\Sigma = \{a, b\}$ است که شامل رشته‌هایی با تعداد برابر a و b می‌باشد، مانند aabb.■

عملیات روی زبانها

عملیات قابل انجام بر روی زبانها عبارتند از: اتصال، معکوس، مکمل، اجتماع، اشتراک و تفاضل. از آنجا که زبان ها مجموعه هستند، عملیات اجتماع، اشتراک و تفاضل دو زبان به راحتی قابل تعریف است. عملیات دیگر به صورت زیر تعریف می شود:

۱- اتصال : $L_1 . L_2 = \{xy : x \in L_1 , y \in L_2\}$

۲- معکوس: $L^R = \{W^R : W \in L\}$

۳-مکمل : $\bar{L} = \Sigma^* - L$

L^n **به معنای اتصال L به تعداد n بار با خودش می‌باشد:** $L^1 = L$ $L^0 = \{\lambda\}$,

بستار ستاره ای زبان L : $L^* = L^0 \cup L^1 \cup L^2 \ldots = \bigcup_{i \ge 0} L^i$

بستار مثبت زبان L : $L^+ = L^1 \cup L^2 \ldots = \bigcup_{i \ge 1} L^i$

مثال : اگر $L = \{a^n b^n : n \ge 0\}$، آنگاه L^2 کدام است؟

حل: $L^2 = \{a^n b^n a^k b^k : n \ge 0, k \ge 0\}$

توجه کنید که n و k ارتباطی به هم ندارند و رشته aabbab(یعنی n=2,k=1) در L^2 موجود است. ■

مثال: تعداد اعضای $L_1 L_2$ را تعیین کنید.($L_1 = \{10,1\}$ $L_2 = \{01,011,11\}$,)

$L_1 L_2 = \{1001,10011,1011,101,1011,111\} = \{1001,10011,101,111\}$

بنابراین $L_1 L_2$ دارای 4 عضو می باشد.■

مثال: اگر $L=\{a^n b^n : n \ge 0\}$، آنگاه معکوس زبان L یعنی L^R کدام است؟

حل: $L^R = \{b^n a^n : n \ge 0\}$ ■

مثال: با فرض $\Sigma = \{a,b\}$ و $L=\{aa,bb\}$، مکمل L را بدست آورید.

حل: $\overline{L} = \{\lambda, a, b, ab, ba\} \cup \{w \in \{a,b\}^* : |w| \ge 3\}$ ■

مثال: با توجه به زبانهای زیر، عبارت $L_1 = L_4{}^*(L_2 \cup L_3)$ نمایانگر چیست؟

$L_1 = \{w \mid w \in \{a,b\}^*, |w| \bmod 3 \ge 1\}$ $L_2 = \{a,b\}$

$L_3 = \{aa, ab, ba, bb\}$ $L_4 = \{aaa, aab, aba, abb, baa, bab, bba, bbb\}$

حل: با توجه به زبانها داریم: $L_2 = \Sigma,\ L_3 = \Sigma^2, L_4 = \Sigma^3$ و بنابراین $L_1 = L_4{}^*(L_2 \cup L_3)$ را می توان به فرم $(\Sigma^3)^*(\Sigma \cup \Sigma^2)$ نمایش داد، که معرف مجموعه رشته‌هایی می‌باشد که طول آنها مضرب 3 نمی‌باشد.

روابط زیر برقرار است:

$L_1L_2 \ne L_2L_1$ / $L_1(L_2 \cup L_3) = L_1L_2 \cup L_1L_3$ / $L_1(L_2 \cap L_3) \ne L_1L_2 \cap L_1L_3$

عملگر بستار ستاره ای نسبت به هیچیک از عملگرهای "اجتماع ، اشتراک ، تفاضل و الحاق" خاصیت پخشی ندارد:

$(L_1 \cup L_2)^* \ne L_1^* \cup L_2^*$ / $(L_1 \cap L_2)^* \ne L_1^* \cap L_2^*$ / $(L_1.L_2)^* \ne L_1^*.L_2^*$ / $(L_1 - L_2)^* \ne L_1{}^* - L_2{}^*$

عملگر بستار مثبت نیز نسبت به هیچیک از عملگرهای "اجتماع ، اشتراک ، تفاضل و الحاق" خاصیت پخشی ندارد.

برای زبانهای L_2, L_1 روابط زیر برقرار است: $(L_1L_2)^R = L_2{}^R L_1{}^R$ $(L_1 + L_2)^R = L_1^R + L_2^R$,

انواع زبان ها

1- زبان های منظم(نوع 3)

2- زبان های مستقل از متن(نوع 2)

3- زبان های حساس به متن(نوع 1)

4- زبان های بازگشتی شمارش پذیر (نوع 0)

گرامرها

گرامر G به صورت $G = (V, T, S, P)$ تعریف می‌شود که:

V : مجموعه متناهی از اشیاءبه نام متغیرها.

T : مجموعه متناهی از اشیاء به نام سمبل های پایانی(ترمینال).

S : سمبل ویژه ای به نام متغیر شروع

P : مجموعه متناهی از قوانین

تذکر: مجموعه های V و T غیر تهی و جدا از هم می باشند.

قوانین گرامر به شکل $x \to y$ می‌باشند که در آن $x \in (V \cup T)^+$ و $y \in (V \cup T)^*$ می باشد.

زبان تولید شده توسط گرامر

با استفاده از گرامرها می توان بوسیله بکار بردن قوانین با ترکیب های مختلف، رشته های متعددی تولید کرد. مجموعه این رشته های پایانی، زبانی است که بوسله گرامر تولید می شود.فرض کنیم که $G = (V, T, S, P)$ یک گرامر باشد. آنگاه مجموعه $L(G) = \left\{ W \in T^* : S \overset{*}{\Rightarrow} W \right\}$ ، زبانی است که توسط گرامر G تولید می شود.

تذکر: عبارت $W_1 \overset{*}{\Rightarrow} W_n$ یعنی W_n از W_1 با تعداد نامشخص (حتی صفر) مشتق می‌شود.

مثال: نحوه تولید رشته aabbcc را با توجه به گرامر زیر نمایش دهید.

$1. S \to aSBC \mid abc$

$2. cB \to Bc$

$3. bB \to bb$

$4. cC \to cc$

حل: شماره قاعده استفاده شده در هر مرحله نمایش داده شده است:

$$S \xrightarrow{1} aSBC \xrightarrow{1} aabcBC \xrightarrow{2} aabBcC \xrightarrow{3} aabbcC \xrightarrow{4} aabbcc$$

مثال: نحوه تولید رشته abbaba ، توسط گرامر زیر را مشخص کنید.

$S \to aB \mid bA$

$A \to aS \mid bAA \mid a$

$B \to bS \mid aBB \mid b$

حل: $S \to aB \to abs \to abbA \to abbas \to abbabA \to abbaba$

مثال:زبان تولید شده توسط گرامر G را تعیین کنید.

$G = (\{S\}, \{a, b\}, S, P) \quad P : S \to aSb \mid \lambda$

حل: توسط این گرامر رشته‌های شروع شونده با a که تعداد a و b در آنها برابر است تولید می شود.

$$L(G) = \{a^n b^n \mid n \geq 0\}$$

مثلاً می‌توان aabb را به ترتیب زیر تولید کرد:

$S \Rightarrow asb \Rightarrow aasbb \Rightarrow aabb$

تذکر: می توان این گرامر را به صورت زیر نیز نوشت:

$$S \to aAb \mid l$$
$$A \to aAb \mid l$$

(در این گرامر سمبل های S,A که با حروف بزرگ مشخص شده اند، متغیر و سمبل های a و b ، ترمینال می باشند.)

مثال: گرامر سازنده زبان $L(G) = \{a^n b^{n+1} : n \geq 0\}$ را بنویسید.

حل: این مثال با حالت قبلی فقط در یک b تفاوت دارد : $a^n b^n b$

$S \to Ab$

$A \to aAb \mid \lambda$

مثال: گرامر تعیین کننده زبان $L = \{W : n_a(W) = n_b(W)\}$ را بنویسید. $((\Sigma = \{a, b\})$

حل: زبان تولید شده شامل رشته‌هایی با تعداد a و b های برابر است.(جملات با a و یا b شروع می شوند.)

$S \to SS \mid aSb \mid bSa \mid \lambda$

مثال: گرامر سازنده زبان $L = \{w : n_a(w) = n_b(w) + 1\}$ را بنویسید.

حل:

$S \to aX \mid XS$

$X \to XX \mid aXb \mid bXa \mid \lambda$

مثال: گرامر تعیین کننده زبان $L = \{a^n b^m : n \geq 0, m > n\}$ را بنویسید. $((\Sigma = \{a, b\})$

حل: ابتدا به تعداد مساوی aو b و سپس در صورت نیاز، یک یا چند b تولید می شود:

$S \to AB$

$A \to aAb \mid l$

$B \to bB \mid b$

مثال: زبان تولید شده توسط گرامر زیر را تعیین کنید.

$S \to aSBC \mid \lambda$

$cB \to Bc$

$aB \to ab$

$bC \to bc$

$bB \to bb$

$cC \to cc$

حل:گرامر داده شده رشته هایی را تولید می کند که تعداد a ها و b ها و c ها با یکدیگر برابر باشند و ابتدا a ها، سپس b هـا و در نهایت c ها ظاهر شوند. مثلا رشته aabbcc .

$L = \{a^n b^n c^n \mid n \geq 0\}$

مثال: زبان تولید شده توسط گرامر زیر را تعیین کنید.

$S \to aSBCD \mid abcd$, $dB \to Bd$, $dC \to Cd$, $cB \to Bc$, $bB \to bb$, $cC \to cc$, $dD \to dd$

حل: $L = \{a^n b^n c^n d^n \mid n > 0\}$

مثال: گرامر سازنده زبان $L = \{a^n b^{n-3} : n \geq 3\}$ را بنویسید.

حل: زبان L را می توان به صورت $L = \{a^{k+3} b^k : k \geq 0\}$ نیز نشان داد. که در این صورت گرامر به راحتی بدست می آید:

$S \to aaaA$

$A \to aAb \mid l$

مثال: گرامر سازنده زبان $L=\{w:|w| \mod 3>0\}$ را بنویسید.

حل: باقیمانده تقسیم صحیح هر عدد به عدد سه، 0 یا 1 و یا 2 می باشد که حالتهای بیشتر از صفر آن، عبارتند:

حالت اول: $|w| \mod 3=1$: $S_1 \to aaaS_1 \mid a$

حالت دوم : $|w| \mod 3=2$: $S_2 \to aaaS_2 \mid aa$

بنابراین گرامر نهایی $S \to S_1 \mid S_2$ می باشد. ■

مثال: گرامری بنویسید که زبان $L_1 \cup L_2$ را تولید کند. $L_1=\{a^n b^n c^m \mid n,m>0\}$, $L_2=\{a^n b^m c^m \mid n,m>0\}$

حل : گرامر هر یک از زبانها در زیر آورده شده است:

G_1:
$$S_1 \to AB$$
$$A \to aAb \mid ab$$
$$B \to cB \mid c$$

G_2:
$$S_2 \to XY$$
$$X \to aX \mid a$$
$$Y \to bYc \mid bc$$

بنابراین گرامر $L_1 \cup L_2$ به صورت $S \to S_1 \mid S_2$ می باشد.

مثال: برای زبان $L=\{a^n b^m \mid n \neq m\}$ گرامر بنویسید.

حل: دو حالت $n>m$ و $n<m$ را بررسی کرده و سپس آن دو را ترکیب می‌کنیم:

$n>m$	$n<m$
$S \to AS_1$ $S_1 \to aS_1b \mid \lambda$ $A \to aA \mid a$	$S \to S_1B$ $S_1 \to aS_1b \mid \lambda$ $B \to bB \mid b$

در نهایت:

$$S \to AS_1 \mid S_1B \quad , \quad S_1 \to aS_1b \mid \lambda \quad , \quad A \to aA \mid a \quad , \quad B \to bB \mid b$$

مثال : گرامری را برای $\Sigma=\{a,b\}$ پیدا کنید که همه رشته های دارای حداقل سه a را تولید کند.

حل: ابتدا سه a تولید کرده و سپس تعداد دلخواهیa و b را در هر جای دلخواه از رشته اضافه می کنیم:

$$S \to AaAaAaA$$
$$A \to aA \mid bA \mid l$$

مثال : آیا گرامرهای زیر هم ارز می باشند؟

$$G_1: S \to aSb \mid bSa \mid SS \mid a$$
$$G_2: S \to aSb \mid bSa \mid a$$

حل: خیر- چون رشته aa توسط گرامر اول اشتقاق می شود، اما توسط گرامر دوم اشتقاق نمی شود.

مثال : گرامری برای مجموعه اعداد صحیح در C پیدا کنید. (بدون محدویت برای تعداد ارقام)

حل:

$$S \to AB$$
$$A \to + \mid - \mid l$$

$B \rightarrow D \mid DB$

$D \rightarrow 0|1|2|3|4|5|6|7|8|9$

مثال :

گرامر	زبان
$S \rightarrow aSa \mid aBa$ $B \rightarrow bB \mid b$	$L = \{a^n b^m a^n \mid n, m \geq 1\}$
$S \rightarrow abScd \mid abcd$	$L = \{(ab)^n (cd)^n \mid n > 0\}$
$S \rightarrow aaSb \mid aab \mid \lambda$	$L = \{a^n b^m \mid n = 2m\}$
$S \rightarrow 0S \mid 0A$ $A \rightarrow 0A1 \mid \lambda$	$L = \{0^n 1^m \mid n > m \geq 0\}$
$S \rightarrow 0S0 \mid 1A0 \mid \lambda$ $A \rightarrow 1A0 \mid \lambda$	$L = \{0^n 1^m 0^k \mid n + m = k\}$,(n,m,k$\geq$ 0)
$S \rightarrow aSa \mid bSb \mid c$	$L = \{wcw^R \mid w \in \{a, b\}^*\}$

انواع گرامرها

گرامرها را می توان به چهار دسته تقسیم کرد. این انواع عبارتند از:

۱- منظم (regular)

گرامری که خطی از راست یا خطی از چپ باشد. گرامری که همه قواعد آن به صورت $A \rightarrow Bx \mid x$ باشد را خطی از چپ می گویند و گرامری که همه قواعد آن به صورت $A \rightarrow xB \mid x$ باشد را خطی از راست می گویند.($A, B \in V$ و $x \in T^*$)

تذکر: در تمامی گرامرهای منظم، حداکثر یک متغیر در سمت راست هر یک از قوانین قرار می گیرد. که این متغیر باید همواره آخرین سمبل از سمت راست یا از سمت چپ، طرف راست هر یک از قوانین باشد.

۲- مستقل از متن (context free)

گرامری که در سمت چپ کلیه قواعد آن، فقط یک غیر پایانه (متغیر) باشد.

تذکر: هر گرامر منظم، مستقل از متن نیز می‌باشد.(عکس آن همواره صادق نیست.)

۳- حساس به متن (context sensitive)

گرامری که تمامی قوانین آن به فرم $x \rightarrow y$ باشند که در آن x وyعضو $(V \cup T)^+$ باشند و $|x| \leq |y|$.

تذکر: طبق تعریف بالا، قاعده $x \rightarrow l$ غیر مجاز است. بنابراین گرامرهای حساس به متن هرگز قادر به تولید زبانهای دارای رشته تهی نمی باشند.

تذکر: نام دیگر این گرامر، گرامر وابسته به متن می باشد.

۴- بدون محدودیت (unrestricted)

گرامری که تمامی قوانین آن به فرم $x \to y$ باشند که در آن ، x عضو $(V \cup T)^+$ و y عضو $(V \cup T)^*$ می باشد. در گرامرهای بدون محدودیت، اساسا هیچ شرط و محدودیتی برای قواعد تولید قائل نمی شویم. بعلاوه هر تعداد غیر پایانی و پایانی را مـی تـوان با هر ترتیبی در طرفین راست و چپ قرار داد.

تذکر: در گرامرهای بدون محدودیت، l نمی تواند در سمت چپ قواعد تولید رخ دهد.

ماشین (Automaton)

ماشین بر دو نوع است:

۱- پذیرنده (Acceptor) :ماشین هایی که برای پذیرش زبان ها استفاده می شوند.

۲- تراگذار(Transducer):ماشین هایی که برای محاسبه استفاده می شوند.

انواع ماشین ها

۱-ماشین متناهی (Finite Automata)

ماشین پذیرنده ای که حافظه ندارد و خروجی آن دارای دو حالت پذیرش یا عدم پذیرش است.

۲- ماشین پشته ای(Pushdown Automata)

ماشین پذیرنده ای که حافظه ای آن به صورت پشته بوده و خروجی آن دارای دو حالت پذیرش یا عدم پذیرش است.

۳- ماشین کراندار خطی (Linear Bounded Automata)

ماشینی که دارای حافظه از دو سر محدود با قابلیت خواندن و نوشتن باشد.

۴- ماشین تورینگ(Turing Machine)

ماشینی که دارای حافظه نامحدود با قابلیت خواندن و نوشتن باشد.

- ماشین های کراندار خطی و تورینگ را می توان هم به عنوان پذیرنده و هم به عنوان تراگذار به کار گرفت.
- ماشین متناهی، قادر به پذیرش زبان های منظم هستند.
- ماشین پشته ای، قادر به پذیرش زبان های مستقل از متن هستند.
- ماشین کراندار خطی، قادر به پذیرش زبان های حساس به متن هستند.
- ماشین های تورینگ، قادر به پذیرش زبان های بازگشتی شمارش پذیر هستند.

تذکر: تعریف کامل انواع هر یک از زبان ها، گرامرها و ماشین ها در فصل مرتبط با آن آورده شده است.

مجموعه تست

۱- گرامر G و رشته های w_1 و w_2 به شرح زیر مفروض اند:

$S \to acBdeA \mid BAB$

$B \to aSb \mid ae \mid \varepsilon$

$A \to aAb \mid b \mid \varepsilon$

$w_1 = acaacabbdebdeb$

$w_2 = acaacaaeebdebbdeabb$

کدام گزینه صحیح است؟

1) $w_1 \in L(G),\ \ w_2 \in L(G)$
2) $w_1 \notin L(G),\ \ w_2 \in L(G)$
3) $w_1 \in L(G),\ \ w_2 \notin L(G)$
4) $w_1 \notin L(G),\ \ w_2 \notin L(G)$

۲- گرامر وابسته به متن G مفروض است. زبان گرامر G کدام است؟

$G:$

$S \to S_1B$

$S_1 \to aS_1b$

$bB \to bbbB$

$aS_1b \to aa$

$B \to e$

1) $\{a^{n+1}b^{n+k} \mid n \geq 1, k \geq 0\}$
2) $\{a^n b^k \mid n \geq 2, k \geq 0\}$
3) $\{a^n b^{n+2k} \mid n \geq 2, k \geq 0\}$
4) $\{a^{n+1}b^{n+2k-1} \mid n \geq 1, k \geq 0\}$

۳- گرامر G را درنظر می گیریم و زبان آن را L می نامیم. رشته های w_1 و w_2 با تعریف زیر را نیز در نظر مـــی گیـــریم. کدام گزاره صحیح است؟

$G: S \to aSD \mid bB$

$D \to dS \mid a$

$B \to bB \mid \varepsilon$

$w_1 = a^{10}ba^7bdb^{10}d$

$W_2 = a^{10}b^9a^{10}d$

1) $w_1, w_2 \in L$
2) $w_1, w_2 \notin L$
3) $w_2 \notin L, w_1 \in L$
4) $w_1 \notin L, w_2 \in L$

۴- گرامر G و زبان های L_1 و L_2 مفروضند. ارتباط L(G) با L_1 و L_2 کدام است؟

(ε نشانه رشته ای به طول صفر است.)

$S \to S\ a\ b$

$S \to S\ b\ a$

$S \to a\ S\ b$

$S \to b\ S\ a$

$S \to a\ b\ S$

$S \to b\ a\ S$

$S \to \varepsilon$

$L_1 = \{w \in (a+b)^* \mid$ **تعداد aهای w با bهای w برابر است }**

$L_2 = \{w \in (a+b)^* \mid$ **تعداد abهای w با baهای w برابر است }**

1) $L(G) \subset L_1$
2) $L(G) = L_1$
3) $L(G) \subset L_1 \cup L_2$
4) $L(G) \supset L_2$

۵- گرامر G به شرح زیر مفروض است. $L(G)$ کدام است؟

(w^R عبارت است از w که از آخر به اول خوانده می شود و ε نشانه رشته های به طول صفر است.)

$G:$
$S \to aA$
$S \to bB$
$S \to \varepsilon$
$A \to Sa$
$A \to \varepsilon$
$B \to Sb$
$B \to \varepsilon$

1) $(a+b)^*$

2) $\{w \in (a+b)^* \mid w = w^R\}$

3) $\{w(a+b)w^R \mid w \in (a+b)^*\}$

4) $\{ww^R \mid w \in (a+b)^*\}$

۶- گرامر G به شرح زیر مفروض است. کدام یک از مجموعه رشته های ۱ تا ۴ ، زیر مجموعه L(G) است؟

$S \to ACaB$
$Ca \to aaC$
$CB \to DB$
$CB \to E$
$aD \to Da$
$AD \to AC$
$aE \to Ea$
$AE \to a$

1) $\{aa,aaaa\}$

2) $\{aaa,aaaaa\}$

3) $\{a,aaa,aaaaa\}$

4) $\{aaaa,aaaaaa\}$

۷- اگر $\Sigma = \{a,b,c\}$ و $L - \Sigma^* = \varphi$ باشد، آنگاه L کدام یک از زبان های زیر می تواند باشد؟

$\varepsilon - IV$, $\varphi - III$, $a^n b^{n^2} c^n - II$, $\Sigma^* - I$

1) فقط I
2) فقط IV
3) فقط III,I
4) IV,III,II,I

۸- زبان گرامر G کدام است؟

$G: S \to aAb \mid bBa \mid bCa$
$A \to aaAb \mid ab$
$B \to bBa \mid a$
$C \to aC \mid bC$

1) $k \geq 0$ $\quad a^{2k+2}b^{k+1} \cup b^+a^+$

2) $k \geq 0$ $\quad a^{2k}b^k \cup (ba)^*a$

3) $l \geq 1, k \geq 2$ $\quad a^{k+1}b^k \cup b^l a^l$

4) $k \geq 0, l \geq 1$ $\quad a^2 a^{2k} b^k b^2 \cup b^l a^{l+1}$

۹- برای هر دو زبان B,A زبان $(A \cup B)^*$ برابر است با:

1) AB^*A
2) $A^* \cup B^*$
3) $A^*(BA^*)^*$
4) $A^*B^* \cup B^*A^*$

۱۰- اگر $L = \{a^n b^n \mid n \geq 0\}$، آنگاه زبان L^* را می‌توان با کدام گرامر توصیف کرد؟

1) $S \to aS \mid bS \mid \lambda$

2) $S \to SA \mid \lambda$
$A \to aAb \mid \lambda$

3) $S \to aSb \mid \lambda$

4) $S \to SS \mid A$
$A \to aAb \mid ab$

۱۱- توصیفی معادل جهت زبان $L=1+0(0+10)^*11$ برابر است با:

1) $(00^*1)1^*$
2) $(00^*1)^*1$
3) $(0^*1)^*1$
4) $(0^*1)1$

۱۲- کدام یک از گرامرهای زیر، زبان $L=\{a^n b^m c^k \mid k=|n-m|\}$ را تولید می نماید؟

1)
$S \to A \mid B$
$A \to aAc \mid C$
$C \to aAc$
$B \to DE$
$D \to aDb$
$E \to \lambda$

2)
$S \to A \mid B$
$A \to aBc$
$C \to aAc$
$B \to DE$
$D \to aDb$
$E \to bEc \mid \lambda$

3)
$S \to A \mid B$
$A \to aAc \mid C$
$C \to aCb \mid \lambda$
$B \to DE$
$D \to aDb \mid \lambda$
$E \to bEc \mid \lambda$

4)
$S \to A \mid B$
$A \to ac \mid C$
$C \to aC \mid Cb$
$B \to DE \mid \lambda$
$D \to aDB \mid \lambda$
$E \to bEc \mid \lambda$

پاسخ تشریحی

۱-۲) با توجه به گرامر زیر و دو رشته داده شده، مشخص است که $w_1 \notin L(G),\ \ w_2 \in L(G)$.

$S \to acBdeA \mid BAB$

$B \to aSb \mid ae \mid \varepsilon$

$A \to aAb \mid b \mid \varepsilon$

$w_1 = acaacabbdebdeb$

$w_2 = acaacaaeebdebbdeabb$

۲-۴) گرامر داده شده رشته aa را تولید می کند که فقط توسط زبانهای گزینه 2و4 پذیرش می شوند. از طرفی رشته a^2b توسط زبان گزینه 2 تولید می شود که متعلق به گرامر نمی باشد. بنابراین گزینه 4 صحیح است.

راه حل کلی:

اگر در قاعده $S \to S_1B$ به جای S_1 از aS_1b استفاده کنیم و این عمل را n مرتبه تکرار کنیم به $a^nS_1b^nB$ می رسیم. حال اگر به جای aS_1b از aa استفاده کنیم، به رابطه $a^{n+1}b^{n-1}B$ می رسیم. در نهایت به جای bB از $bbbB$ استفاده کرده و این عمل را k مرتبه تکرار می کنیم و عبارت $a^{n+1}b^{n+2k-1}$ حاصل می شود.

۳-۲) رشته های w_1 و w_2 هر دو به d ختم می شوند و این گرامر رشته هایی که به d ختم می شوند را نمی تواند تولید کند.

۴-۱) توسط گرامر رشته ababab را می توان تولید کرد که در آن تعداد ab ها با تعداد ba ها برابر نمی باشند. همچنین رشته aababbbbaa که به زبانهای L_1 و L_2 تعلق دارد، توسط گرامر قابل تولید نمی باشد. بنابراین گزینه های 2و3و4 نادرست می باشند.

۵-۲) اگر چند رشته قابل تولید توسط گرامر را بررسی کنیم، متوجه خواهیم شد که جمله های زبان این گرامر به صورت $\{w \in (a+b)^* \mid w = w^R\}$ می باشد مانند abaکه از هر دو طرف یکسان خوانده می شود.

۶-۲) اگر یک اشتقاق ساده را دنبال کنیم به aaa می رسیم:

$$S \to ACaB \to AaaCB \to AaaE \to AaEa \to AEaa \to aaa$$

۷-۴)

$$L - \sum^* = L \cap (\overline{\sum^*}) = \varnothing \Rightarrow L \cap \varnothing = \varnothing$$

از آنجا که اشتراک هر زبانی با زبان تهی، زبانی تهی می باشد، بنابراین زبان L می تواند هر زبانی باشد. بنابراین زبان L هر چهار زبان داده شده می تواند باشد.

۸-۴) رشته a^2b^2 متعلق به گرامر، فقط توسط گزینه 4 قابل تولید است.

۹-۳) گزینه 1 نادرست است، چون باید حتما با A شروع شود. رشته ABA را نمی توان توسط گزینه 2و4 تولید کرد، بنابراین نادرست می باشند.

۱۰-۲) گرامر گزینه یک زبان $(a+b)^*$را تولید می کند. گزینه 3 زبان L را تولید می کند نه L^*. گزینه 4 رشته l را تولید نمی کند.

۱۱-۲) رشته 1 توسط گزینه های 1و4 قابل تولید نمی باشد. همچنین گزینه 3 رشته 111 را تولید مـی‌کنـد کـه متعلـق بـه زبـان نمی‌باشد.

۱۲-۳) برای بدست آوردن گرامر زبان $L = \{a^n b^m c^k \mid \; k = |n - m|\}$ باید اجتماع دو حالـت k=n-m وk=m-n را در نظـر گرفـت. در گرامر زیر $S \to A$ حالت اول و $S \to B$ حالت دوم را بررسی می کند:

$S \to A \mid B$
$A \to aAc \mid C$
$C \to aCb \mid \lambda$
$B \to DE$
$D \to aDb \mid \lambda$
$E \to bEc \mid \lambda$

فصل دوم: اتوماتای متناهی قطعی(DFA)

ماشین‌ها ابزارهایی هستند برای تشخیص رشته‌های زبان که رشته را از چپ به راست بررسی کرده و نهایتاً اعلام مـی‌کننـد کـه آیـا رشته متعلق به زبان هست یا نه. ماشین‌ها را می‌توان به عنوان مدلهای ریاضی برای کامپیوترهای واقعی در نظر گرفت. یکی از ساده ترین انواع ماشین ها، ماشین متناهی می باشد که از آن در شناخت زبان های منظم استفاده می شود. ماشین‌های متنـاهی(FA) ، به دو دسته قطعی (DFA) و غیر قطعی (NFA) تقسیم می‌شوند.

پذیرنده متناهی معین (DFA)

DFA : Deterministic Finite Accepter

ماشین DFA از سه بخش تشکیل شده است: نوار ورودی، هد و کنترل متناهی. در این ماشین جمله ورودی روی نوار قـرار دارد و توسط هد خوانده می‌شود، اگر در لحظه‌ای که خواندن جمله ورودی پایان یافت، ماشین در یک وضعیت نهـایی قـرار داشـته باشـد، آنگاه جمله پذیرفته شده است. به عبارتی DFA بوسیله پنج تایی $M=(Q,\sum,\delta,q_0,F)$ تعریف می شود که در آن:

Q: مجموعه متناهی ازحالات داخلی $\sum$: الفبای ورودی

d: تابع انتقال ($Q\times\sum\to Q$) q_0 : حالت شروع

F : مجموعه حالات نهایی

عبارت $\delta(q_i,a)=q_j$ ، یعنی اگر ماشین در وضعیت q_i باشد و a مقابل هد نوار خوان باشد، آنگـاه ماشـین بـه حالـت q_j تغییـر حالت می دهد.

تذکر: منظور از عبارت $(q,w)\mid\frac{*}{M}(q',w')$ به این معنی است که ماشین M طی صفر یا چند تحول، می‌توانـد از پیکربنـدی (q,w) به (q',w') برسد.

تذکر: رشته w توسط ماشین M پذیرفته می‌شود را به صورت $(S,w)\mid\frac{*}{M}(q,\lambda)$ نمایش می دهیم.(q یکی از حالات نهایی است.)

هد نوار در ماشین متناهی، فقط به سمت راست حرکت می‌کند.

مثال : نشان دهید که رشته W=aba متعلق به زبان ماشین M است.

$M=(Q,\sum,\delta,q_0,F)$, $\sum=\{a,b\}$, $Q=\{q_0,q_1\}$, $F=\{q_1\}$

$\delta(q_0,a)=q_1$, $\delta(q_0,b)=q_0$, $\delta(q_1,a)=q_1$, $\delta(q_1,b)=q_0$

حل: نمایش تعلق رشته aba به زبان ماشین M:

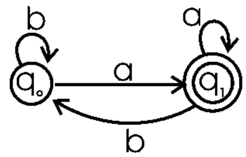
a	b	a		…
$\uparrow$ q_0				

a	b	a		…
	$\uparrow$ q_1			

a	b	a		…
		$\uparrow$ q_0		

a	b	a		…
			$\uparrow$ $q_1\in F$	

چون زمانی که هد بر روی فضای خالی بعد از رشته است، در حالت نهایی (q_1) هستیم، بنابراین رشته پذیرفته می شود. نمودار تغییر وضعیت DFA این مثال در شکل زیر نشان داده شده است:



تذکر: حالت q_1 نهایی است. حالت نهایی به صورت دوایری دو خطی نمایش داده می شود.

نحوه پذیرش رشته توسط ماشین:

این ماشین آنقدر در حالت شروع یعنی q_0 باقی می ماند تا با اولین a برخورد کند. این DFA برای پذیرش رشته aba، با شروع از حالت q_0، ابتدا سمبل a را می خواند. با نگاهی به یال های گراف مشاهده خواهیم کرد که ماشین به حالت q_1 می رود. سپس، سمبل b خوانده شده و ماشین به حالت q_0 می رود. در نهایت سمبل a خوانده شده و ماشین به حالت q_1 می رود. در این لحظه، هم در پایان رشته و هم در حالت پایانی قرار داریم. بنابراین رشته aba پذیرفته می شود.

با استدلالی مشابه، مشخص می شود که ماشین رشته های a و aaa و aabbba را پذیرفته ولی abb یا bab را رد می کند.

تابع انتقال گسترش یافته

تابع انتقال گسترش یافته به صورت $d^* : Q\times\Sigma^* \to Q$ تعریف می شود. آرگومان دوم این تابع، یک رشته است نه یک سمبل. به عنوان مثال اگر $d(q_0,a)=q_1$ و $d(q_1,b)=q_2$ آنگاه: $d^*(q_0,ab)=q_2$.

زبان ها و dfa ها

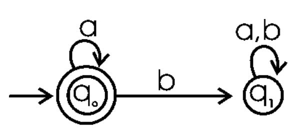
می توان گفت که زبان مجموعه ای از تمام رشته های پذیرفته شده توسط اتومات می باشد.

تعریف : زبان پذیرفته شده توسط dfa ی $M=(q,\Sigma,d,q_0,F)$ ، مجموعه تمام رشته های روی Σ است که توسط M پذیرفته می شوند. فرم صوری آن به صورت $L(M)=\{w\in\Sigma^* : d^*(q_0,w)\in F\}$ است.

تعریف: dfa پس از پردازش هر یک از رشته های Σ^* ، یا آنها را می پذیرد و یا رد می کند. عدم پذیرش به این معنی است که dfa در یکی از حالت های غیر پایانی متوقف شده و در نتیجه : $\overline{L(M)}=\{w\in\Sigma^* : d(q_0,w)\notin F\}$

حالت دام یا تله (trap state)

می خواهیم ماشینی رسم کنیم که زبان a^* را روی الفبای $\{a,b\}$ بپذیرد. برای این کار یک DFA با یک حالت رسم می کنیم که هم حالت شروع و هم حالت پایانی است به طوری که یال با برچسب a از آن حالت خارج شده و به همان حالت وارد می شود. در این ماشین اگر قبل یا بعد از حرف a، حرف b ظاهر شود، رشته نباید پذیرفته شود، بنابراین یک حالت دیگر به نام q_1 رسم کرده و با یالی با برچسب b به آن حالت می رویم که خروج از آن نیز ممکن نمی باشد. در شکل زیر این DFA رسم شده که وضعیت q_1 همان وضعیت تله است:



تذکر: در این DFA رشته λ نیز پذیرفته می‌شود چون q_0 حالت پایانی نیز می‌باشد.

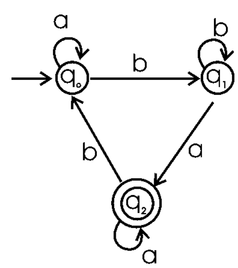
استفاده از جدول برای نمایش ماشین

هر چند گراف ها برای نمایش ماشین بسیار مناسب هستند، می توان از روشهای دیگر نیز استفاده کرد. بعنوان مثال می تـوان تـابع انتقال را به صورت جدول ارائه کرد. در این جدول نام سطر بیانگر حالت فعلی و نام ستون بیـانگر سـمبل ورودی فعلـی مـی باشـد. درایه های جدول هم معرف حالت بعدی خواهد بود.

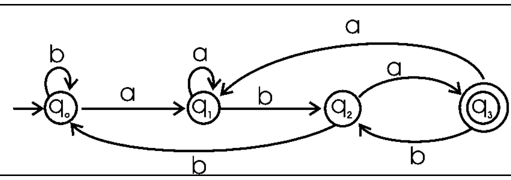
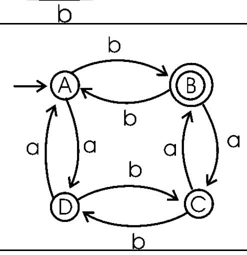
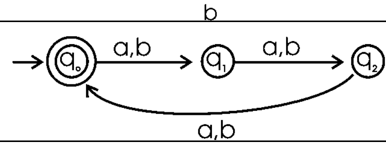
مثال : با توجه به تابع مبدل داده شده، DFA را رسم نمایید. (q_2 حالت پایانی است.)

	a	b
q_0	q_0	q_1
q_1	q_2	q_1
q_2	q_2	q_0

حل:



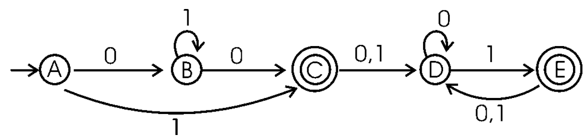
مثال : در جدول زیر DFA پذیرنده هر زبان در مقابل آن آورده شده است: (در الفبای $\sum=\{a,b\}$)

رشته‌هایی که به aba ختم می‌شود.	
تعداد زوجی a و تعداد فردی b را بپذیرد.	
رشته‌هایی که طول آنها 3 باشد.	

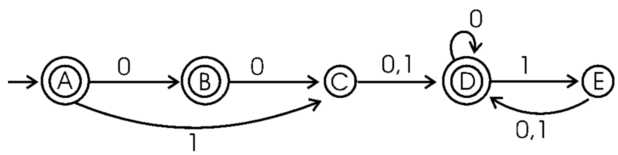
مکمل DFA

برای رسم مکمل یک DFA، کافی است که کلیه حالتهای غیر پایانی را به حالت پایانی و کلیـه حـالات پایـانی را بـه حـالات غیـر پایانی تبدیل کنیم.

مثال : ماشین مکمل DFA زیر را رسم نمایید.



حل:

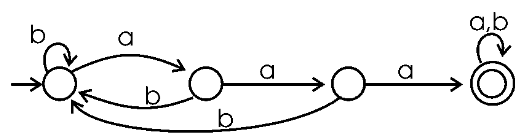


اگر $M=(Q,\sum,d,q_0,F)$ و $\hat{M}=(Q,\sum,d,q_0,Q-F)$ دو DFA باشند، آنگاه: $\overline{L(M)}=L(\hat{M})$

مثال : DFA پذیرنده زبان L را رسم نمایید.($\sum=\{a,b\}$)

L = {w | زیر رشته‌ای از w باشد aaa}

حل:



مثال : DFA ای رسم نمایید که زبان L را بپذیرد.

L = {w | در آن وجود ندارد aaa}

حل: کافی است DFA مثال قبل را مکمل کنیم.

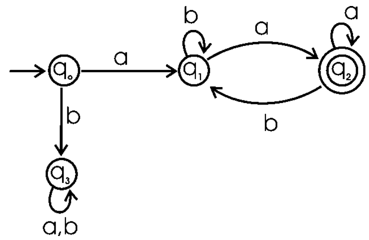
زبان های منظم

هر ماشین متناهی زبان خاصی را می پذیرد. بنابراین اگر همه ماشین های متناهی ممکن را در نظر بگیریم، یک مجموعه زبان متناظر با آنها وجود دارد. این مجموعه زبان ها خانواده نامیده می شوند. خانواده زبان هایی که توسط DFA پذیرفته می شوند بسیار محدود است.

تعریف: زبان L منظم است اگر و فقط اگر یک DFA مانند M وجود داشته باشد به طوریکه L = L(M).

مثال : نشان دهید که زبان $L=\{awa \mid w\in\{a,b\}^*\}$ منظم است.

حل: برای اثبات منظم بودن زبان داده شده ، کافی است که یک DFA برای آن پیدا کنیم. زبان L رشته هایی را می پذیرد که با حرف a شروع شده و با حرف a نیز تمام می شود. بین این دو حرف a می تواند هیچ یا تعداد نامحدودی حرف a یا b ظاهر شود. در این DFA که در شکل زیر رسم شده، وضعیت q_3 یک تله است و اگر رشته با حرف b شروع شود، ماشین به این حالت می رود.



مثال : نشان دهید که زبان $L^2 = \{aw_1aaw_2a : w_1, w_2 \in \{a,b\}^*\}$ منظم است.

حل: با رسم یک DFA برای این زبان می توان نشان داد که L^2 نیز منظم است. بـرای رسـم ایـن DFA ، کـافی اسـت کـه در ماشین مثال قبل، حالت هـای q_4 و q_5 را بعـد از q_2 اضـافه کنـیم و از q_2 یـالی بـه q_4 بـا برچسـب a وصـل کنـیم. (حالـت q_4 معادل q_1و حالت q_5 معادل q_2 می باشد.) (همچنین q_2 دیگر حالت نهایی نبوده و q_5 نهایی می باشد.) ■

مثال : برای هر یک از زبان های زیر می توان یک DFA رسم کرد، بنابراین منظم می باشند:

$L = \{vwv : v, w \in \{a,b\}^*, |v| = 2\}$

$L = \{a^n : n \geq 4\}$

$L = \{a^n : n \geq 0, n \neq 4\}$

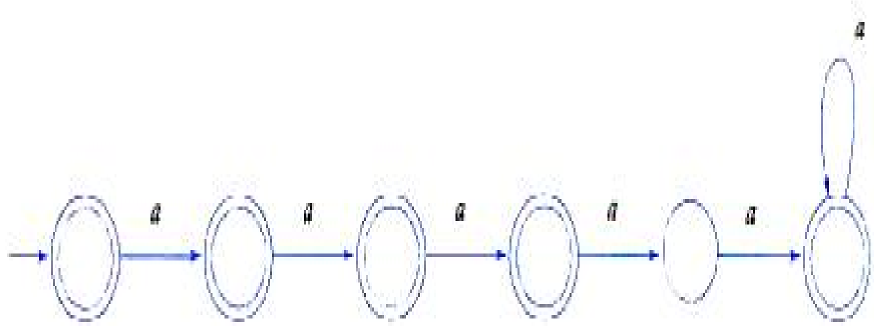
$L = \{a^n :$ n یا ضریب سه است یا ضریب پنج $\}$

$L = \{a^n :$ nضریب سه است و ضریب پنج نیست $\}$

- اگر زبان مفروض L منظم باشد، L^2 و L^3و ... هم منظم خواهد بود.
- تمام زبانهای متناهی ، منظم هستند.
- اگر L منظم باشد، L^* هم منظم خواهد بود.

مثال : نشان دهید که زبان $L = \{a^n : n \geq 0, n \neq 4\}$ منظم است؟

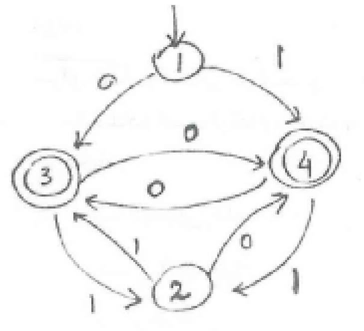
حل:می توان برای آن یک dfa رسم کرد، بنابراین منظم است:



■

مجموعه تست

۱- ماشین متناهی M با ساختار زیر را در اختیار داریم. کدام عبارت منظم معادل $L(M)$ است؟



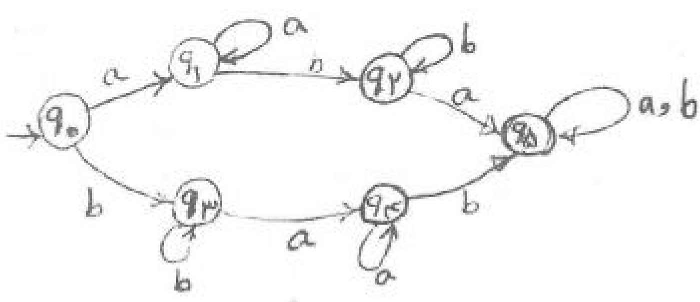
1) $(0\,|\,1)(011\,|\,010)^*$

2) $(0\,|\,1)(0\,|\,010\,|\,011)^*$

3) $(1\,|\,0)(011\,|\,11\,|\,10\,|\,0)^*$

4) $(0(011)^*\,|\,1(011)^*\,|\,0(10)^*\,|\,0(011)^*)^*$

۲- ماشین مقابل چه زبانی را معرفی می کند؟



1) رشته هایی به صورت $(a+b)^*(abba+baab)(a+b)^*$

2) تمام رشته هایی که هم شامل زیر رشته ab و هم زیر رشته ba هستند.

3) رشته هایی که با a شروع می شوند و تناوبی ab دارند یا رشته های که با b شروع می شوند و تناوبی ba دارند.

4) رشته هایی به صورت $w(a+b)^*+\overline{w}(a+b)^*$ ، $\overline{w}$ همان w است که هر a با b و هر b با a جایگزین شده.

۳- L زبانی است با الفبای $\Sigma=\{0,1\}$ به قسمی که کلیه رشته های L دارای حداقل یک زیر رشته ۱۱ و فاقد زیر رشته ۰۰ هستند. کوچک ترین آتاماتایی که این زبان را شناسایی کند دارای چند وضعیت (حالت) است؟

1) 5 وضعیت که 2 وضعیت آن از نوع شناسایی است.

2) 6 وضعیت که 2 وضعیت آن از نوع شناسایی است.

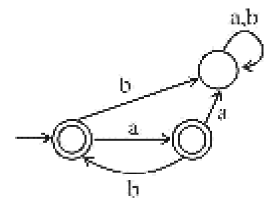
3) 5 وضعیت که 3 وضعیت آن از نوع شناسایی است.

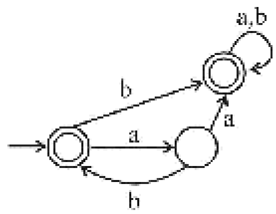
4) 6 وضعیت که 3 وضعیت آن از نوع شناسایی است.

۴- یک DFA برای زبان منظم $L = (a^*(b \cup \{\lambda\})a^*)^*$ عبارت است از:

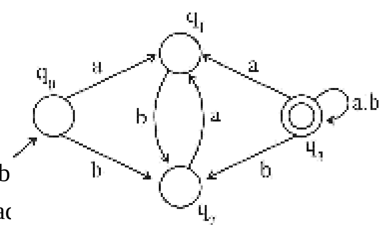
1) 

2) 

3) 

4) 

۵- با فرض آن که ماشین متناهی قطعی زیر را داشته باشیم، گرامر متناظر با این ماشین برابر است با:



1)
$q_0 \to aq_1 \mid bq_2 \mid \lambda$
$q_1 \to aq_1 \mid bq_2$
$q_2 \to bq_3 \mid bq_0 \mid \lambda$
$q_3 \to aq_3 \mid bq_3 \mid \lambda$

2)
$q_0 \to aq_1aq_3 \mid bq_2b$
$q_1 \to bq_2aq_1 \mid aq_3a$
$q_2 \to bq_3b \mid bq_3aq_3 \mid bq_1bq_2$
$q_3 \to aq_3bq_3 \mid \lambda$

3)
$q_0 \to aq_1 \mid bq_2$
$q_1 \to aq_2 \mid aq_3 \mid a$
$q_2 \to bq_1 \mid bq_3 \mid b$
$q_3 \to abq_3 \mid \lambda$

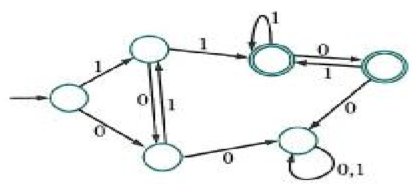
4)
$q_0 \to aq_1 \mid bq_2$
$q_1 \to bq_2 \mid aq_3 \mid a$
$q_2 \to aq_1 \mid bq_3 \mid b$
$q_3 \to aq_3 \mid bq_3 \mid a \mid b$

پاسخ تشریحی

۱-۳) عبارت منظم $(1|0)(011|11|10|0)^*$ معادل ماشین داده شده است.

۲-۲) با چک کردن چند رشته، مشخص است که ماشین داده شده تمام رشته هایی که هم شامل زیر رشته ab و هم زیر رشـته ba هستند را می پذیرد.

۳-۲) این ماشین دارای 6 وضعیت است که دو وضعیت آن از نوع شناسایی (پایانی) است:



۱-۴) زبان $L = (a^*(b \cup \{\lambda\})a^*)^*$ برابر $(a \cup b)^*$ می باشد که DFA آن به صورت زیر است:

۴-۵) گرامر DFA داده شده برابر است با:

$$q_0 \to aq_1 \mid bq_2$$
$$q_1 \to bq_2 \mid aq_3 \mid a$$
$$q_2 \to aq_1 \mid bq_3 \mid b$$
$$q_3 \to aq_3 \mid bq_3 \mid a \mid b$$

فصل سوم: اتوماتای متناهی غیر قطعی(NFA)

پذیرنده های متناهی نامعین(غیر قطعی)، پیچیده تر از انواع معین خود هستند. نامعین بودن باعث می شود تا بتوان حرکات ماشین را انتخاب کرد. در هر حالت می توان مجموعه ای از حرکات مجاز را انتخاب کرد. به عبارتی به ازای دریافت یک ورودی در هر حالت، ماشین می تواند به چندین حالت مختلف تغییر حالت دهد.

NFA : Nondeterministic Finite Accepter

تعریف: یک پذیرنده متناهی نامعین (NFA) بوسیله پنج تایی $M=(Q,\sum,\delta,q_0,F)$ تعریف می شود که در آن Q و $\sum$ و q_0 و F همانند DFA تعریف می شوند، ولی تابع انتقال به صورت $Q\times\sum \mathbf{U}\{l\}\rightarrow 2^Q$ تعریف می شود. برد این تابع مجموعه ای از حالات مجاز برای ماشین می باشد.

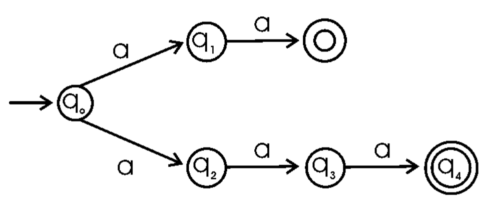
سه تفاوت عمده بین تعریف nfa و تعریف dfa وجود دارد. در nfa:

1- محدوده تابع δ در مجموعه توانی 2^Q است. بنابراین مقدار آن یک عنصر از Q نیست. مثلاً اگر وضعیت فعلی q_0 باشد و حرف a خوانده شود، آنگاه هریک از حالتهای q_1, q_2 می‌تواند وضعیت بعدی باشد: $\delta(q_0,a)=\{q_1,q_2\}$

2- λ بعنوان ورودی قابل قبول است. یعنی NFA می تواند بدون استفاده از سمبل ورودی، دست به انتقال بزند. هد می تواند در بعضی انتقال ها حرکت نکند.

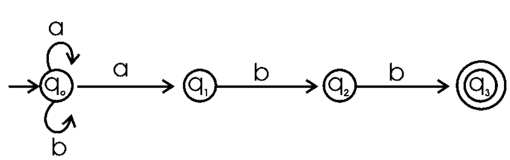
3- $d(q_i,a)$ می‌تواند تهی باشد، یعنی هیچ انتقالی برای این وضعیت خاص تعریف نشده است.

مثال : ماشین زیر یک NFA می باشد. چون دو انتقال با برچسب a از q_0 دارد:



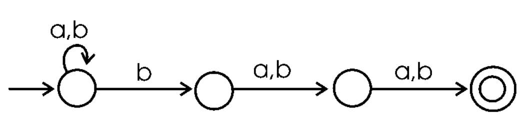
اولین انتخاب منجر به پذیرش تمام رشته های دارای تعداد زوجی از a می شود و دومین انتخاب منجر به پذیرش رشته a^3 می شود. بنابراین زبان پذیرفته شده ، $L=\{a^{2n}:n\geq 1\}\mathbf{U}\{a^3\}$ می باشد.

مثال : ماشین NFA زیر چه زبانی را می پذیرد؟ $(\sum=\{a,b\})$



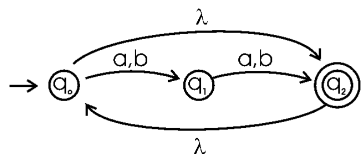
حل: زیر رشته‌هایی را که به زیر رشته abb ختم شوند.■

مثال : ماشین NFA زیر چه زبانی را می پذیرد؟ $(\sum=\{a,b\})$



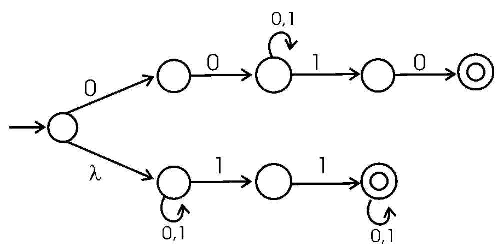
حل: رشته های که سومین نماد از سمت راست آنها، b باشد.■

مثال : ماشین NFA زیر چه زبانی را می پذیرد؟ $(\sum=\{a,b\})$



حل: ماشین داده شده، رشته‌هایی با طول زوج را می پذیرد.■

مثال : ماشین NFA زیر چه زبانی را می پذیرد؟ $(\sum=\{a,b\})$



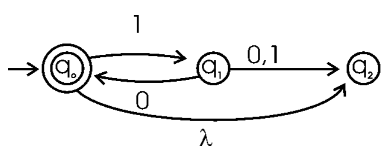
حل: رشته‌هایی که با 00 شروع و به 10 ختم شوند یا شامل زیر رشته 11 باشند. ■

تعریف: زبان پذیرفته شده توسط nfa ی $M=(q,\Sigma,\delta,q_0,F)$ ، توسط مجموعه ای از تمام رشته های پذیرفته شده بر اساس مفاهیم تابع انتقال گسترش یافته، تعریف می شود. به بیان دقیق،

$L(M)=\{w\in\Sigma^* : \delta^*(q_0,w)\cap F\neq\phi\}$

به عبارت دیگر، این زبان شامل تمام رشته های w است که به ازای آن، یک قدم با برچسب w از راس شروع گراف انتقال به حداقل یکی از رئوس پایانی وجود داشته باشد.

مثال : زبان پذیرفته شده توسط اتومات شکل زیر چیست؟

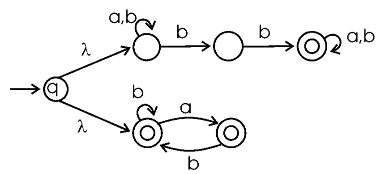


حل: با توجه به گراف، به راحتی می توان مشاهده کرد که تنها راه برای اینکه nfa داده شده در حالت پایانی قرار بگیرد، آن است که ورودی یا تکراری از رشته 10 باشد و یا رشته تهی. بنابراین، اتومات زبان $L=\{(10)^n : n\geq 0\}$ را می پذیرد. تذکر: این ماشین رشته 110 را نمی پذیرد. چون با خواندن پیشوند 11، به حالت q_2 رفته و انتقال تعریف نشده $\delta(q_2,0)$ را می بیند. این وضعیت را **پیکر بندی مرده** می نامند. اینکه با پردازش 110 نمی توان به هیچکدام از حالت های پایانی رسید را به صورت $\delta^*(q_0,110)=\phi$ نمایش می دهیم.■

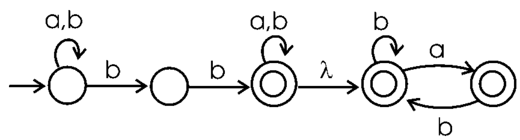
- زبان پذیرفته شده توسط nfaو dfa منظم است.
- به ازای هر nfa با چندین حالت شروع، یک nfa با دقیقا یک حالت شروع وجود دارد که همان زبان را می‌پذیرد.
- nfa ای که در آن هیچ انتقال λ وجود ندارد و به ازای هر $q\in Q$ و هر $a\in\Sigma$ ، $\delta(q,a)$حاوی حداکثر یک عضو باشد را dfa ناقص می گویند. در این dfa ها، امکان انتخاب هر انتقالی وجود ندارد و برای برخی انتقال ها نمی توان حرکتی کرد.
- اگر L یک زبان منظم فاقد λ باشد، یک nfa بدون انتقال λ و فقط با یک حالت پایانی وجود دارد که L را می پذیرد.

مثال : ماشین nfa ای برای پذیرش زبان $L(M_1)\cup L(M_2)$ طراحی نمایید، به طوریکه M_1، ماشین پذیرنده رشته‌های شامل زیر رشته bb و ماشین M_2، پذیرنده رشته‌هایی که شامل زیر رشته aa نباشند.

حل: در شکل زیر، ماشین M_1 در بالا و ماشین M_2 در پایین می باشد:



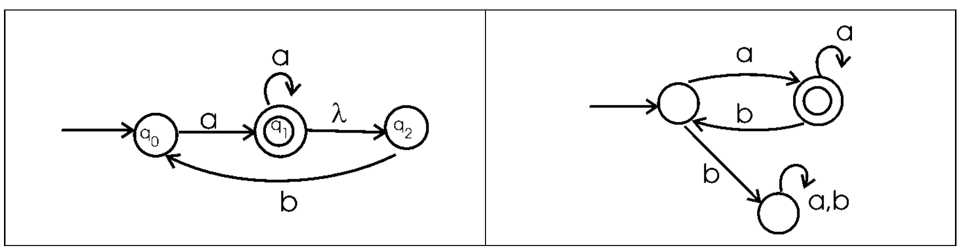
مثال : با توجه به مثال قبل، ماشینی طراحی نمایید که پذیرنده زبان $L(M_1).L(M_2)$ باشد.



هم ارزی DFAو NFA

دو ماشین متناهی را هم ارز می گویند اگر هر دو، زبانی یکسان را بپذیرند. قبلا گفتیم که به ازای هر زبان معمولا تعداد زیادی پذیرنده وجود دارد، بنابراین هر dfa یا nfa نیز تعداد زیادی پذیرنده هم ارز دارد.

مثال : در شکل زیر دو ماشین هم ارز نشان داده شده است(شکل سمت راستdfa و سمت چپ nfa است)



قضیه: به ازای هر زبانی که توسط یک nfa پذیرفته می شود، یک dfa هم وجود دارد که آن را می پذیرد.

کلاس های dfa ها و nfa ها دارای قدرت یکسان می باشند.

برای هر nfa با هر تعداد دلخواه حالت پایانی، یک dfa با فقط یک حالت پایانی، هم ارز با آن nfa وجود دارد.

روال تبدیل NFA به DFA

۱-گراف مفروض G_D با راس $\{q_0\}$را ایجاد کرده و آن راس را بعنوان راس شروع در نظر بگیرید.

۲- تا زمانی که همه یال ها در نظر گرفته نشده اند، مراحل زیر را تکرار کنید:

الف- هر یک از رئوس $\{q_i, q_j, ..., q_k\}$ از G_Dرا در نظر بگیرید که برای $a \in \Sigma$آن هیچ یالی خارج نشود.

ب- $d^*(q_i,a), d^*(q_j,a), ..., d^*(q_k,a)$ را محاسبه کنید.

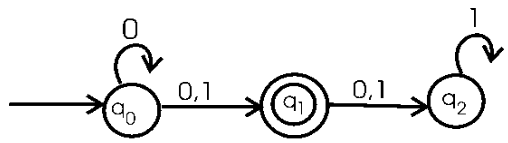
ج- اجتماع همه این d_N^* ها را تشکیل دهید در نتیجه مجموعه $\{q_1, q_m, ..., q_n\}$ بدست می آید.

د- در صورت عدم وجود راسی با برچسب $\{q_1, q_m, ..., q_n\}$ در G_D ، ایـن راس را ایجـاد کنیـد. یـالی از $\{q_i, q_j, ..., q_k\}$ بـه $\{q_1, q_m, ..., q_n\}$ به نام a به G_D اضافه کنید.

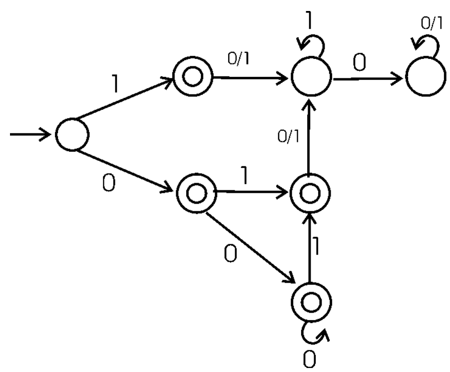
۳- تمامی حالت های G_D که برچسب آن حاوی حداقل یک $q_f \in F_N$ باشد، بعنوان راس پایانی شناخته می شود.

۴- اگر M_N ، l را بپذیرد، راس $\{q_0\}$ در G_D نیز، راس پایانی می باشد.

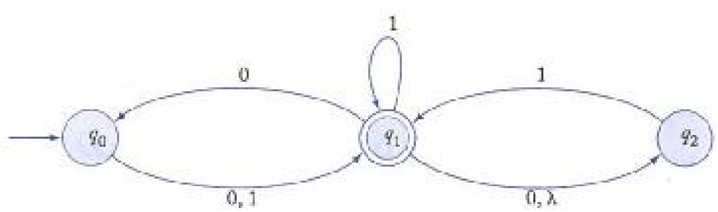
مثال : NFA زیر را به DFA تبدیل کنید.



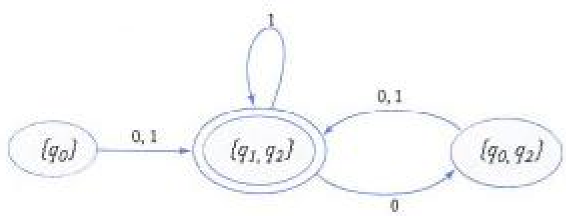
حل: با انجام روال تبدیل nfa به dfa به شکل زیر می رسیم:



مثال : NFA زیر را به DFA تبدیل کنید.



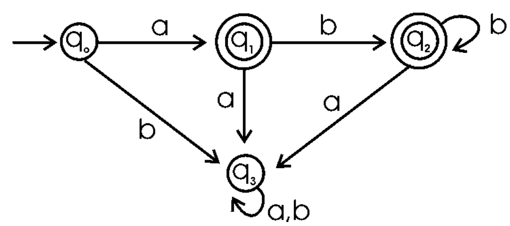
حل: با اجرای روال تبدیل nfa به dfa شکل زیر بدست می آید:



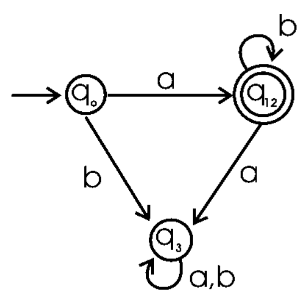
کاهش تعداد حالات در ماشین های متناهی

هر dfa یک زبان منحصر بفرد را تعریف می کند، اما عکس این جمله صحیح نیست. در عمل ممکـن اسـت از بـین چنـد dfa کـه برای یک زبان وجود دارد، یکی را انتخاب کرد. معمولا این dfa دارای حـالات کمتـری مـی باشـد. در dfa مـی تـوان حـالتی کـه دسترس پذیر نباشد را حذف کرد و بعضی از حالتها را که ادغام پذیر هستند را با هم ادغام کرد.

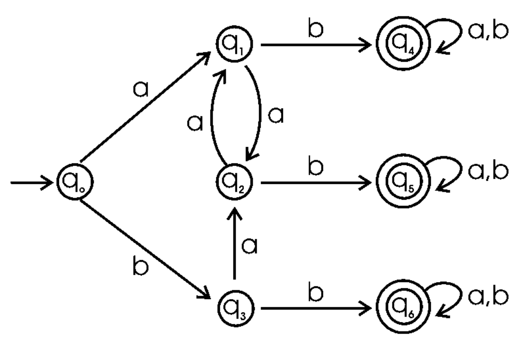
مثال : DFA زیر را کمینه نمایید.



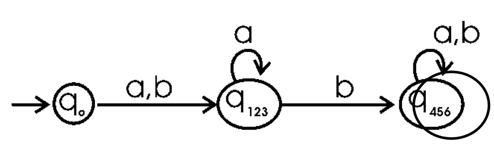
حل: حالت q_1 و q_2 ادغام پذیرند و به جای آنها حالت q_{12} را قرار می‌دهیم:



مثال : DFA زیر را کمینه نمایید.

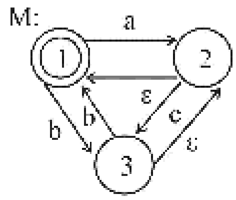


حل: حالت های q_1 و q_2 و q_3 ادغام پذیرند و به جای آنها حالت q_{123} را قـرار مـی‌دهـیم. همچنـین حالـت هـای q_4 و q_5 و q_6 ادغام پذیرند و به جای آنها حالت q_{456} را قرار می‌دهیم.



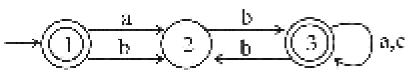
مجموعه تست

۱- ماشین متناهی M به شکل زیر مفروض است. گزاره صحیح کدام است؟



1) $L(M) = (a^* | (b | ac)^* (b | \varepsilon))^*$

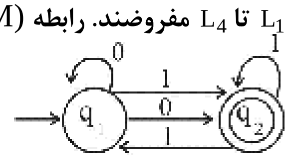
2) ماشین قطعی زیر معادل M است:



3) $L(M) = \{w \in (a | b | c)^* \mid$ w با c شروع نمی شود $\}$

4) زبان گرامر مقابل همان $L(M)$ است:
$$S \to aS | bS | acS | bA | acA$$
$$A \to cA | b | \varepsilon$$

۲- اتومات متناهی M و زبان های L_1 تا L_4 مفروضند. رابطه $L(M)$ با L_1 تا L_4 کدام است؟



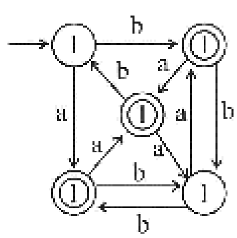
$L_1 = (0+1)(0+1)^*$
$L_2 = (0+(0+1)1^*1)^*(0+1)1^*$
$L_3 = 0^*(0+1)1^*(10^*(0+1)1^*)^*$
$L_4 = (0+110)(0+1)^*$

1) $L(M) = L_2 = L_3 = L_4$ 2) $L(M) = L_1 = L_2 = L_3$

3) $L(M) = L_2 = L_3$ 4) $L(M) = L_4$

۳- اتومات متناهی زیر را درنظر می گیریم. اتومات کمینه مربوطه دارای چند حالت خواهد بود؟

(شماره حالتها از بالا به پایین و از چپ به راست به صورت ۱۲۳۴۵ می باشد.)



1) 3 2) 2 3) 5 4) 4

پاسخ تشریحی

۱-۳) گزینه 1 نادرست است، چون acc که توسط ماشین پذیرفته می شود، را نمی توان توسط آن بدست آورد.

گزینه 2 نادرست است، چون a که توسط ماشین پذیرفته می شود، توسط این ماشین پذیرفته نمی شود.

گزینه 4 نادرست است، چون a که توسط ماشین پذیرفته می شود، توسط این گرامر قابل تولید نمی باشد.

۲-۳) زبان L_1 رشته 10 را تولید می کند که توسط ماشین پذیرفته نمی شود. زبان L_4 رشته 110 را تولید می کند که توسط ماشین پذیرفته نمی شود.

۳-۱) حالتهای 2و4 معادلند و اگر به جای آنها از یک حالت استفاده شود، آنگاه مشخص است که حالتهای 1و5 نیز معادل می باشند. بنابراین در نهایت 3 حالت خواهیم داشت.

۴-۱) گزینه یک رشته 01 را نمی پذیرد.

فصل چهارم: زبان ها و گرامرهای منظم

بر اساس تعریفی که در فصل قبل ارائه شد، یک زبان در صورتی منظم است کـه پذیرنـده متنـاهی بـرای آن وجـود داشـته باشـد. بنابراین هر زبان منظمی را می توان بوسیله یک dfa یا nfa تعریف کرد. در این فصل به برخی دیگر از روشهای توصیف زبان های منظم می پردازیم.

عبارات منظم

یک روش برای توصیف زبانهای منظم، استفاده از مجموعه سمبل های عبارات منظم است. این مجموعه سمبل ها شامل ترکیبـی از سمبل ها، از قبیل الفبای Σ ، پرانتزها و عملگرهای + ، . و* می باشند.

تعریف صوری برای یک عبارت منظم

عبارات منظم با انجام متوالی برخی قوانین بازگشتی روی اجزاء پایه ای و به روشی مشابه عبارات ریاضی ایجاد می شوند.

تعریف: Σ را الفبای مفروض در نظر می گیریم. آنگاه:

1- ϕ ، l و $a \in \Sigma$ همگی عبارات منظم هستند. این عبارات را عبارات منظم پایه می خوانیم.

2- اگر r_1 و r_2 عبارات منظمی باشند، آنگاه $r_1 + r_2$ ، $r_1.r_2$ ، r_1^* و (r_1) نیز عبارات منظم خواهند بود.

3- یک رشته فقط و فقط در صورتی عبارت منظم است که با بکارگیری تعداد محدودی از قوانین بنـد (2)، از عبـارات مـنظم پایـه بدست آیند.

مثال : آیا رشته $(a+b+c)^*(c+f)$، یک عبارت منظم است؟ $(\Sigma = \{a,b,c\})$

حل: بله- چون در ساخت آن از قوانین فوق استفاده شده است.

زبان های مرتبط با عبارات منظم

از عبارات منظم می توان برای توصیف برخی زبانهای ساده استفاده نمود. اگر r یک عبارت منظم باشد، زبان مرتبط با آنرا بـا $L(r)$ نمایش می دهیم.

تعریف: زبان $L(r)$ مربوط به عبارت منظم r ، با قوانین زیر تعریف می شود:(r_1 و r_2 عبارت منظم می باشند)

1- ϕ عبارت منظم است که به مجموعه تهی دلالت می کند.

2- l یک عبارت منظم مربوط به $\{l\}$ است.

3- به ازای هر $a \in \Sigma$ ، a عبارت منظم مربوط به $\{a\}$ است.

4- $L(r_1 + r_2) = L(r_1) \cup L(r_2)$ 5- $L(r_1.r_2) = L(r_1).L(r_2)$ 6- $L((r_1)) = L(r_1)$ 7- $L(r_1^*) = (L(r_1))^*$

معادلات مربوط به عبارات منظم

1	$\phi^* = \{\lambda\}$	13	$(a+b)^* = (a^*+b)^*$
2	$\phi^+ = \phi$	14	$(a+b)^* = (a+b^*)^*$
3	$l^* = l$	15	$(a+b)^* = (a^*+b^*)^*$
4	$\lambda^*.\phi^* = \lambda$	16	$(a+b)^* = (a^*b^*)^*$
5	$l^*f^* = \{l\}$	17	$(a+b)^* = a^*(a+b)^*$
6	$\lambda - \phi^* = \phi$	18	$(a+b)^* = a^*(a+b)^*b^*$
7	$\lambda^* - \phi^* = \phi$	19	$(a+b)^* = a^*(ba^*)^*$
8	$(a^*)^+ = a^*$	20	$(a+b)^* = b^*(ab^*)^*$
9	$(a^+)^* = a^*$	21	$(a+b)^* = (a^*b+ab^*)^*$
10	$(a^*)^* = a^*$	22	$(a+b)^R = a^R + b^R$
11	$a^*a^* = a^*$	23	$(ab)^R = b^R a^R$
12	$(ab)^*a = a(ba)^*$	24	$(a^*)^R = (a^R)^*$

تذکر: به جای + از $\mathbf{U}$ نیز استفاده می شود. مثلا عبارت $(a+b)$ معادل $(a \mathbf{U} b)$ می باشد.

اگر a ,b عبارتهای منظم باشد، روابط زیر همواره برقرارند:

$$(a+b)^*a(a+b)^* = b^*a(a+b)^* = (a+b)^*ab^*$$

$$(\Sigma-\{a\})^*a\Sigma^* = \Sigma^*a(\Sigma-\{a\})^*$$

مثال : با فرض $\Sigma=\{a,b\}$ داریم:

$$(ab)^+ = \{ab, abab, ababab, ...\}$$

$$(a+b)^+ = \{a, b, ab, ba, aa, bb, aab, baa, ...\}$$

مثال :نمایش زبان $L(a^*.(a+b))$ برحسب مجموعه رشته ها:

$$L(a^*.(a+b)) = L(a^*).L(a+b) = L(a^*).(L(a)\mathbf{U}L(b)) = \{\lambda, a, aa, aaa, ...\}\{a,b\} = \{a, b, aa, ab, aaa, aab, ...\}$$

مثال: معادل عبارت $((a^*b^*)^*(b^*a^*)^*)^*$ را بدست آورید.

حل: می‌دانیم که $(a^*b^*)^* = (a\mathbf{U}b)^*$ ، بنابراین داریم:

$$((a\mathbf{U}b)^*(b\mathbf{U}a)^*)^* = ((a\mathbf{U}b)^*)^* = (a\mathbf{U}b)^*$$

مثال : معکوس عبارت منظم $ab(c+d^*(efg))$ را مشخص کنید.

$$[ab(c+d^*(efg))]^R \Rightarrow [c+d^*(efg)]^R.(ab)^R = (c^R + [d^*(efg)]^R)(ba) = (c+(efg)^R(d^*)^R)(ba) =$$
$$(c+(gfe)(d^R)^*)(ba) = (c+(gfe)d^*)(ba) = ((gfe)d^*+c)(ba)$$

تذکر: در واقع برای محاسبه معکوس یک عبارت منظم می توان عبارت را از راست به چپ نوشت.

مثال : برای $\Sigma=\{a,b\}$ ، عبارت منظم $r=(a+b)^*(a+bb)$ به چه زبانی اشاره دارد؟

حل: بخش $(a+b)^*$ به معنای هر رشته ای از aها و bها است. بخش $(a+bb)$، بیانگر یک a یا دو b می باشد. در نتیجه ، L(r) مجموعه تمام رشته های روی {a,b} است که به یک a یا یک bb ختم می شوند.

$$L(r) = \{a, bb, aa, abb, ba, bbb, ...\}$$

مثال : عبارت $r=(aa)^*(bb)^*b$ به چه زبانی اشاره دارد؟

حل: به مجموعه تمام رشته هایی با تعداد زوجی از a ها که قبل از تعداد فردی از b ها می آیند:

$L=\{a^{2n}b^{2m+1}: n\geq 0, m\geq 0\}$

مثال : عبارت منظم aa*(bb)*bb ، زبان $L=\{a^nb^{2k}: n\geq 1, k\geq 1\}$ را توصیف می کند. ■

برای هر زبان تعداد نامحدودی عبارت منظم وجود دارد.

مثال : برای $\Sigma=\{0,1\}$ ، عبارت منظم r را به صورتی ارائه دهید که:

$L(r)=\{w\in\Sigma^*:$ دارای حداقل دو صفر متوالی است w $\}$

حل: در یک مکان از هر رشته این زبان، دو صفر متوالی وجود دارد و قبل و بعد 00 ، اختیاری است و هر رشته دلخواه روی الفبای $\Sigma=\{0,1\}$ که به صورت $(0+1)^*$ تعریف می شود، می تواند قرار گیرد:

$r=(0+1)*00(0+1)*$

البته می توان عبارت منظم $r=((0+1)(0+1)^*)^*00(0+1)^*$ را نیز ارائه داد. ■

مثال : برای $\Sigma=\{0,1\}$ ، عبارت منظم r را به صورتی ارائه دهید که:

$L(r)=\{w\in\Sigma^*:$ دارای هیچ دو صفر متوالی نمی باشد w $\}$

حل:این زبان مکمل زبان مثال قبل است. در واقع زبان L ، تکراری از رشته های 1 و 01 می باشد:

$r=(1+01)^*(0+l)$

البته می توان عبارتهای منظم زیر را نیز ارائه داد:

$r=(1^*011^*)^*(0+l)+1^*(0+l)$

$r=(1+01)^*(0+1^*)$

مثال : عبارت منظمی برای زبان $L=\{a^nb^m: n\geq 4, m\leq 3\}$ ارائه دهید.

حل: مساله را به حالتهای m=0،1،2،3 تقسیم می کنیم. تعداد 4 یا بیشتر a تولید کرده و پس از آن بـه تعـداد لازم b قـرار مـی دهیم:

$r=aaaaa^*(l+b+bb+bbb)$

مثال : عبارت منظمی برای مکمل زبان $L=\{a^nb^m: n\geq 4, m\leq 3\}$ ارائه دهید.

حل: در $\overline{L}$ رشته هایی به فرم a^nb^m و با شرایط $n<4$ یا $m>3$ وجود دارند. همچنین پس از یک a ، یک b قرار می گیرد:

$(l+a+aa+aaa)b^*+a^*bbbbb^*+(a+b)^*ba(a+b)^*$

مثال : عبارت منظمی برای $L=\{a^nb^m: n\geq 1, m\geq 1, nm\geq 3\}$ بنویسید.

حل: اینکار را طی سه مرحله انجام دهید: 1- $m=1, n\geq 3$ 2- $n=1, m\geq 3$ 3- $n\geq 2, m\geq 2$

راه حل هر کدام به راحتی بدست می آید. ■

مثال : عبارت منظمی برای $L=\{vwv: v,w\in \{a,b\}^*, |v|=2\}$ بنویسید.

حل: تمام حالت های با ضابطه $|v|=2$ را شمارش می کنیم:

$$aa(a+b)^*aa+ab(a+b)^*ab+ba(a+b)^*ba+bb(a+b)^*bb$$

مثال : یک عبارت منظم برای تمام رشته هایی که از هر یک از سمبل هـای $\Sigma=\{a,b,c\}$ در آنهـا حـداقل یـک سـمبل وجـود داشته باشند، بنویسید.

حل:کافی است هر یک از سمبل ها را یک مرتبه بنویسیم: $(a+b+c)^*a(a+b+c)^*b(a+b+c)^*c(a+b+c)^*$

اما در این جمله a قبل از b و b قبل از c قرار می گیرد. برای بدست آوردن جواب نهایی، تمام جایگشتهای سـه سـمبلی را ایجـاد کرده و شش عبارت حاصله را با هم جمع می کنیم. ■

مثال : عبارت منظمی برای "تمام رشته های حاوی تعداد زوجی 0 " روی $\{0,1\}$ بنویسید.

حل: این عبارت منظم از دو جمله تشکیل شده است:

1- دو 0 که چند 1 در میان آنها قرار گرفته است. 2- رشته فاقد صفر(صفر عدد زوج است)

$$(1^*01^*01^*)^*+1^*$$

مثال : عبارت منظمی برای $L=\{w: |w| \mod 3=0\}$ روی $\{0,1\}$ بنویسید.

حل: در واقع می خواهیم رشته های به طول 0 و 3 و 6 و ... را تولید کنیم. بنابراین تمام رشته های ممکن با طـول سـه را ایجـاد و تکرار می کنیم:

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

مثال : یک عبارت منظم برای نمایش مجموعه $\{<8^n+1>_2 \mid n\geq 1\}$ مشخص کنیـد؟ ($<a>_2$ نمـایش عـدد a در مبنـای 2 است)

حل: به ازای چند n ، حاصل را بدست آوریم:

$$n=1\rightarrow 8^n+1=9\rightarrow 1001$$
$$n=2\rightarrow 8^n+1=65\rightarrow 1000001$$
$$n=3\rightarrow 8^n+1=513\rightarrow 1000000001$$

با نگاه به سه جمله تولید شده، متوجه می شویم که تمامی رشته ها با 100 شروع و بعد $(000)^*$ و در نهایت به 1 ختم می شـوند. بنابراین عبارت منظم آن برابر است با: $100(000)^*1$

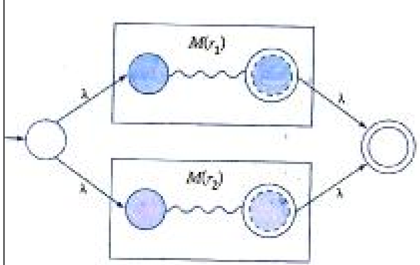
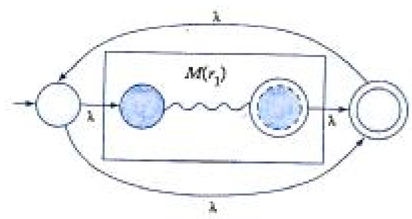
ارتباط بین عبارات منظم و زبانهای منظم

زبانهای منظم و عبارات منظم ارتباط نزدیکی با هم دارند. در واقع این دو مفهوم اساسا یکـی هسـتند، بطوریکـه بـه ازای هـر زبـان منظم، یک عبارت منظم و به ازای هر عبارت منظم، یک زبان منظم وجود دارد. اگر r یک عبارت منظم باشد،آنگاه L(r) یک زبـان منظم است.

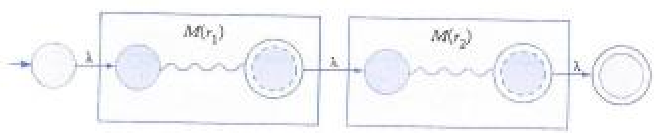
قضیه: فرض کنیم r یک عبارت منظم باشد. آنگاه یک nfa وجود دارد که L(r) را می پذیرد. در نتیجه L(r) یک زبان منظم است.

قانون ایجاد NFA یک عبارت منظم

با فرض اینکه ماشین $M(r_1)$ و $M(r_2)$ به ترتیب زبانهای تعریفی عبارتهای منظم r_1 و r_2 را بپذیرند، ماشین های مربوط به $L(r_1+r_2)$ و $L(r_1r_2)$ و $L(r_1^*)$ در زیر رسم شده است:

$L(r_1+r_2)$	$L(r_1^*)$
	

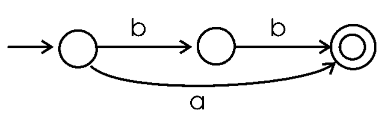
ماشین های مربوط به $L(r_1r_2)$:



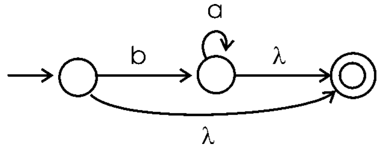
مثال :nfa ای را پیدا کنید که L(r) را بپذیرد. بطوریکه:

$r=(a+bb)^*(ba^*+l)$

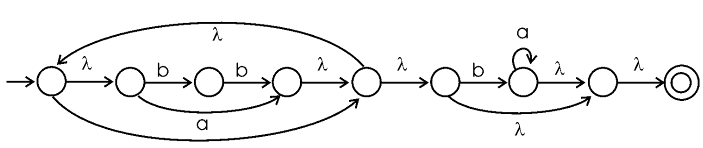
حل: ماشین مربوط به $(a+bb)^*$ به صورت زیر می باشد:



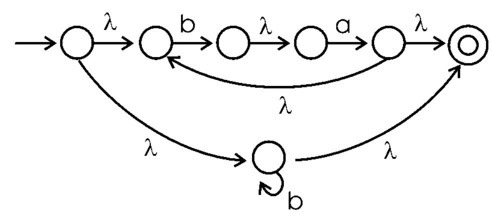
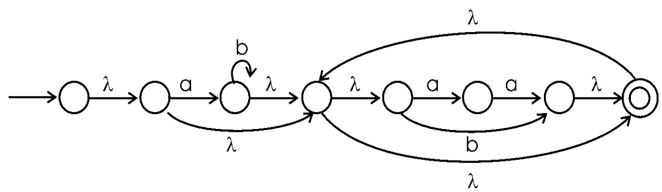
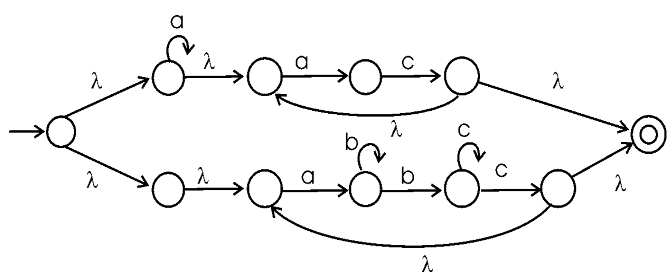
ماشین مربوط به (ba^*+l) به صورت زیر می باشد:



از کنار هم قرار دادن این ماشین ها، به ماشین نهایی می رسیم:



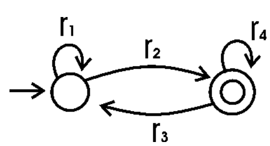
مثال : در شکل زیر چند NFA که از روی عبارت منظم مقابل آن ساخته شده، آورده شده است:

$b^* + (ba)^*$	
$r = (\lambda + ab^*)(aa + b)^*$	
$r = a^*(ac)^+ + (ab^+c^+)^*$	

عبارات منظم برای زبانهای منظم

برای هر زبان منظم، یک عبارت منظم متناظر وجود دارد. برای راحتی کار از گراف های انتقال تعمیم یافته(GTG) استفاده می کنیم. این گراف مانند گراف انتقال است با این تفاوت که یالهای آن با عبارت منظم برچسب دار شده است.

مثال :عبارت منظم متناظر با گراف انتقال تعمیم یافته زیر را بدست آورید.

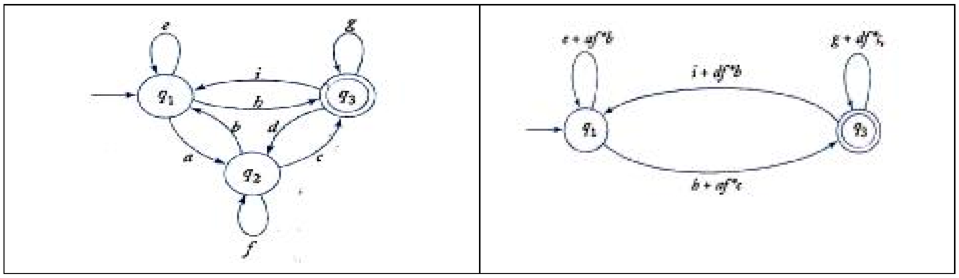


حل: عبارت منظم متناظر با گراف داده شده به شکل $r = r_1^* r_2 (r_4 + r_3 r_1^* r_2)^*$ می باشد.توجه کنید که این عبارت منظم، تمام مسیرهای ممکن از حالت شروع به حالت پایانی را پوشش داده است. ■

به ازای هر زبان منظم، یک GTG وجود دارد که آن را می پذیرد. بالعکس، هر زبانی که بوسیله یک GTG پذیرفته می شود، منظم است.

GTG کامل، گرافی است که تمام یال ها در آن حضور داشته باشند. اگر در یک GTG، پس از تبدیل از nfa ، برخی یال ها حضور نداشته باشند، آنها را با f برچسب دار می کنیم.

مثال : اگر یک GTG بیش از دو حالت داشته باشد، می توان گراف متناظر با آن که دارای دو حالت است را پیدا کرد. GTG متناظر با GTG کامل داده شده در زیر نشان داده شده است:



مثال :یک عبارت منظم برای زبان L پیدا کنید.

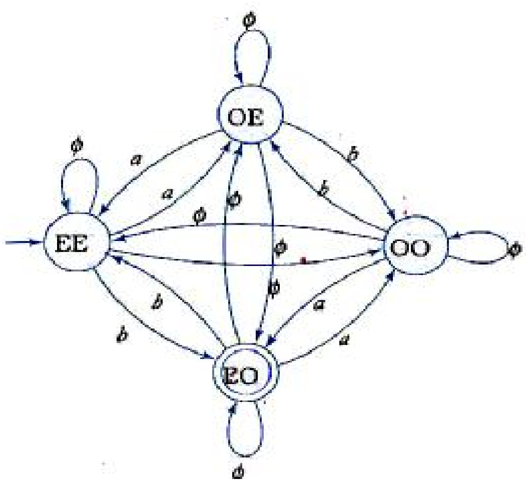
$L = \{w \in \{a,b\}^* :$ زوج است $n_a(w)$ و فرد است $n_b(w)$ $\}$

حل: برای پیدا کردن عبارت منظم، ابتدا یک nfa برای آن رسم کرده و سپس آن را به GTG تبدیل می کنیم. برچسب گذاری nfa ، را به صورت زیر انجام می دهیم:

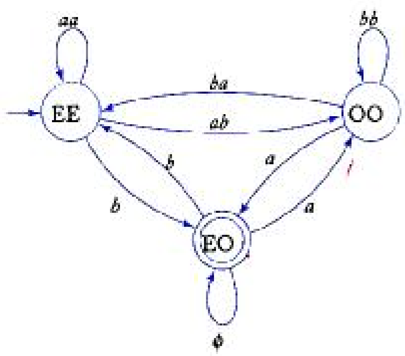
OE : نمایش فرد بودن تعداد aها و زوج بودن تعداد bها

EO : نمایش زوج بودن تعداد aها و فرد بودن تعداد bها

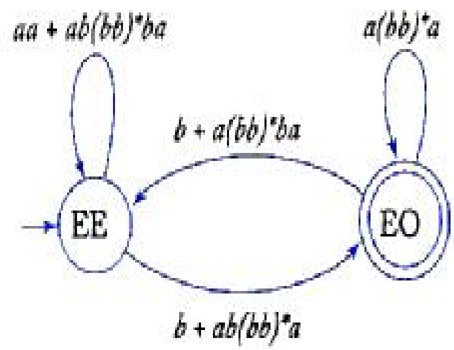
EE : نمایش زوج بودن aها و bها وOO: نمایش فرد بودن aها و bها



حذف حالت OE :



حذف حالت OO :



قضیه: L را یک زبان منظم فرض می کنیم. بنابراین، عبارت منظمی به ازای r وجود خواهد داشت، بطوریکه L=L(r) .

گرامرهای منظم

یکی از روشهای شرح زبانهای منظم، استفاده از برخی گرامرهای ساده است. گرامر منظم، گرامری است که خطی از راست یا خطی از چپ باشد. هر گرامر منظم، یک زبان منظم تولید می‌کند.

گرامر خطی از راست : گرامری که همه قواعد آن به صورت $A \to xB \mid x$ باشد.

گرامر خطی از چپ : گرامری که همه قواعد آن به صورت $A \to Bx \mid x$ باشد. **که** $A, B \in V$ **و** $x \in T^*$ **می باشد.**

تذکر: در تمامی گرامرهای منظم، حداکثر یک متغیر در سمت راست هر یک از قوانین قرار می گیرد. بعلاوه، این متغیر باید همواره آخرین سمبل از سمت راست **یا** از سمت چپ، طرف راست هر یک از قوانین باشد.

مثال : آیا گرامر زیر منظم است؟

$S \to abS$

$S \to a$

حل: بله- چون همه قوانین آن خطی از راست می باشند. زبان این گرامر، زبان منظم $(ab)^*a$ است.■

مثال : آیا گرامر زیر منظم است؟

$S \to Aab$

$A \to Aab \mid B$

$B \to a$

حل: بله- چون همه قوانین آن خطی از چپ می باشند. زبان این گرامر، زبان منظم $aab(ab)^*$ است.■

تعریف: گرامر خطی، گرامری است که در آن، حداکثر یک متغیر می تواند در سمت راست هر قانون وجود داشته و بعلاوه، هیچ محدودیتی در مورد محل این متغیر وجود ندارد.

گرامرهای منظم، همواره خطی هستند، اما لزوما همه گرامرهای خطی منظم نیستند.

مثال :آیا گرامر زیر منظم است؟

$S \to A$

$A \to aB \mid l$

$B \to Ab$

حل: این گرامر منظم نمی باشد، چون بعضی از قوانین آن خطی از راست و بعضی خطی از چپ هستند. گرامر داده شده، خطی است.

برای هر زبان منظم فاقد l ، یک گرامر خطی از راست وجود دارد که قوانین آن فقط به فرم $A \to aB$ یا $A \to a$ هستند.
که در آن $A, B \in V, a \in T$.

مثال : گرامر خطی از راست برای زبان $L = \{a^n b^m : n \geq 2, m \geq 3\}$:

$$S \to aaA$$
$$A \to aA \mid B$$
$$B \to bbbC$$
$$C \to bC \mid l$$

مثال : گرامر خطی از چپ برای زبان $L = \{a^n b^m : n \geq 2, m \geq 3\}$:

$$S \to Abbb$$
$$A \to Ab \mid B$$
$$B \to aaC$$
$$C \to aC \mid l$$

مثال :گرامر منظمی بنویسید که زبان $L = \{a^n b^m :$ n+m زوج است $\}$ را تولید کند.

حل: مساله را به دو قسمت تقسیم می کنیم:

الف- n و m هر دو زوج باشند.

ب- n و m هر دو فرد باشند.

با توجه به قسمت اول جواب زیر بدست می آید:

$$S \to aaS \mid A$$
$$A \to bbA \mid l$$

مثال :گرامر منظمی برای زبان $L = \{w :$ $n_b(w), n_a(w)$ هر دو زوج هستند $\}$ روی {a,b} بنویسید.

حل: ابتدا یک dfa برای L ساخته که انتقالات زیر بدست می آید:

$$d(q_0, a) = q_1$$
$$d(q_1, a) = q_0$$
$$d(q_2, a) = q_3$$
$$d(q_3, a) = q_2$$
$$d(q_0, b) = q_2$$
$$d(q_1, b) = q_3$$
$$d(q_2, b) = q_0$$
$$d(q_3, b) = q_1$$

که در آن q_0 حالت شروع و پایانی می باشد.

سپس از روی انتقالات بدست آمده ، گرامر زیر می نویسیم:

$q_0 \to aq_1 \mid bq_2 \mid l$
$q_1 \to bq_3 \mid aq_0$
$q_2 \to aq_3 \mid bq_0$
$q_3 \to aq_2 \mid bq_1$

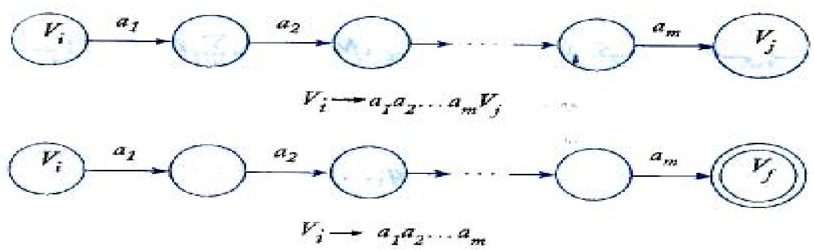
در جدول زیر عبارتهای منظم تولید شده توسط هر گرامر در مقابل آن آورده شده است:

عبارت منظم	گرامر
a^*	$S \to aS$ $S \to \lambda$
a^+b^+	$S \to aS \mid aB$ $B \to bB \mid b$
a^*b^*	$S \to aS \mid bA \mid \lambda \mid aA$ $A \to bA \mid \lambda \mid b$
$(ab)^*$	$S \to aA \mid \lambda$ $A \to bS$
$a(a+b)^*$	$S \to aA \mid a$ $A \to aA \mid bA \mid a \mid b$
$a^+ + bc^*$	$S \to aA \mid bC$ $A \to aA \mid \lambda$ $C \to cC \mid \lambda$
b^*ab^*	$S \to bS \mid aA$ $A \to bA \mid \lambda$
$(ba+a+c)^*$	$S \to baS \mid as \mid cs \mid \lambda$
$(ab+ac+b+c)^*(a+l)$	$S \to abS \mid acS \mid bS \mid cS \mid a \mid \lambda$
$(a+b+c)^3$	$S \to aA \mid bA \mid cA$ $A \to aB \mid bB \mid CB$ $B \to a \mid b \mid c$
$(a+b+c+l)^2$	$S \to aA \mid bA \mid cA \mid \lambda$ $A \to a \mid b \mid c \mid \lambda$
$b^*(ab^*ab^*ab^*)^*$	$S \to bS \mid aA \mid \lambda$ $A \to bA \mid aB$ $B \to bB \mid aS$

قضیه: فرض کنیم $G=(V,T,S,P)$ یک گرامر خطی از راست باشد، آنگاه $L(G)$ یک زبان منظم خواهد بود.

در اثبات این قضیه از نکته زیر استفاده شده است:

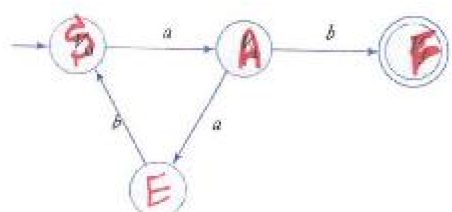
شمای کلی به ازای قانون $V_i \to a_1a_2...a_mV_j$ و $V_i \to a_1a_2...a_m$ در زیر آورده شده است:



مثال : ماشین متناهی بسازید که زبان تولید شده بوسیله گرامر زیر را بپذیرد.

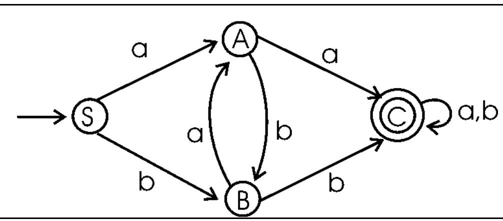
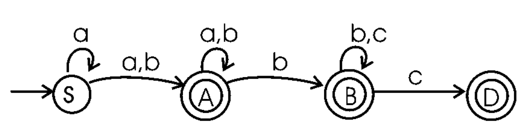
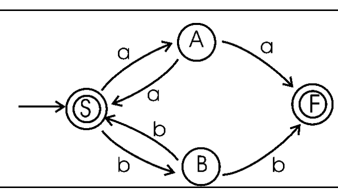
$$S \to aA$$
$$A \to abS \mid b$$

حل: گراف انتقالی با سه راس S و A و F ایجاد می کنیم. یالی با برچسب a بین S و A ایجاد مـی نمـاییم. سـپس راس E را بـه گونه ای ایجاد می کنیم که مسیری با برچسب ab بین A و S وجود داشته باشد. در نهایت یالی با برچسب b بـین A و F ایجـاد می کنیم تا ماشین شکل زیر بدست آید.



زبان تولید و پذیرفته شده توسط این گرامر، زبان منظم $L((aab)^*ab)$ خواهد بود. ■

مثال : در زیر ماشین متناهی متناظر با هر گرامر در مقابل آن نشان داده شده است:

گرامر	ماشین متناهی
$S \to aA \mid bB$ $A \to aC \mid bB \mid a$ $C \to aC \mid bC \mid a \mid b$ $B \to aA \mid bC \mid b$	
$S \to aS \mid aA \mid bA$ $A \to aA \mid bA \mid bB \mid \lambda$ $B \to bB \mid cB \mid cD \mid \lambda$ $D \to \lambda$	
$S \to aA \mid bB \mid \lambda$ $A \to aS \mid a$ $B \to bS \mid b$	

هم ارزی زبان های منظم و گرامرهای منظم

دو قضیه قبل، ارتباط بین زبان های منظم و گرامرهای خطی از راست را نشان می دهد. همچنین می توان بین زبان هـای مـنظم و گرامرهای خطی از چپ نیز ارتباط مشابهی را برقرار کرده و به این ترتیب، هم ارزی کامل گرامرهای مـنظم و زبـان هـای مـنظم را اثبات نمود. با کنار هم قرار دادن دو قضیه زیر، به هم ارزی زبان های منظم و گرامرهای منظم می رسیم.

قضیه زبان L منظم است اگر و تنها اگر گرامر خطی از چپ G وجود داشته باشد بطوریکه L=L(G) .
قضیه: زبان L منظم است اگر و تنها اگر گرامر منظم G وجود داشته باشد بطوریکه L=L(G) .

در این فصل با روشهای مختلف توصیف زبانهای منظم، یعنی استفاده از dfa ها و nfa ها ، عبارات منظم و گرامرهـای مـنظم آشـنا شدید. ارتباط بین این مفاهیم در قالب چهار قضیه آمده است:

1- فرض کنیم r یک عبارت منظم باشد. آنگاه یک nfa وجود دارد که L(r) را می پذیرد. در نتیجه L(r) یک زبان منظم است.

2- L را یک زبان منظم فرض می کنیم. بنابراین، عبارت منظمی به ازای r وجود خواهد داشت، بطوریکه L=L(r) .

3- فرض کنیم $G=(V,T,S,P)$ یک گرامر خطی از راست باشد. آنگاه L(G) یک زبان منظم خواهد بود.

4- اگر L یک زبان منظم روی الفبای Σ باشد، آنگاه گرامر خطی از راست $G=(V,\Sigma,S,P)$ به صورتی وجود خواهد داشت که L=L(G) .

مجموعه تست

۱- عبارت منظم R و گرامرهای G_1 ,G_2 و G_3 با تعریف زیر مفروضند. اگر زبان R را L بنامیم، L_1 , L_2 و L_3 به ترتیب زبان گرامرهای مذکور می باشند، کدام گزاره صحیح است؟

$R = ((aa \mid b)^* b)^* a$

$G_1: S \to bS \mid aA \mid aC$, $A \to aS$, $C \to \varepsilon$

$G_2: S \to bS \mid aA \mid aC$, $A \to Sa$, $C \to \varepsilon$

$G_3: S \to bS \mid aA \mid aC$, $A \to aS$, $C \to \varepsilon$

1) $L = L_1 = L_2 = L_3$
2) $L_1 \neq L_3, L = L_1$
3) $L_2 \neq L, L = L_1 = L_3$
4) $L_3 \neq L_2, L = L_1 = L_2$

۲- زبان های زیر با $\alpha, \gamma \in \sum^*$, $\beta \in \sum^+$ مفروضند. کدام گزینه صحیح است؟

$L_1 = \{\alpha^i (\alpha\beta)^j (\gamma_\alpha)^i \quad j \geq 0 \ , \ i \geq 0\}$

$L_2 = \{\alpha^i (\alpha\beta)^j (\gamma_\alpha)^{i+j} \quad j \geq 0 \ , \ i \geq 1\}$

$L_3 = \{\alpha^i (\alpha\beta)^j (\gamma_\alpha)^i \quad j \geq 1 \ , \ i \geq 1\}$

1) L_1 و L_3 هر دو منظم هستند.
2) L_1 منظم و L_3 نامنظم است.
3) L_1 منظم و L_2 نامنظم است.
4) L_1 و L_2 و L_3 همگی نامنظم هستند.

۳- کدام یک از زبان های زیر نامنظم است؟

1) $\{a^n b^n (a+b)^* \mid n \geq 0\}$
2) $\{b^* a^n b^n a^* \mid n \geq 0\}$
3) $\{a^* a^n b^n b^* \mid n \geq 0\}$
4) هر سه نامنظم هستند.

۴- کدام عبارت منظم، مجموعه $\{<8^n + 1>_2 \mid n \geq 0\}$ که در آن $<t>_2$ نمایش عدد t در مبنای ۲ است، را مشخص می کند؟

1) $1(000)^*1$
2) $(1000)^*+1$
3) $1(01)^*+10$
4) $100(000)^*1+10$

۵- عبارت منظم $0(0+10)^*11$ با کدام عبارت داده شده معادل است؟

1) $(00^*1)^+1$
2) $(00^+1)^*1$
3) $(0^*10)^*1$
4) $(0^*11)^+1$

۶- کدام عبارت منظم، زبان زیر را توصیف می کند؟

$L = \{a^n b^{3m} c^{2k} \mid m \geq 1, n \geq 1, k \geq 1\}$

1) a*(bbb)*(cc)*
2) a*b*b*b*c*c*
3) aa*(bbb)*bbb(cc)*cc
4) aa*bbb*b*b*ccc*c*

۷- عبارت منظم معادل گرامر زیر کدام است؟

$$S \to T \mid D$$
$$T \to aT \mid bD$$
$$D \to bD \mid aT \mid l$$

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

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

3) $(a+b)^*$

4) $(a^*b + l)b^*(aa^*bb^*)^*$

۸- کدام یک از گزینه های زیر نادرست است؟ $(\Sigma = \{0,1\})$

1) تمام رشته هایی که شامل 101 نباشد: $0*(1+1000*)*(\lambda+10)$

2) تمام رشته هایی که شامل تعداد زوج صفر باشند: $(1+01*0)*$

3) تمام رشته هایی که به 01 ختم نمی شوند : $(0+1)*(0+11)+1+\lambda$

4) تمام رشته هایی که شامل 100 نباشد: $0*(1+1000*)*(\lambda+10)$

۹- چنانچه α و β عبارت های منظم بر روی الفبای Σ باشند، کدام یک از روابط زیر نادرست است؟

1) $(\alpha\beta)^R = \alpha^R\beta^R$

2) $(\alpha \mathbf{U} \beta)^* = (\alpha \mathbf{U} \beta^*)^* = \beta^*(\alpha\beta^*)^*$

3) $(\alpha^*)^R = (\alpha^R)^*$

4) $(\Sigma - \{a\})^* a \Sigma^* = \Sigma^* a(\Sigma - \{a\})^*$

۱۰- با فرض آن که α و β عبارات منظم هستند، کدام یک از عبارات زیر نادرست است؟

1) $a(ba)^* = (ab)^*a$

2) $(ab+a)^*a = a(ba+a)^*$

3) $(a+b)^* = (a^*+b)^*$

4) هیچکدام

۱۱- عبارت منظم برای زبان زیر بیان نمائید.

$L = \{w \in \{0,1\}^*$: دارای یک زوج صفر متوالی نباشد $W\}$

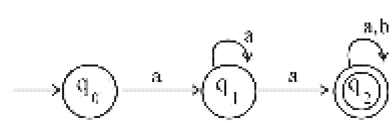
1) $r = (1+10)^*(10+\lambda)$

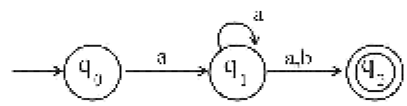
2) $r = (1^*011^*)^*(0+\lambda) + 1^*(0+\lambda)$

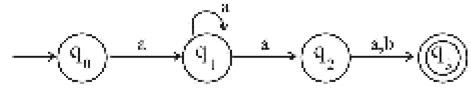
3) $r = (1+01)^*(0+\lambda)$

4) گزینه های 2و3

۱۲- یک NFA بیابید که زبان $L(aa^*(a+b))$ را بپذیرید.

1)


2)


3)


4) هیچکدام

پاسخ تشریحی

۱-؟) زبان L رشته aaa را تولید نمی کند در حالی که زبانهای L_1 , L_2 و L_3 این رشته را تولید می کنند. بنابراین L با هیچ یک از این سه زبان برابر نمی باشد و هیچکدام از گزینه ها درست نیست.

۲-۱) a و b و g می توانند هر مقداری باشند و وابستگی آنها در زبانهای L_1 و L_3 از بین رفته است.

۳-۲) زبانهای گزینه های 1و3 منظم می باشند.

در گزینه یک آمدن عبارت $(a+b)^*$ در کنار $a^n b^n$، وابستگی بین a وb را از بین برده است.

در گزینه سه نیز وابستگی بین a وb از بین رفته است. چون آمدن a^* قبل از a^n تعداد a را نامشخص کرده است.

۴-۴) به ازای چند n ، حاصل را بدست آوریم:

$$n=0 \rightarrow 8^n+1=2 \rightarrow 10$$
$$n=1 \rightarrow 8^n+1=9 \rightarrow 1001$$
$$n=2 \rightarrow 8^n+1=65 \rightarrow 1000001=100(000)1$$
$$n=3 \rightarrow 8^n+1=513 \rightarrow 1000000001=100(000)^2 1$$

متوجه می شویم که عبارت منظم آن $10+1*(000)100$ می باشد. (به ازای n=0 عبارت 10 و به ازای n>0 عبارت $100(000)^*1$)

تذکر: به ازای n=1 حاصل 8^n+1 برابر 9 است که در مبنای دو به صورت (1001) نمایش داده می شود و تنها توسط گزینه 4 قابل تولید است.

۵-۱) به عنوان مثال رشته 011 فقط توسط گزینه یک قابل تولید است. (رشته 1 توسط گزینه 2و3 قابل تولید است بنابراین نادرست می باشند. رشته 111 توسط گزینه 4 قابل تولید است، بنابراین نادرست است.)

۶-۳) گزینه 1و2 رشته l را تولید می کنند که توسط زبان L قابل تولید نمی باشد.

۷-۴) رشته aba توسط گرامر تولید نمی شود در حالی که عبارت منظم گزینه 1 و3 آن را تولید می کنند. گزینه 2 نیز نادرست است، چون رشته باید حداقل یک b داشته باشد که در گرامر چنین محدودیتی وجود ندارد.

۸-۴)

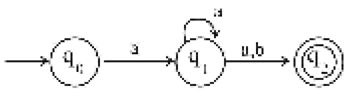
۹-۱) گزینه یک نادرست است و صحیح آن به صورت مقابل می باشد: $(ab)^R = b^R a^R$

۱۰-۴) تمامی عبارات داده شده درست می باشند.

۱۱-۴) هر دو عبارت منظم زیر، زبان $L=\{w \in \{0,1\}^* :$ W دارای یک زوج صفر متوالی نباشد$\}$ را بیان می کنند:

$$r=(1^*011^*)^*(0+\lambda)+1^*(0+\lambda)$$
$$r=(1+01)^*(0+\lambda)$$

۱۲-۲) ماشین NFA پذیرنده زبان $(aa^*(a+b))$ به صورت زیر می باشد:



فصل پنجم: **ویژگی های زبان های منظم**

در این فصل نگاهی به انواع ویژگی های زبان های منظم خواهیم داشت. این ویژگی ها اطلاعات خوبی در مورد نقاط قوت و ضعف زبان های منظم به ما می دهد. همچنین به پرسشهای زیر پاسخ می دهیم:

1- عملگرهای مختلف چه تاثیری روی زبانهای منظم می گذارند؟

2- توانایی ما در تصمیم گیری راجع به ویژگی هایی از قبیل متناهی یا نامتناهی بودن زبان منظم چقدر است؟

3- چه معیاری برای تشخیص منظم بودن یا منظم نبودن یک زبان خاص وجود دارد؟

ویژگی های بستاری زبان های منظم

مسئله بسته بودن(بستار)، یعنی آیا زبان حاصل در اثر اعمال عملگرها، باز هم منظم خواهد بود؟

ابتدا بسته بودن تحت عملگرهای ساده روی مجموعه ها مانند اشتراک و اجتماع را بررسی خواهیم کرد.

قضیه:اگر L_1 و L_2 زبان های منظمی باشند، آنگاه زبان های $L_1 \cup L_2$، $L_1 \cap L_2$، L_1L_2، $\overline{L_1}$ و $\overline{L_1^*}$ منظم خواهند بود. یعنی خانواده زبان های منظم تحت اجتماع، اشتراک، الحاق، مکمل گیری و بستار ستاره ای، بسته هستند.

مثال :نشان دهید که اگر L_1 و L_2 زبان های منظمی باشند، آنگاه زبان $L_1 - L_2$ لزوما منظم خواهد بود. یعنی خانواده زبان های منظم تحت تفاضل بسته هستند.

حل: تفاضل را می توان بر اساس تعریف $L_1 - L_2 = L_1 \cap \overline{L_2}$ بر اساس اشتراک تعریف کرد. حال از آنجا که L_2 منظم است، $\overline{L_2}$ نیز منظم خواهد بود و چون زبانهای منظم تحت اشتراک بسته هستند، $L_1 \cap \overline{L_2}$ نیز منظم خواهد بود.■

قضیه: خانواده زبان های منظم تحت معکوس بسته است.

مثال :با فرض اینکه L_2, L_1 زبانهای منظم هستند، کدام یک از زبان های زیر منظم است؟

الف) $L_3 = L_1 \cap L_2^R$ ب) $L_4 = \overline{L_1} \cap \overline{L_2}$

حل: هر دو زبان منظم هستند، چون زبانهای منظم نسبت به وارون، متمم و اشتراک بسته هستند. ■

مثال : با فرض اینکه زبانهای L_2, L_1 روی الفبای $\sum$ تعریف شده باشند، آنگاه کدام گزینه درست است؟

الف) اگر L_1, $L_1 \cap L_2$ منظم باشند، آنگاه L_2 منظم است.

ب)اگر L_1, $L_1 \cup L_2$ منظم باشند، آنگاه L_2 منظم است.

ج)اگر L_1, $L_1 - L_2$ منظم باشند، آنگاه L_2 منظم است.

حل: هیچکدام از موارد الف و ب و ج صحیح نیست.

گزینه الف نادرست است، چون با فرض $L_1 = \phi$، آنگاه $L_1 \cap L_2 = \phi$ است و L_2 می‌تواند هر نوع زبانی باشد.

گزینه ب نادرست است، با فرض $L_1 = \sum$، آنگاه $L_1 \cup L_2 = \sum$ است و L_2 می‌تواند هر نوع زبانی باشد.

گزینه ج نادرست است، چون با فرض $L_1 = \phi$، آنگاه $L_1 - L_2 = \phi$ است و L_2 می‌تواند هر نوع زبانی باشد.

بسته بودن تحت سایر عملگرها

می توان به غیر از عملگرهای استاندارد روی زبان ها، عملگرهای دیگری مانند همریختی و تقسیم راست، نیز تعریف کـرد و خـواص بسته بودن روی آنها را بررسی کرد. این عملگرها مانند : همریختی وتقسیم راست.

همریختی(homomorphism)

با فرض اینکه Σ و Γ دو الفبا باشند، آنگاه تابع $h:\Sigma\rightarrow\Gamma^*$ همریختی نامیده می شود. همریختی یک نوع جایگزینی است کـه در آن به جای یک سمبل، از یک رشته استفاده می‌شود. تصویر همریختی زبان L به صورت $h(L)=\{h(w):w\in L\}$ تعریف می‌شود.

مثال : بـا فـرض $\sum=\{a,b\}$ و $\Gamma=\{a,b,c\}$ ، تـابع h را بصـورت $h(a) = ab$ و $h(b) = bbc$ تعریـف مـی کـنـیم. مطلـوب اسـت $h(aba)$.

$h(aba)=abbbcab$

قضیه : خانواده زبانهای منظم تحت همریختی بسته است.

تقسیم راست

با فرض اینکه L_2,L_1 زبان های تعریف شده بر روی یک الفبای یکسان باشند، آنگاه تقسیم راست L_1 به L_2 به صـورت زیـر تعریـف می‌شود:

$$L_1/L_2=\{s: y\in L_2 \text{ برای برخی } xy\in L_1\}$$

برای تعیین L_1/L_2، تمام رشته‌های موجود در L_1 با پسوندهای متعلق به L_2 را در نظر می گیریم. هر رشته با این فـرض، پـس از حذف مذکور، متعلق به L_1/L_2 خواهد بود.

مثال : با توجه به تعاریف L_2,L_1، مطلوب است L_1/L_2 .

$$L_1=\{a^nb^m \mid n\geq 1, m\geq 0\}\cup\{ba\} \ , L_2=\{b^m \mid m\geq 1\}$$

حل: رشته‌های موجود در L_2 از یک یا چند b تشکیل شده اند. بنابراین با حذف یک یا چند b از رشته‌های عضو L_1 که بـه حـداقل یک b ختم می‌شوند، به جواب می‌رسیم. $L_1/L_2=\{a^nb^m \mid n\geq 1, m\geq 0\}$ ■

مثال : برای زبانهای منظم زیر، L_1/L_2 را پیدا کنید.

$$L_1=L(a^*baa^*), L_2=L(ab^*)$$

حل : $L_1/L_2=L(a^*ba^*)$ ■

قضیه: خانواده زبانهای منظم تحت تقسیم راست بر یک زبان منظم، بسته است.

مثال : فرض کنید L_2,L_1 به صورت زیر تعریف شده باشند، L_1/L_2 را بیابید.

$$L_1=L(a^*baa^*) \ , L_2=L(aba^*)$$

حل :

$$L_1/L_2=\{a^*\} \ (aba\in L_2)$$

مثال : با توجه به زبانهای زیر، حاصل L_1/L_2, L_2/L_1 , L_1/L_3 , L_3/L_2 را به دست آورید.

$$L_1=\{a^*b^*c\}, L_2=\{a^*\} \ , L_3=\{b^*c\}$$

حل: $L_1/L_2=\{a*b*c\}$, $L_3/L_2=\{b*c\}$, $L_1/L_3=\{a*b*\}$, $L_2/L_1=\{\ \}$,

مثال : تقسیم راست L_1/L_2 و L_2/L_1 را تعیین کنید.

$L_1=\{0,01,111\}$, $L_2=\{0,1,11\}$

حل: $L_1/L_2=\{\lambda,0,1,11\}$ $L_2/L_1=\{\lambda\}$,

سوالات مقدماتی در رابطه با زبان های منظم

زبان های منظم فقط و فقط در صورتی قابل ارائه استاندارد هستند که بوسیله یک ماشین متناهی، عبارت منظم و یا گرامر منظم معرفی شوند.

قضیه ۱: یک الگوریتم برای تعیین وجود یا عدم وجود w در L قابل بررسی است. $(w\in\Sigma^*)$ (مسئله عضویت)

قضیه ۲: آیا الگوریتمی برای تعیین تهی بودن و متناهی یا نامتناهی بودن آن وجود دارد؟

قضیه۳ : آیا $L_1=L_2$ هست یا خیر؟ (L_2,L_1 منظم هستند.)

با فرض اینکه L و L_1 و L_2 زبان های منظم باشند که به صورت استاندارد تعریف شده اند، الگوریتم های وجود دارد که تعیین کند که:

1- آیا $w\in L_1-L_2$؟ 2- آیا $L_1\subseteq L_2$ ؟ 3- آیا $l\in L$ ؟ 4- آیا $w^R\in L$ ؟

5-آیا $L=L_1L_2$ ؟ 6- آیا $L=L^*$ ؟ 7- آیا $L(G)=\Sigma^*$ ؟ (G یک گرامر منظم است) و ....

شناسایی زبان های نامنظم

یک زبان در صورتی منظم است که در جریان پردازش رشته ها، اطلاعاتی که باید در هر مرحله به خاطر آورده شوند، کاملا محدود باشد. این گفته درست است، اما باید قبل از هر نوع استفاده ای به دقت اثبات شود. یکی از راه های اثبات نامنظم بودن یک زبان، استفاده از لم تزریق است.

لم تزریق: اگر L یک زبان منظم نامتناهی باشد، آنگاه عدد صحیح مثبت m وجود دارد بطوریکه هر $w\in L$ با شرط $|w|\geq m$، را می‌توان به صورت $w=xyz$ تجزیه کرد، با فرض $|xy|\leq m$ و $|y|\geq 1$ بطوریکه $w_i=xy^iz$ به ازای تمام $i=0,1,2,...$ عضو L باشد.

توسط لم پمپاژ(تزریق) فقط می‌توان منظم نبودن یک زبان را بررسی کرد و برای بررسی اینکه آیا یک زبان منظم هست از لم پمپاژ نمی‌توان استفاده کرد.

تذکر: لم تزریق در کتاب مرجع همین مولف به طور مفصل بررسی شده است.

مثال :کدام یک از زبان های زیر منظم هستند؟

$$L_1=\{uww^Rv:u,v,w\in\{a,b\}^+\}$$
$$L_2=\{uww^Rv:u,v,w\in\{a,b\}^+,|u|\geq|v|\}$$

حل: L_1 منظم است ولی L_2 منظم نمی باشد. یکی از عبارات منظم برای L_1، به صورت زیر است:

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

مثال :آیا زبان $L=\{w: n_a(w)\neq n_b(w)\}$ منظم است؟

حل: می دانیم که زبان های منظم تحت مکمل گیری بسته هستند. بنابراین اگر L منظم باشد، باید مکمل آن هم منظم باشد. اما چون $\overline{L}=\{w: n_a(w)=n_b(w)\}$، منظم نیست، بنابراین L نیز منظم نمی باشد.

تذکر: منظم نبودن $\overline{L}$ ، توسط لم تزریق قابل اثبات است.

مثال : کدام یک از زبان های زیر منظم هستند؟

$$L_1=\{a^n b^l a^k : n+l+k>5\}$$
$$L_2=\{a^n b^l a^k : n>5, l>3, k\le l\}$$

حل: L_1 منظم است ولی L_2 منظم نمی باشد. برای اثبات منظم بودن زبان L_1، کافی است آن را به حالت هایی مثل $l=0, k=0, n>5$ تقسیم کرد. در اینصورت می توان برای هر یک عبارات منظمی ارائه داد.

- هر زبان متناهی، منظم است.
- اگر L_1 و L_2 زبان های نامنظمی باشند، آنگاه $L_1 \cup L_2$ ممکن است منظم باشد. به طور نمونه، زبان های $L_1=\{a^n b^m : n\le m\}$ و $L_2=\{a^n b^m : n>m\}$ هر دو نامنظم هستند، امااجتماع آنها یعنی $L(a^* b^*)$، منظم است.
- اگر L_1 و L_2 زبان های منظمی باشند، آنگاه $L=\{w: w\in L_1, w^R\in L_2\}$ لزوما منظم است.

مجموعه تست

۱- کدام گزاره صحیح است؟

1) شرایط لازم و کافی برای منظم نبودن یک زبان وجود دارند ولی هنوز کشف نشده اند.

2) هیچ شرط لازم و کافی برای منظم نبودن یک زبان وجود ندارد.

3) لمpumping یک شرط لازم برای نبودن یک زبان ارائه می دهد.

4) لم pumpingیک شرط کافی برای منظم نبودن یک زبان ارائه می دهد.

۲- کدام گزینه در مورد تقسیم دو زبان L و L' با تعریف زیر درست نیست؟

$$\frac{L}{L'} = \{x \mid \exists y(xy \in L \wedge y \in L')\}$$

1) اگر $\lambda \in L'$ باشد آنگاه $L \subseteq \frac{L}{L'}$ خواهد بود.

2) اگر $\frac{L}{L'} = \phi$ باشد آنگاه $L = \phi$ و یا $L' = \phi$ است.

3) اگر $L \cap L' \neq \phi$ باشد آنگاه $\lambda \in \frac{L}{L'}$ خواهد بود.

4) اگر $L' = \phi$ باشد آنگاه $\frac{L}{L'} = \phi$ خواهد بود.

۳- اگر $M = (Q, Q_0, \Sigma, F, \delta)$ یک اتومات متناهی باشد تعریف مـی کنـیم: $\overline{M} = (Q, Q_0, \Sigma, Q - F, \delta)$. همچنـین d(M) اتومات قطعی معادل M خواهد بود. اگر M_1 و M_2 دو اتومات متناهی باشند، $M_1 + M_2$ اتومات متناهی است که زبـان آن اجتماع زبان های M_1 و M_2 است. فرض کنید G_1 و G_2 دو گرامر منظم باشند که زبان آنها به ترتیـب معـادل زبـان های M_1 و M_2 هستند. کدام عبارت زیر صحیح است؟

1) $L(G_1) - L(G_2) = L(\overline{\overline{M}_1 + M_2})$

2) $L(G_1) - L(G_2) = L(\overline{d(\overline{d(\overline{M}_1)} + M_2)})$

3) $L(G_1) - L(G_2) = L(\overline{\overline{d(\overline{M}_1)} + d(M_2)})$

4) $L(G_1) - L(G_2) = L(\overline{\overline{d(\overline{M}_1)} + \overline{d(\overline{M}_2)}})$

۴- کدام یک از دلایل زیر برای اینکه نشان دهیم زبان L منظم نیست، کافی است؟

1) عدد ثابت مثل n وجود دارد به طوری که برای هر رشته $|z| \geq n\ ,\ z \in L$ داشته باشیم:

$$z = uvwxy\ ,\ |vx| \neq 0\ ,\ |vwx| \leq n\ ,\ \forall i \geq 0\ uv^i wx^i y \in L$$

2) عدد ثابت مثل n وجود دارد به طوری که برای هر رشته $|z| \geq n\ ,\ z \in L$ داشته باشیم:

$$z = xyw\ ,\ |y| \neq 0\ ,\ |xy| \leq n\ ,\ \forall i \geq 0\ xy^i \in L$$

3) هیچ عدد ثابت مثل n وجود ندارد به طوری که برای هر رشته $|z| \geq n\ ,\ z \in L$ داشته باشیم:

$$z = uvwxy\ ,\ |vx| \neq 0\ ,\ |vwx| \leq n\ ,\ \forall i \geq 0\ uv^i wx^i y \in L$$

4) هیچ کدام

۵- فرض کنید L زبانی منظم باشد، کدام یک از زبان های زیر منظم نیست؟

1) $L' = \{x \in L \mid x^R \in L\}$

2) $L' = \{x \in L \mid \exists y \in L\ \ x = yy\}$

3) $L' = \{x \in L \mid \exists y\ \ xy \in L\}$

4) $L' = \{x \in L \mid \exists y \in L\ \ \exists z \in L\ \ x = yz\}$

۶- کدام گزاره نادرست است؟

1) اشتراک دو زبان منظم روی یک مجموعه الفبای مشخص، حتما منظم است.

2) هر زبان نامنظم، زیر مجموعه یک زبان منظم است.

3) هر زبان ناتهی، حتما شامل یک زبان ناتهی و منظم است.

4) اجتماع تعداد دلخواهی از زبان های منظم، حتما منظم است.

۷- برای کدام تابع $f : N \to N$، زبان $L_f = \{0^n 1^{f(n)} \mid n \in N\} \subseteq \{0,1\}^*$ منظم نیست؟

1) $f(n) = \begin{cases} 2(n+1) & \text{n زوج} \\ 2n+3 & \text{n فرد} \end{cases}$

2) $f(n) = \begin{cases} 3 & \text{n زوج} \\ 4 & \text{n فرد} \end{cases}$

3) $f(n) = 235$

4) گزینه های 1و2

۸- کدام یک از زبان های زیر منظم هستند؟

1) $\{a^n b^l a^k : n > 5, l > 3, k \le 1\}$

2) $\{a^n b^l a^k : k \ge n + l\}$

3) $\{a^n b^l : n \le l\}$

4) $\{a^n b^l a^k : n + l + k > 5\}$

۹- کدام یک از زبان های زیر منظم هستند؟

1) $L = \{w : n_a(w) \ne n_b(w)\}$

2) $L = \{w_1 c w_2 : w_1, w_2 \in \{a,b\}^*, w_1 \ne w_2\}$

3) $L = \{a^n b^l a^k : n + l + k > 5\}$

4) $L = \{a^n b^l a^k : k \ne n + l\}$

۱۰- کدام یک از گزینه های زیر نادرست است؟

1) اگر L_1 و L_2 زبانهای نامنظم باشند، آنگاه $L_1 \cup L_2$ نیز نامنظم خواهد بود.

2) اگر L_1 و L_2 زبانهای منظم باشند، آنگاه $L = \{w \mid w \in L_1 \text{ and } w^R \in L_2\}$ نیز منظم خواهد بود.

3) الگوریتمی وجود دارد که می تواند تعیین کند که آیا یک زبان نوع سوم (منظم) نامتناهی است یا خیر.

4) الگوریتمی وجود دارد که می تواند تعیین کند که آیا یک زبان نوع سوم (منظم) تهی است یا خیر.

۱۱- کدام یک از زبان های زیر منظم هستند؟

1) $L = \{w_1 c w_2 : w_1, w_2 \in \{a,b\}^*, w_1 \ne w_2\}$

2) $L = \{u w w^R v : u, v, w \in \{a,b\}^+\}$

3) $L = \{a^n b^k : n > k\} \cup \{a^n b^k : n \ne k - 1\}$

4) $L = \{ww : w \in \{a,b\}^*\}$

پاسخ تشریحی

۱-۴) لم pumpingیک شرط کافی برای منظم نبودن یک زبان ارائه می دهد.

۲-۲) در مورد تقسیم دو زبان L و L' موارد زیر درست می باشند:

الف- اگر $\lambda \in L'$ باشد آنگاه $L \subseteq \frac{L}{L'}$ خواهد بود.

ب- اگر $L \cap L' \neq \phi$ باشد آنگاه $\lambda \in \frac{L}{L'}$ خواهد بود.

ج- اگر $L' = \phi$ باشد آنگاه $\frac{L}{L'} = \phi$ خواهد بود.

۳-۲) رابطه زیر برقرار است:

$$L(G_1) - L(G_2) = L(G_1) \cap \overline{L(G_2)} = \overline{\overline{L(G_1)} \cup L(G_2)}$$

همچنین اگر بخواهیم مکمل یک ماشین متناهی را پیدا کنیم، باید ابتدا آن را به ماشین متناهی قطعی تبدیل کنیم. که در این تست این عمل با d(M) انجام می شود. بنابراین داریم: $\overline{\overline{L(G_1)} \cup L(G_2)} = L(\overline{d(\overline{d(\overline{M_1})} + M_2)})$

۴-۴) طبق لم پمپاژ (تزریق) که به صورت زیر تعریف می شود، جواب گزینه 4 است.

اگر L یک زبان منظم نامتناهی باشد، آنوقت یک عدد صحیح مثبت m وجود دارد بطوریکه هر رشته w متعلق به L با $|w| \geq m$، می‌تواند به xyz تجزیه شود به طوری که:

الف- $y \neq \lambda$ ب- $|xy| \leq m$ ج- برای تمام $i \geq 0$ ها ، رشته xy^iz نیز در L موجود باشد.

۵-۲) با فرض $L = (a+b)^*$، آنگاه $L' = \{x \in L \mid \exists y \in L\ x = yy\}$ منظم نمی باشد.

۶-۴) زبان های منظم تحت اجتماع نامتناهی بسته نمی باشند.

۷-۱) در زبان $\{0^n 1^{f(n)}\}$، تعداد 1 ها تابعی از تعداد 0 ها می باشد که تابع f(n) در چهار گزینه به صورت های مختلف داده شده است. در گزینه های 2و3و4 نیازی به نگهداری n نمی باشد و بنابراین با یک ماشین متناهی، قابل پذیرش می باشند و در نتیجه زبان آنها منظم است.

بررسی گزینه 1: در این تابع باید تعداد 0 ها را بدانیم(در جایی ذخیره کنیم) تا بتوان تعداد 1 ها را بررسی کرد. این کار توسط ماشین متناهی ممکن نیست، بنابراین زبان منظم نمی باشد.(برای پیدا کردن تعداد یک ها باید تعداد صفرها یعنی n را در رابطه 2(n+1) و یا 2n+3 قرار داد.)

روش تستی : برای آنکه زبان داده شده منظم نباشد، باید تابع f یک به یک باشد. که گزینه 1 فقط اینچنین است.

۸-۴) برای اثبات منظم بودن زبان L_1، کافی است آن را به حالت هایی مثل $l = 0, k = 0, n > 5$ تقسیم کرد. در اینصورت می توان برای هر یک عبارات منظمی ارائه داد.

۹ـ۳)

۱۰-۱) در رابطه با زبان منظم L ، می‌توان با ارائه الگوریتم‌هایی، به پرسشهایی چون،" آیا L متناهی هست یا خیر؟" و یا"آیا L تهی هست یا خیر؟"، پاسخ داد. بنابراین گزینه های 3و4 درست می باشند.

۱۱-۲) L_1 منظم است. یکی از عبارات منظم برای L_1، به صورت زیر است:

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

فصل ششم: زبان ها و گرامرهای مستقل از متن

برای پوشش دادن برخی از ویژگی های زبان های برنامه سازی باید خانواده زبان ها را گسترش دهیم. بـه همـین دلیـل زبـان هـا و گرامر های مستقل از متن را مطرح می کنیم. زبان های مستقل از متن در طراحی زبان های برنامه سازی و ساخت کامپایلر کـاربرد دارد.

گرامرهای مستقل از متن

برای ساخت برنامه های قدرتمندتر باید تا حدی از قید محدویت های موجود در گرامرهای منظم رها شـویم. گرامرهـای مسـتقل از متن در طرف چپ همان محدودیت گرامرهای منظم را دارند ولی در طرف راست آزادتر هستند.

تعریف: گرامر مفروض $G=(V,T,S,P)$ در صورتی مستقل از متن خوانده می شود که تمام قوانین P بـه فـرم $A \to x$ باشـند که در آن $A \in V$ و $x \in (V \cup T)^*$.

به طور کلی شرط مستقل از متن بودن این است که در سمت چپ قوانین فقط یک متغیر وجود داشته باشد.

تعریف: زبان L مستقل از متن نامیده می شود، اگر و تنها اگر گرامر مستقل ازمتن G وجود داشته باشد بطوریکه L=L(G) .

- تمام گرامرهای منظم، مستقل از متن هستند و بنابراین هر زبان منظمی، یک زبان مستقل از متن نیز می باشد.
- خانواده زبان های منظم یکی از زیر مجموعه های محض خانواده زبان های مستقل از متن هستند.

مثال: آیا گرامر زیر مستقل از متن است؟

$S \to ABS$

$S \to \lambda$

$A \to aA$

$A \to a$

$B \to bB$

$B \to b$

حل: بله- چون در سمت چپ قواعد، یک متغیر وجود دارد. ■

مثال: گرامر $G=(\{S\},\{a,b\},S,P)$ با قوانین زیر مستقل از متن است:

$S \to aSa$

$S \to bSb$

$S \to l$

و زبان $L(G)=\{ww^R : w \in \{a,b\}^*\}$ مستقل از متن است. (این زبان منظم نمی باشد)

■

مثال: نشان دهید که زبان $L=\{a^n b^m \mid n \neq m\}$ مستقل از متن است.

حل: کافی است که برای این زبان یک گرامر مستقل از متن بسازیم.

دو حالت $n > m$ و $n < m$ را بررسی کرده:

$n < m$	$n > m$
$S \to S_1B$ $S_1 \to aS_1b \mid \lambda$ $B \to bB \mid b$	$S \to AS_1$ $S_1 \to aS_1b \mid \lambda$ $A \to aA \mid a$

و سپس آن دو را ترکیب می‌کنیم:

$S \to AS_1 \mid S_1B$

$S_1 \to aS_1b \mid \lambda$

$A \to aA \mid a$

$B \to bB \mid b$

توجه شود که این گرامر خطی نمی باشد.

مثال: گرامری با قوانین زیر مستقل از متن و غیر خطی می باشد:

$$S \to aSb \mid SS \mid l$$

و زبان تولید شده توسط این گرامر به صورت زیر می باشد:

$L = \{w \in \{a,b\}^* : n_a(v) \geq n_b(v)$ برای هر v پیشوند w $n_a(w) = n_b(w)\}$

تذکر: اگر به جای a و b از پرانتزهای باز و بسته استفاده کنیم، زبان L شامل رشته هایی مثل (( )) و ( ) ( ) ( ) می باشد که به عنوان مجموعه تمامی ساختارهای پرانتزهای تودرتوی متداول در زبان های برنامه سازی استفادهمی شوند.

مثال: گرامر مستقل از متنی برای زبان $L = \{a^n b^m : n \leq m+3\}$ بنویسید. $(n \geq 0, m \geq 0)$

حل: ابتدا حالت n=m+3 را حل می کنیم:

$$S \to aaaA$$
$$A \to aAb \mid B$$
$$B \to Bb \mid l$$

چون در این حالت حداقل سه a بوجود می آید، جواب هنوز کامل نمی باشد و برای بررسی حالتهای $n = 0,1,2,$ ، قانون زیر را اضافه می کنیم: $S \to l \mid aA \mid aaA$ ■

مثال: گرامر مستقل از متنی برای زبان $L = \{a^n b^m : 2n \leq m \leq 3n\}$ بنویسید. $(n \geq 0, m \geq 0)$

حل: $S \to aSbb \mid aSbbb \mid l$ ■

مثال: گرامر مستقل از متنی برای زبان زیر بنویسید. $(n \geq 0, m \geq 0, k \geq 0)$

$L = \{a^n b^m c^k : n = m$ یا $m \leq k\}$

حل: مساله شامل دو حالت است:

ب- $m \le k$ و n اختیاری است:	الف- n=m و k اختیاری است:
$S_2 \to BD$ $B \to aB \mid l$ $D \to bDc \mid E$ $E \to Ec \mid l$	$S_1 \to AC$ $A \to aAb \mid l$ $C \to Cc \mid l$

در نهایت ، قوانین را با $S \to S_1 \mid S_2$ شروع می کنیم. ■

مثال: گرامر مستقل از متنی برای زبان $L=\{a^n b^m c^k : k=|n-m|\}$ بنویسید. $(n \ge 0, m \ge 0, k \ge 0)$

حل: مسئله را به دو قسمت تقسیم می کنیم: $n=k+m$ و $m=k+n$. حالت اول با قونین زیر حل می شود:

$S \to aSc \mid S_1 \mid l$

$S_1 \to aS_1b \mid l$

مثال: گرامر مستقل از متنی برای مجموعه تمام عبارات منظم روی الفبای $\{a,b\}$ بنویسید.

حل:

$E \to E+E \mid E.E \mid E^* \mid (E) \mid l \mid f \mid a \mid b$

مثال: فرض کنید $L=\{a^n b^n : n \ge 0\}$،آنگاه $\overline{L}$ و L^* مستقل از متن هستند. همچنین به ازای هر $k \ge 1$، L^k مستقل از متن است.■

اشتقاق های سمت راست ترین و سمت چپ ترین

اگر در هر مرحله از اشتقاق، آخرین متغیر از سمت راست فرم جمله ای جایگزین شود، اشتقاق را سمت راست ترین مـی خـوانیم و اگر در هر مرحله از اشتقاق، آخرین متغیر از سمت چپ فرم جمله ای جایگزین شود، اشتقاق را سمت چپ ترین می خوانیم.

مثال: با توجه به گرامر زیر، برای رشته abbbb یک اشتقاق چپ و یک اشتقاق راست مشخص کنید.

$S \to aAB$

$A \to bBb$

$B \to A \mid l$

حل:

چپ:

$S \Rightarrow aAB \Rightarrow abBbB \Rightarrow abAbB \Rightarrow abbBbbB \Rightarrow abbbbB \Rightarrow abbbb$

راست:

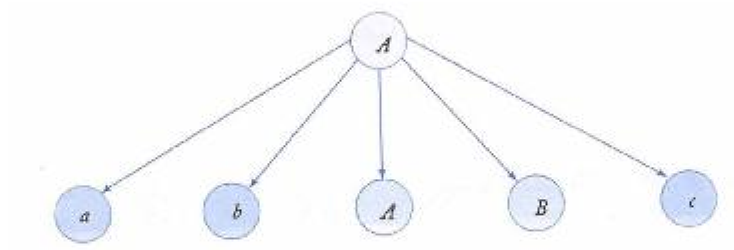
$S \Rightarrow aAB \Rightarrow aA \Rightarrow abBb \Rightarrow abAb \Rightarrow abbBbb \Rightarrow abbbb$

■

درخت های اشتقاق

برای نمایش اشتقاق ها می توان از درخت اشتقاق (تجزیه) نیز استفاده کرد. در این روش ترتیب بکارگیری قوانین اهمیتی ندارد. درخت اشتقاق، درخت مرتبی است که در آن گره ها با سمت چپ قوانین نامگذاری می شوند و فرزندان یک گره به معرفی سمت راست آن گره می پردازند و به ترتیب از چپ به راست، سمبل ها و متغیرهای سمت راست می آیند.

مثال: در شکل زیر بخشی از درخت اشتقاق معادل قانون $A \to abABc$ نمایش داده شده است:



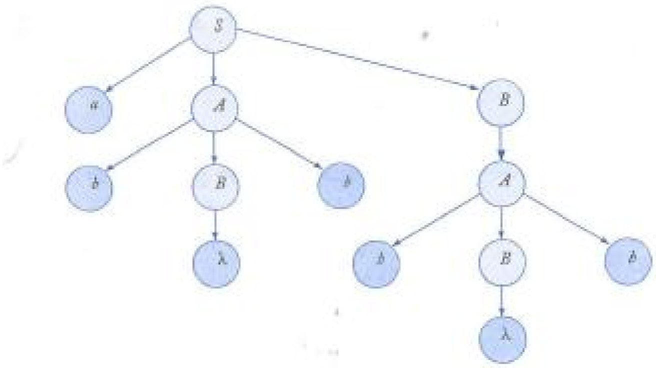
تذکر: در تمامی درخت های اشتقاق، با شروع از ریشه که با متغیر شروع گرامر نامگذاری می شود و خاتمه یافتن به برگ هایی که پایانی ها و یا l هستند، درخت تکمیل می شود.

تعریف: فرض کنید $G=(V,T,S,P)$ یک گرامر مستقل از متن باشد. یک درخت مرتب، درخت اشتقاقی برای G خواهد بود اگر و تنها اگر خواص زیر را داشته باشد: 1- ریشه دارای نام S است. 2- هر یک از برگها دارای نامی از $T \cup \{l\}$ باشد. 3- هر یک از گره های میانی دارای نامی از V است. 4- اگر یکی از گره ها دارای نام $A \in V$ بوده و فرزندان آن هم از چپ به راست به صورت $a_1, a_2, ..., a_n$ نامگذاری شوند، آنگاه P باید قانونی به فرم $A \to a_1 a_2 ... a_n$ داشته باشد. 5- برگ های دارای نام l، هیچ خواهر و برادری ندارند.

رشته ای از سمبل ها که با خواندن برگهای درخت از چپ به راست و حذف تمامی l های مسیر ایجاد شود، تولید درخت نامیده می شود.

مثال: نمایش درخت اشتقاق برای گرامر Gبا قوانین زیر:

$S \rightarrow aAB$
$A \rightarrow bBb$
$B \rightarrow A \mid l$



تولید این درخت ، یعنی abbbb ، رشته ای از L(G) است. ■

اگر G یک گرامر مستقل از متن باشد، آنگاه هر $w \in L(G)$ دارای یک اشتقاق چپ ترین و راست ترین خواهد بود.

تجزیه

تجزیه به معنای یافتن دنباله ای از قوانین است که یک $w \in L(G)$ از آن مشتق شده است. یک روش تجزیه به نام تجزیه جستجوی کامل وجود دارد که مدلی از تجزیه بالا به پایین است و می توان آنرا نتیجه یک درخت اشتقاق از ریشه به سمت پایین در نظر گرفت.این روش در اغلب موارد بسیار ناکارا می باشد. این تجزیه به صورت توانی بر حسب طول رشته تغییر کرده و هزینه زمانی روش مورد نظر را بسیار زیاد می کند.

قضیه:به ازای هر گرامر مستقل از متن، الگوریتمی وجود دارد که هر $w \in L(G)$ را در تعداد مراحلی متناسب با $|w|^3$ تجزیه می کند.

گرامر ساده(S–گرامر)

گرامر مستقل از متن G=(V,T,S,P)در صورتی گرامر ساده یا S-گرامر نامیده می شود که تمامی قوانین آن به فرم $A \rightarrow ax$ باشند که در آن $x \in V^*$ و $a \in T$ و $A \in V$ و هر زوج (A, a) حداکثر یک بار در P وجود داشته باشد.

مثال:گرامر زیر یک گرامر ساده نمی باشد، چون زوج (s,a) در دو قانون 1و3 وجود دارد.

1.$S \rightarrow aS$
2.$S \rightarrow bSS$
3.$S \rightarrow aSS$
4.$S \rightarrow c$

■

تذکر: بسیاری از ویژگی های زبان های برنامه سازی، بوسیله گرامرهای ساده قابل توصیف هستند.

مثال: یک S- گرامر برای $L = \{a^n b^n : n \geq 1\}$ بنویسید.

$S \rightarrow aA$
$A \rightarrow aAB \mid b$
$B \rightarrow b$

اگر G یک گرامر ساده باشد، آنگاه هر رشته w عضو L(G) را می توان با مجموعه عملیات های متناسب با $|w|$ تجزیه کرد.

ابهام در گرامر ها و زبان ها

گرامر مستقل از متن G در صورتی مبهم خوانده می شود که یک رشته $w\in L(G)$ وجود داشته باشد که حداقل دو درخت اشتقاق مجزا داشته باشد.

مثال: گرامر $S\rightarrow aS\,|\,aa\,|\,a$ مبهم است، چون برای رشته aa ، می توان دو درخت اشتقاق رسم کرد:

مثال: نشان دهید که گرامر زیر مبهم است.

$$S\rightarrow AB\,|\,aaB$$
$$A\rightarrow a\,|\,Aa$$
$$B\rightarrow b$$

حل: گرامر مبهم است، چون دو اشتقاق چپ ترین برای w=aab وجود دارد:

$$S\Rightarrow aaB\Rightarrow aab,$$
$$S\Rightarrow AB\Rightarrow AaB\Rightarrow aaB\Rightarrow aab.$$

مثال: گرامر G=(V,T,E,P) که V={E,I} و T={a,b,c,+,*,(,)} و قوانین زیر مفروض است:

$$E\rightarrow I\,|\,E+E\,|\,E*E\,|\,(E)$$
$$I\rightarrow a\,|\,b\,|\,c$$

این گرامر مبهم است، چون به طور نمونه برای رشته a+b*c دو درخت اشتقاق وجود دارد. برای رفع ابهام این گرامر، V را به صورت {E,I,T,F} در نظر گرفته و قوانین را به صورت زیر جایگزین می کنیم:

$$E\rightarrow T\,|\,E+T$$
$$T\rightarrow F\,|\,T*E$$
$$F\rightarrow (E)\,|\,I$$
$$I\rightarrow a\,|\,b\,|\,c$$

با توجه به گرامر جدید، فقط یک درخت اشتقاق برای رشته a+b*c می توان رسم کرد.

تذکر: در این مثال، ابهام ناشی از گرامر بود که توانستیم با معرفی یک گرامر فاقد ابهام معادل با آن، ابهام را رفع کنیم. اما در برخی موارد، ابهام در دل زبان است و از بین بردن آن مشکل است.

مثال: گرامر زیر مبهم است، این گرامر را رفع ابهام کنید. (n بیانگر عدد صحیح مثبت بدون علامت است.)

$$S\rightarrow T+S\,|\,S+T\,|\,T$$
$$T\rightarrow F*T\,|\,T*F\,|\,F$$
$$F\rightarrow (S)\,|\,n$$

حل:

$S \to TS'$

$S' \to +S \mid \lambda$

$T \to FT'$

$T' \to *T \mid \lambda$

$F \to (S) \mid n$

مثال: نشان دهید که گرامری با قوانین زیر مبهم است، اما زبان تولید شده بوسیله آن مبهم نیست.

$$S \to aSb \mid SS \mid l$$

حل: گرامر مبهم است، چون برای رشته w=ab دو اشتقاق وجود دارد:

$$S \Rightarrow aSb \Rightarrow ab,$$
$$S \Rightarrow SS \Rightarrow abS \Rightarrow ab.$$

یکی از گرامرهای غیر مبهم نظیر گرامر زیر می باشد. بنابراین زبان تولیده شده مبهم نیست.

$$S \to A \mid l$$
$$A \to aAb \mid ab \mid AA$$

تعریف: اگر L مستقل از متن بوده و هر گرامر تولید کننده L مبهم باشد، آنگاه این زبان ذاتا مبهم خوانده می شود.

- تمام s-گرامرها، فاقد ابهام هستند.
- زبان های منظم نمی توانند ذاتا مبهم باشند.

مجموعه تست

۱- در گرامر مستقل از متن G هیچ سمبل غیر پایانی A وجود ندارد به طوری که $A \overset{+}{\Rightarrow} UAV$. کدام گزینه صحیح است؟

1) زبان معادل آن منظم نیست
2) یک زبان منظم را معرفی می کند.
3) زبان معادل آن بی پایان و نامنظم است.
4) زبان معادل آن بی پایان ولی منظم است.

۲- زبان های L_1 و L_2 با تعاریف زیر را در نظر می گیریم :

$w_1, w_2 \in \{a,b\}^*$

$L_1 = \{w_1 w_2 \mid |w_1| = |w_2|, w_1 \neq w_2\}$

$L_2 = \{w_1 w_2 \mid w_1, w_2 \in \{0,1\}^*, n_a(w_1) = n_b(w_2)\}$

1) L_1 و L_2 هر دو آزاد از متن هستند.
2) L_1 از نوع آزاد از متن است ولی L_2 از این نوع نیست.
3) L_2 از نوع آزاد از متن است ولی L_1 از این نوع نیست.
4) هیچ کدام از دو زبان آزاد از متن نیست.

۳- گرامر روبرو را درنظر بگیرید. کدام یک از جملات زیر صحیح است؟

$S \to 0S1 \mid 1S0 \mid AA$
$A \to 0A \mid \lambda$
$A \to A1 \mid \lambda$

1) گرامر فوق یک گرامر مستقل از متن است که زبان منظم تولید می کند.
2) گرامر فوق یک گرامر مستقل از متن است که زبان نامنظم تولید می کند.
3) گرامر فوق یک گرامر وابسته به متن است که زبان منظم تولید می کند.
4) گرامر فوق یک گرامر خطی است که زبان نامنظم تولید می کند.

۴- اگر $L \subseteq \{0,1\}^*$ زبان گرامر زیر باشد، کدام گزاره نادرست است؟

$G: S \to 00S \mid X$
$X \to 11X \mid \lambda$

1) L منظم است.
2) L^c منظم است.
3) L مستقل از متن است.
4) $L = \{0^n 1^m\}$ (n+m زوج است)

۵- کدام گزاره در مورد گرامر $G: S \to SS \mid (S) \mid \lambda$ ، درست است؟

1) G با گرامر $[G': S \to S(S) \mid \lambda]$ معادل است.
2) G با گرامر $[G'': S \to S(SS) \mid \lambda]$ معادل نیست.
3) زبان G مستقل از متن نیست.
4) زبان G منظم است.

۶- کدامیک از زبان های زیر ذاتا مبهم هستند؟

1) $L_1 = \{a^n b^n c^m\} \cup \{a^n b^m c^m\}$
2) $L_2 = \{ww^R : w \in \{a,b\}^*\}$
3) $L_3 = \{a^n b^m c^k : k = n+m\}$
4) $L_4 = \{a^n b^n c^k : k \geq 3\}$

۷- عبارت منظم $(a \mid b)(a \mid b \mid 0 \mid 1)^*$ **وگرامر مستقل از متن** $\begin{array}{l} S \to aS' \mid bS' \\ S' \to aS' \mid bS' \mid 0S' \mid 1S' \mid \varepsilon \end{array}$ **داده شده اند:**

1) با هم هیچ ارتباطی ندارند.

2) هر دو یک زبان را تولید می نمایند.

3) اصولا عبارات منظم، زبان های منظم و گرامرهای مستقل از متن، زبان های مستقل از متن را تولید می کنند، لذا هیچ عبارت منظمی نمی تواند همان زبانی را تولید کند که یک گرامر مستقل از متن تولید می کند.

4) زبانهایی را که عکس هم می باشند، تولید می کنند.

۸- زبان های مستقل از متن L_1 **و** L_2 **را درنظر بگیرید. کدام گزینه در مورد زبان** $L = \{x \mid xy \in L_1 \text{ and } y \in L_2\}$ **درست است؟**

$L_1 = \{a^n baa^m \mid n \geq m \geq 0\}$

$L_2 = \{a^n b^m \mid n > m \geq 0\}$

1) $L = \{a^n ba^m \mid n \geq m \geq 0\}$

2) $L = \{a^n b \mid n \geq 0\}$

3) $L = \{a^n ba^{m+1} \mid n \geq m \geq 0\}$

4) $L = \{a^n ba \mid n \geq 0\}$

پاسخ تشریحی

۱-۱) اگر در گرامر مستقل از متنی هیچ سمبل غیر پایانی A وجود ندارد به طوری که $A \overset{+}{\Rightarrow} UAV$، آنگاه زبان معادل آن منظم نیست.

۲-۳) با توجه به دو زبان زیر، مشخص است که فقط L_2 مستقل از متن است:

$$L_1 = \{w_1 w_2 \mid |w_1| = |w_2|, w_1 \neq w_2\}$$

$$L_2 = \{w_1 w_2 \mid w_1, w_2 \in \{0,1\}^*, n_a(w_1) = n_b(w_2)\}$$

۳-۱) زبان گرامر داده شده برابر است با:

$$0^n AA1^n \cup 1^n AA0^n = 0^n (0+1)^* 1^n \cup 1^n (0+1)^* 0^n = (0+1)^*$$

که زبانی منظم می باشد.

۴-۴) زبان گرامر داده شده برابر $^*(11)^*(00)$ می باشد که منظم است و مکمل آن نیز منظم است. بنـابراین گزینـه 1و2 درسـت است. همچنین هر زبان منظم، مستقل از متن نیز می باشد و گزینه 3 نیز درست است. گزینه 4 نادرست اسـت، چـون مـثلا رشـته 01 را می پذیرد که به $^*(11)^*(00)$ تعلق ندارد.

۵-۱) گزینه 1 درست است، چون 3 گرامر زیر معادل هستند:

$$G1: S \to SS \mid (S) \mid l$$
$$G2: S \to S(S) \mid l$$
$$G3: S \to S(SS) \mid l$$

زبان این گرامرها، نامنظم و مستقل از متن است و رشته هایی شامل عبارات پرانتزی خوش ساخت می باشد.

۶-۱) زبان $L_1 = \{a^n b^n c^m\} \cup \{a^n b^m c^m\}$ ذاتا مبهم است، چون برای آن هیچ گرامر غیر مبهمی نمی‌توان نوشت.

۷-۲) عبارت منظم و گرامر مستقل از متن داده شده، هر دو یک زبان را تولید می کنند.

۸-۱) اگر ساده ترین رشته قابل تولید توسط L_2 یعنی a را از سمت راست زبان L_1 حذف کنیم،آنگاه زبان زیر باقی می ماند:

$$L = \{a^n b a^m \mid n \geq m \geq 0\}$$

فصل هفتم: ساده سازی گرامرهای مستقل از متن - فرم های نرمال

در تعریف گرامرهای مستقل از متن، هیچ محدودیتی برای سمت راست قانون در نظر گرفته نشده است که این آزادی در برخی استدلال ها، مشکل ایجاد می کند. در بسیاری موارد بهتر است محدودیت شدیدی در مورد گرامرها قائل شویم.

روش های تبدیل گرامرها

با فرض اینکه L یک زبان مستقل از متن بوده و $G=(V,T,S,P)$ گرامر مستقل از متن برای $L-\{l\}$ باشد. آنگاه، گرامری که با افزودن متغیرجدید S_0 به V، به عنوان متغیر شروع و افزودن قوانین $S_0 \to S \mid l$ به P ایجاد شود، زبان L را ایجاد خواهد کرد. بنابراین هر نتیجه کلی که بتوانیم برای $L-\{l\}$ داشته باشیم، همواره قابل تبدیل به L خواهد بود. همچنین، با داشتن هر گرامر مستقل از متن G، روشی برای بدست آوردن G' وجود دارد، بطوریکه $L(G')=L(G)-\{l\}$.بنابراین در این بحث، برای راحتی کار فقط زبان های فاقد l را بررسی می کنیم.

یک قانون جایگزینی مناسب

بسیاری از قوانین می توانند با استفاده از جایگزینی، گرامرهای معادلی را برای تولید زبان بوجود آورند.

مثال:گرامر G با قوانین زیر مفروض است.گرامر معادل حاصل از جایگزینی متغیر B را بدست آورید.

$$A \to a \mid aaA \mid abBc$$
$$B \to abbA \mid b$$

حل:

$$A \to a \mid aaA \mid ababbAc \mid abbc$$
$$B \to abbA \mid b$$

حذف متغیرها و قوانین بی فایده

یک متغیر مفید است اگر و تنها اگر در حداقل یک اشتقاق حضور داشته باشد. عوامل غیرمفید بودن یک متغیر عبارتند از:

1- قابل دسترس نبودن از طریق متغیر شروع گرامر2- ناتوانی در اشتقاق یک رشته پایانی

تعریف: قانونی که شامل یک متغیر بی فایده باشد، قانون بی فایده نامیده می شود.

مثال:در گرامر زیر، متغیر A غیر مفید می باشد، چون نمی تواند رشته ای از پایانی ها را تولید کند:

$$S \to aSb \mid A \mid l$$
$$A \to aA$$

بنابراین قانون $S \to A$ را می توان حذف کرد، بدون اینکه تغییری در زبان ایجاد کند. ■

مثال: در گرامر زیر متغیر B بی فایده می باشد، چون از طریق متغیر شروع یعنی S ، قابل دسترس نمی باشد:

$$S \to A$$
$$A \to aA \mid l$$
$$B \to bA$$

بنابراین قانون $B \to bA$ بی فایده است و می توان آن را حذف کرد، بدون اینکه تغییری در زبان ایجاد کند. ■

مثال: در گرامر زیر متغیر C بی فایده است، چون یک رشته پایانی را تولید نمی کند. همچنین متغیر B بی فایده است، چون از متغیر شروع قابل دستیابی نمی باشد:

$$S \to aS \mid A \mid C$$
$$A \to a$$
$$B \to aa$$
$$C \to aCb$$

بنابراین می توان متغیرهای C و B و قوانین مربوط به آنها را حذف کرد. گرامر نهایی به صورت زیر می باشد:

$$S \to aS \mid A$$
$$A \to a$$

■

مثال: در گرامر زیر متغیرهای بی فایده را حذف کنید.

$S \to AC \mid BS \mid B$, $A \to aA \mid aF$, $B \to CF \mid b$, $C \to cC \mid D$, $D \to aD \mid BD \mid C$, $E \to aA \mid BSA$, $F \to bB \mid b$

حل: ابتدا متغیرهایی را که به رشته‌ای از الفبا نمی‌رسند یعنی C و D را حذف می‌کنیم:

$S \to BS \mid B$, $A \to aA \mid aF$, $B \to b$, $E \to aA \mid BSA$, $F \to bB \mid b$

سپس متغیرهایی را که نمی‌توان از S به آنها رسید، یعنی A و E و F را حذف می‌کنیم:

$S \to BS \mid B$, $B \to b$

مثال:در گرامر زیر متغیرها و قواعد بی‌فایده را مشخص کنید.

$S \to aS \mid A \mid C$, $A \to a$, $B \to aa$, $C \to aCb$

حل: متغیرC بی فایده است (به ترمینال ختم نمی‌شود). با حذف C و قواعد مربوط به آن داریم:

$S \to aS \mid A$, $A \to a$, $B \to aa$

همچنین قاعده $B \to aa$ نیز غیرضروری است و در نهایت داریم:

$S \to aS \mid A$, $A \to a$

حذف قوانین بی فایده لزوما موجب تولید گرامرهای کمینه نخواهد شد. به طور نمونه در گرامر $S \to aA; A \to a$ هیچ قانون بی فایده، قانون واحد یا قانون l به چشم نمی خورد. اما چون $S \to aa$ یکی از گرامرهای متناظر است، گرامر فوق کمینه نمی باشد.

حذف قوانین l

هر قانونی از یک گرامر مستقل از متن به فرم $A \to l$ را قانون l می گویند. این قوانین در بعضی مواقع نامطلوب می باشند. هر متغیر A که اشتقاق $A \overset{*}{\Longrightarrow} l$ برای آن امکان پذیر باشد را **متغیر میرا** می نامند.

برخی گرامرها زبان هایی را تولید می کنند که هر چند فاقد l هستند، تعدادی متغیر میرا یا قانون l در آنها وجود دارند. در این موارد، می توان قوانین l را حذف کرد.

مثال:گرامر زیر، زبان $\{a^n b^n : n \ge 1\}$ را تولید می کند که فاقد l می باشد:

$$S \to aAb$$
$$A \to aAb \mid l$$

برای حذف قانون $A \to l$، دو قانون جدید که با جایگزینی l در A های سمت راست بدست آمده اند را بـه گرامـر اضـافه مـی کنیم:

$$S \to aAb \mid ab$$
$$A \to aAb \mid ab$$

مثال: قوانین l را در گرامر زیر حذف کنید.

$$\mathrm{S \to AB}$$
$$\mathrm{A \to aA \mid \lambda}$$
$$\mathrm{B \to bB \mid \lambda}$$

حل:

$$\mathrm{S \to AB \mid A \mid B \mid \lambda}$$
$$\mathrm{A \to aA \mid a}$$
$$\mathrm{B \to bB \mid b}$$

مثال:یک گرامر مستقل از متن فاقد قانون l هم ارز با گرامر زیر پیدا کنید.

$$S \to ABaC$$
$$A \to BC$$
$$B \to b \mid l$$
$$C \to D \mid l$$
$$D \to d$$

حل:

$$S \to ABaC \mid BaC \mid AaC \mid ABa \mid aC \mid Aa \mid Ba \mid a$$
$$A \to B \mid C \mid BC$$
$$B \to b$$
$$C \to D$$
$$D \to d$$

حذف قوانین واحد

هر قانونی از یک گرامر مستقل از متن به فرم $\mathrm{A \to B}$ که در آن $A, B \in V$، قانون واحد یا یکه نامیده می شود. این قوانین گـاهی اوقات نامطلوب هستند و باید حذف شوند.

مثال: از گرامر زیر قوانین واحد را حذف کنید.

$$\mathrm{S \to Aa \mid B}$$
$$\mathrm{B \to A \mid bb}$$
$$\mathrm{A \to B \mid bc \mid a}$$

حل:

$$S \to a \mid bc \mid bb \mid Aa$$
$$B \to a \mid bb \mid bc$$
$$A \to a \mid bb \mid bc$$

توجه کنید که در اثر حذف قوانین واحد، متغیر B و قوانین مربوط به آن بی فایده شده اند.

ممکن است حذف قوانین l ، باعث تولید قوانین واحد شود که قبلا وجود نداشته اند.

مثال: در گرامر زیر قوانین l را حذف کنید.

$$S \to aA$$
$$A \to BB$$
$$B \to aBb \mid l$$

حل: با حذف قوانین l جواب زیر بدست می آید. در این گرامر قانون واحد $A \to B$ تولید شده که قبلا وجود نداشته است:

$$S \to aA \mid a$$
$$A \to BB \mid B$$
$$B \to aBb \mid ab$$

مثال: در گرامر زیر تمامی قوانین l ، واحد و بی فایده را حذف کنید.

$$S \to aA \mid aBB$$
$$A \to aaA \mid l$$
$$B \to bC \mid bbC$$
$$C \to B.$$

حل: ابتدا قانون l را حذف می کنیم:

$$S \to aA \mid a \mid aBB$$
$$A \to aaA \mid aa$$
$$B \to bC \mid bbC$$
$$C \to B.$$

سپس قانون واحد $C \to B$ و اشتقاق های آن را حذف می کنیم:

$$S \to aA \mid a \mid aBB$$
$$A \to aaA \mid aa$$
$$B \to bC \mid bbC$$
$$C \to bC \mid bbC$$

نهایتا اینکه B و C بی فایده هستند، بنابراین داریم:

$$S \to aA \mid a$$
$$A \to aaA \mid aa$$

زبان تولید شده بوسیله این گرامر $L((aa)^* a)$ می باشد.

فرم‌های نرمال گرامر مستقل از متن

برای گرامر مستقل از متن، دو فرم نرمال چامسکی و گریباخ وجود دارد.

فرم نرمال چامسکی

یک گرامر مستقل از متن در صورتی در فرم نرمال چامسکی قرار دارد که تمام قوانین آن به یکی از دو فرم $A \to BC$ و یا $A \to a$ باشد. که در آن A,B,C عضو V بوده و a عضو T است.

مثال: گرامر زیر در فرم نرمال چامسکی قرار دارد:

$S \to AS \mid BS \mid a$

$A \to SA \mid a$

$B \to SB \mid b$

مثال: گرامر زیر را به فرم نرمال چامسکی تبدیل کنید.

$S \to ABa \;,\; A \to aab \;,\; B \to Ac$

حل:

قدم دوم : نرمال کردن قانون اول و سوم	قدم اول : معرفی متغیرهای جدید X,Y,Z
$S \to AT$ $T \to BX$ $A \to XF$ $F \to XY$ $B \to AZ$ $X \to a$ $Y \to b$ $Z \to c$	$S \to ABX$ $X \to a$ $A \to XXY$ $Y \to b$ $B \to AZ$ $Z \to c$

مثال: گرامر زیر را به فرم نرمال چامسکی تبدیل کنید.

$S \to aAB \;,\; A \to aA \mid b \;,\; B \to bB \mid b$

حل:

$$\begin{array}{ll} S \to TAB & S \to Tk \\ T \to a & K \to AB \\ A \to TA \mid b \Rightarrow & T \to a \\ B \to FB \mid b & A \to TA \mid b \\ F \to b & B \to FB \mid b \\ & F \to b \end{array}$$

مثال: گرامر زیر را به فرم نرمال چامسکی تبدیل کنید.

$S \to aAbB \mid ab \;,\; A \to ABS \mid a \;,\; B \to bb$

حل:

$$\begin{array}{l} S \to TAFB\,|\,TF \\ A \to ABS\,|\,a \\ B \to FF \\ T \to a \\ F \to b \end{array} \Rightarrow \begin{array}{l} S \to Tk\,|\,TF \\ k \to AFB \\ A \to AU\,|\,a \\ U \to BS \\ B \to FF \\ T \to a \\ F \to b \end{array} \Rightarrow \begin{array}{l} S \to Tk\,|\,TF \\ k \to AX\,|\,a \\ X \to FB \\ A \to AU\,|\,a \\ U \to BS \\ B \to FF \\ T \to a \\ F \to b \end{array}$$

■

قضیه: هرگرامر مستقل از متن در صورتیکه شامل l نباشد، دارای گرامری هم ارز، در فرم نرمال چامسکی است.

در فرم نرمال چامسکی، تولید رشته‌ای به طول n دارای اشتقاقی با طول 2n-1 می‌باشد.

(n اشتقاق از $A \to a$ و n-1 اشتقاق از $A \to BC$)

فرم نرمال گریباخ

یک گرامر مستقل از متن در فرم نرمال گریباخ است هرگاه تمام قوانین آن به فرم $A \to ax$ باشند، کـه در آن، $x \in V^*, a \in T$

مثال: گرامر زیر را به فرم نرمال گریباخ در آورید.

$S \to AB\,,\ A \to aA\,|\,bB\,|\,b\,,\ B \to b$

حل: قاعده $S \to AB$ با تعریف فرم نرمال گریباخ مغایر است:

$S \to aAB\,|\,bBB\,|\,bB\ ,\ A \to aA\,|\,bB\,|\,b\ ,\ B \to b$

■

مثال: گرامر زیر را به فرم نرمال گریباخ تبدیل کنید.

$S \to abSb\,|\,aa$

حل: متغیرهای جدید A و B را معرفی می کنیم که مترادف با a و b هستند:

$S \to aBSB\,|\,aA\,,\ A \to a\ ,\ B \to b$

■

مثال: گرامر زیر را به فرم نرمال گریباخ تبدیل کنید.

$S \to ab\,|\,aS\,|\,aaS$

حل:

$$\begin{array}{l} S \to aX\,|\,aS\,|\,aYS \\ X \to b \\ Y \to a \end{array}$$

قضیه : به ازای هرگرامر مستقل از متن در صورتیکه شامل l نباشد، یک گرامر معادل به فرم نرمال گریباخ وجود دارد.

در فرم نرمال گریباخ، برای تولید رشته‌ای به طول n ، به اشتقاقی با طول n نیاز است. چون در هر مرحلـه یکـی از نمادهـای رشته ایجاد می‌شود.

یک الگوریتم عضویت برای گرامرهای مستقل از متن

توسط الگوریتم CYK که توسط کوک، یانگر و کسامی ابداع شد، می‌توان عضویت یا عدم عضویت یک رشته در زبانی کـه بوســیله گرامری که در فرم نرمال چامسکی تولید می شود را تصمیم گیری کرد. پیچیدگی این الگوریتم $|w|^3$ می‌باشد. این الگوریتم مساله را به بخش های کوچکتر تقسیم می کند. روش کار در کتاب لینز آورده شده است.

مجموعه تست

۱- حداقل پیچیدگی زمانی الگوریتم تجزیه ای که بتواند هر رشته متعلق به یک گرامر مستقل از متن مبهم دلخواه بــه فرم نرمال چامسکی را تجزیه(پارس) کند، کدام است؟ (الگوریتم تجزیه، گرامر را به هیچ وجه تغییر نمی دهد)

$O(n^3 \log n)$ (1 $O(2^n)$ (2 $O(n^3)$ (3 $O(n^4)$ (4

۲- کدام گزینه در مورد گرامر زیر درست است؟ (علوم کامپیوتر- دولتی ۸۸)

$S \to AB$
$A \to 0 \mid AS$
$B \to 1$

1) یک گرامر با فرم نرمال چامسکی است.

2) یک گرامر با فرم نرمال گریباخ است.

3) یک گرامر با فرم نرمال چامسکی و فرم نرمال گریباخ است.

4) هیچ کدام

۳- گرامر (G: ({S,X,Y},{A,B},S,R را با مجموعه قوانین زیر درنظر بگیرید. کدام گزاره نادرست است؟

$R: S \to XY$
$S \to a$
$X \to YS \mid b$
$Y \to XS \mid b$

1) گرامر G یک گرامر مستقل از متن است.

2) گرامر G به فرم نرمال چامسکی است.

3) baba توسط گرامر G تولید می شود.

4) baba فقط به یک روش از روی قوانین گرامر G تولید می شود.

۴- اگر از گرامرزیر، قوانین نامطلوب (l ، واحد و بی فایده)حــذف شــوند، تعــداد قواعــد باقیمانــده چیست؟

$S \to aA \mid aBB$
$A \to aaA \mid l$
$B \to bB \mid bbC$
$C \to B$

3 (1 5 (2 4 (3 6 (4

۵- گرامر زیر را درنظر بگیرید. فرم نرمال گریباخ آن عبارت است از:

$S \to ABb \mid a$
$A \to aaA \mid B$
$B \to bAb$

1)
$S \to aA_1ABA_2 \mid bAA_2BA_2 \mid a$
$A \to aA_1A \mid bAA_2$
$B \to bAA_2$
$A_1 \to a$
$A_2 \to B$

2)
$S \to aA_1AB \mid bAA_2BA_2 \mid a$
$A \to aA_1A \mid bA_2$
$B \to bAA_1$
$A_1 \to a$
$A_2 \to b$

3)
$S \to aA_1A \mid bBA_2A \mid a$
$A \to aA_1A \mid bA_2A$
$B \to bAA_2$
$A_1 \to a$
$A_2 \to b$

4) هیچکدام

پاسخ تشریحی

۱-۳) حداقل پیچیدگی زمانی الگوریتم تجزیه ای که بتواند هر رشته متعلق به یک گرامر مستقل از متن مبهم دلخواه به فرم نرمال چامسکی را تجزیه کند، $O(n^3)$ است.

۲-۱) قوائد در گرامر با فرم نرمال چامسکی، به صورت $A \to BC$ و یا $A \to a$ می باشد.

۳-۴) baba به دو روش زیر قابل تولید است:

$$S \to XY \to YSY \to bSY \to baY \to baXS \to babS \to baba$$

$$S \to XY \to XXS \to XXa \to YSXa \to bSXa \to baXa \to baba$$

۴-۳) در گرامر داده شده ابتدا قوانین l را حذف می کنیم:

$$S \to aA \mid a \mid aBB$$
$$A \to aaA \mid aa$$
$$B \to bB \mid bbC$$
$$C \to B$$

سپس قوانین واحد را حذف می کنیم:

$$S \to aA \mid a \mid aBB$$
$$A \to aaA \mid aa$$
$$B \to bB \mid bbC$$
$$C \to bB \mid bbC$$

و در نهایت قوائد بی فایده را حذف می کنیم (متغیرهای B و C ، متغیرهای غیر مفید می باشند.)

$$S \to aA \mid a$$
$$A \to aaA \mid aa$$

مشاهده می شود که گرامر نهایی دارای چهار قائده می باشد.

۵-۱) یک گرامر مستقل از متن در فرم نرمال گریباخ است هرگاه تمام قوانین آن به فرم $A \to ax$ باشند، که در آن،

$x \in V^*, a \in T$.

فصل هشتم: اتوماتای پشته ای

ماشین‌های پشته‌ای پذیرنده زبانهای مستقل از متن می‌باشند. در فصل‌های قبلی دیدیم که هر زبان منظمی را می‌توان توسط یک ماشین متناهی شناسایی کرد و همچنین هر زبانی که توسط یک ماشین متناهی پذیرفته شود، زبانی است منظم. ماشین پشته‌ای علاوه بر اجزاء ماشین متناهی دارای یک پشته نیز می‌باشد. ماشین PDA (Push Down Automaton)، علاوه بر خواندن نوار ورودی از چپ به راست، قادر به نوشتن هر تعداد نماد درون یک پشته (push) و بازیابی آنها (pop) می‌باشد.

وقتی رشته‌ای از زبان $\{a^n b^n \mid n \geq 0\}$ پویش می‌شود، نه تنها که باید کنترل شود که aها قبل ازbها باشند، بلکه a ها هم باید شمارش شوند و چون n نامحدود است، نیاز به ماشینی است که باید بتواند بدون هیچ محدودیتی این شمارش را انجام دهد. در بررسی $\{ww^R\}$ نه تنها نیاز به شمارش نامحدود است، بلکه ترتیب معکوس حروف باید تطبیق شود. در نتیجه نیاز به یک پشته با ظرفیت نامحدود می باشد.

اتوماتای پشته ای نامعین

پذیرنده پشته‌ای نامعین (NPDA) بوسیله هفت تایی $M = (Q, \sum, \Gamma, \delta, q_0, Z, F)$ تعریف می‌شود، که در آن:

Q : مجموعه متناهی از حالت های داخلی واحد کنترل

Σ:الفبای ورودی

Γ:الفبای پشته

d:تابع انتقال $(Q \times (\sum \cup \{\lambda\}) \times \Gamma \rightarrow Q \times \Gamma^*)$

q_0:حالت شروع واحد کنترل

Z : سمبل ته پشته

F : مجموعه حالات پایانی

قانون $d(q_i, a, A) = (q_j, B)$ ، یعنی اگر ماشین در وضعیت q_i باشد و a مقابل هد نوار خوان باشد و A بالای پشته باشد، آنگاه ماشین به حالت q_j تغییر وضعیت داده و A را از پشته pop کرده و B را push می‌کند.

مثال: نحوه عملکرد اتومات زیر را تشریح کنید.

$$d(q_1, a, b) = \{(q_2, cd), (q_3, l)\}$$

حل: هرگاه واحد کنترل در حالت q_1 قرار گیرد، سمبل ورودی خوانده شده a و سمبل واقع در بالای پشته b باشد، آنگاه یکی از دو حالت زیر اتفاق می افتد:

1- واحد کنترل به حالت q_2 رفته و رشته cd ، جایگزین b در بالای پشته می شود.

2- واحد کنترل به حالت q_3 رفته و سمبل b از بالای پشته حذف می شود.

تذکر: درج رشته در پشته، از انتهای سمت راست رشته شروع شده و سمبل به سمبل انجام می شود.

مثال:یک npda با فرض :

$Q=\{q_0,q_1,q_2,q_3\}$

$\sum=\{a,b\}$

$\Gamma=\{0,1\}$

$z=0$

$F=\{q_3\}$

با حالت شروع q_0 و تابع انتقال :

$d(q_0,a,0)=\{(q_1,10),(q_3,l)\}$

$d(q_0,l,0)=\{(q_3,l)\}$

$d(q_1,a,1)=\{(q_1,11)\}$

$d(q_1,b,1)=\{(q_2,l)\}$

$d(q_2,b,1)=\{(q_2,l)\}$

$d(q_2,l,0)=\{(q_3,l)\}$

را در نظر بگیرید. نحوه عملکرد این اتومات را تشریح کنید.

حل:

توسط انتقال $d(q_1,a,1)=\{(q_1,11)\}$ پس از خواندن یک سمبل a از ورودی، یک سمبل 1 به پشته اضافه می شود و در انتقال $d(q_2,b,1)=\{(q_2,l)\}$، به محض برخورد با یک b از ورودی، یک سمبل 1از پشته حذف می کند. به کمک این دو انتقال، تعداد a ها شمارش شده و با تعداد b ها تطبیق داده می شود.

پس از تجزیه و تحلیل بقیه انتقالات، مشاهده می کنیم که تنها در صورتی npda به حالت نهایی می رود که رشته ورودی جزء زبان $L=\{a^nb^n:n\geq 0\}\mathbf{U}\{a\}$ باشد.

مثال: یک npda برای زبان L بسازید.

$L=\{w\in\{a,b\}^*:n_a(w)=n_b(w)\}$

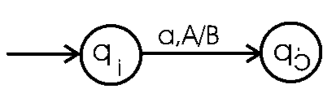
حل: مشابه مثال قبل است، با این تفاوت که ترتیب a ها و b ها اهمیتی ندارد. (علامت Z معرف انتهای پشته است.)

$\delta(q_0,\lambda,Z)=\{q_f,Z\}$, $\delta(q_0,a,Z)=\{q_0,oZ\}$, $\delta(q_0,b,Z)=\{q_0,1Z\}$, $\delta(q_0,a,0)=\{q_0,00\}$

$\delta(q_0,b,o)=\{q_0,\lambda\}$, $\delta(q_0,a,1)=\{q_0,\lambda\}$, $\delta(q_0,b,1)=\{q_0,11\}$

■

گراف انتقال برای $\delta(q_i,a,A)=(q_j,B)$ به صورت زیر می باشد.



یال گراف با سه مولفه "سمبل ورودی جاری، سمبل واقع در بالای پشته و رشته ای که در بالای پشته جایگزین می شود" برچسب دار می شود.

تعریف: زبان مورد پذیرش توسط یک npda ، مجموعه تمام رشته هایی است که npda با رسیدن به انتهای آن رشته ها، می تواند در حالت پایانی قرار گیرد.

مثال: یک npda برای زبان $L = \{ww^R : w \in \{a,b\}^+\}$ بسازید.

حل: رشته abba، نمونه ای از رشته های این زبان است. این رشته از دو بخش تشکیل شده است: w و w^R . در حین خواندن بخش اول، سمبل بعدی را در پشته push می کنیم. برای بخش دوم، سمبل ورودی جاری را با سمبل بالای پشته مقایسه کرده و اینکار را تا زمانیکه دو سمبل برابر هستند، ادامه می دهیم. با استفاده از ماهیت نامعین بودن npda ، می توان به درستی وسط رشته را حدس زده و حالت را در این نقطه عوض کرد.

یک مجموعه برای درج کردن:

$$d(q_0, a, a) = \{(q_0, aa)\}$$
$$d(q_0, b, a) = \{(q_0, ba)\}$$
$$d(q_0, a, b) = \{(q_0, ab)\}$$
$$d(q_0, b, b) = \{(q_0, bb)\}$$
$$d(q_0, a, z) = \{(q_0, az)\}$$
$$d(q_0, b, z) = \{(q_0, ba)\}$$

یک مجموعه برای حدس زدن وسط رشته:

$$d(q_0, l, a) = \{(q_1, a)\}$$
$$d(q_0, l, b) = \{(q_1, b)\}$$

یک مجموعه برای برابری w^R با محتویات پشته:

$$d(q_1, a, a) = \{(q_1, l)\}$$
$$d(q_1, b, b) = \{(q_1, l)\}$$

شناسایی نهایی:

$$d(q_1, l, z) = \{(q_2, z)\}$$

پیکربندی لحظه ای

سه تایی (q,w,u) که در آن، q حالت جاری واحد کنترل، w بخش خوانده نشده رشته ورودی و u محتویات پشته(چپ ترین سمبل بیانگر بالای پشته) است، پیکربندی لحظه ای یک اتومات پشته ای خوانده می شود. انتقال از یک پیکربندی به پیکربندی دیگر را بوسیله نشانه **a** نمایش می دهند.

مثال:دنباله تغییرات پیکربندی برای پذیرش رشته abba در مثال قبل :

$$(q_0, abba, z)\ \mathbf{a}\ (q_0, bba, az)\ \mathbf{a}\ (q_0, ba, baz)\ \mathbf{a}\ (q_1, ba, baz)\ \mathbf{a}\ (q_1, a, az)$$
$$\mathbf{a}\ (q_1, l, z)\ \mathbf{a}\ (q_2, z).$$

توجه شود که در حرکت سوم از ویژگی نامعین بودن اتومات برای تشخیص وسط رشته استفاده شد. در این حالت دو انتخاب زیر برای آن ممکن بود، که از انتخاب دوم استفاده کرد:

$$d(q_0, b, b) = \{(q_0, bb)\}$$
$$d(q_0, l, b) = \{\{q_1, b)\}$$

مثال: npda ای بسازید که زبان $L=\{a^n b^{2n} : n \geq 0\}$ را روی الفبای $\Sigma=\{a,b\}$ پذیرش کند؟

حل: برای هر یک از a ها، دو سمبل روی پشته قرار داده، در حالیکه برای هر یک از b ها از یک سمبل استفاده می شود:

$d(q_0, l, z)=\{(q_f, z)\}$
$d(q_0, a, z)=\{(q_1, 11z)\}$
$d(q_0, a, 1)=\{(q_1, 111)\}$
$d(q_1, b, 1)=\{(q_1, l)\}$
$d(q_1, l, z)=\{(q_f, z)\}$

مثال: npda زیر چه زبانی را پذیرش می کند؟

$M=(\{q_0,q_1,q_2\},\{a,b\},\{a,b,z\},d,q_0,z,\{q_2\})$
$d(q_0,a,z)=\{(q_1,a),(q_2,l)\}$
$d(q_1,b,a)=\{(q_1,b)\}$
$d(q_1,b,b)=\{(q_1,b)\}$
$d(q_1,a,b)=\{(q_2,l)\}.$

حل: با در نظر گرفتن مسیرها، مشخص می شود که زبان زیر را می پذیرد:

$L=\{a\} \mathbf{U} L(abb^*a)$

مثال: یک npda با حداکثر دو حالت داخلی پیدا کنید که زبان $L(aa^*ba^*)$ را بپذیرد.

حل: در این مثال آنچه معمولا بوسیله حالت های مختلف انجام می شد، بوسیله سمبل داخل پشته انجام می شود:

$d(q_0,a,z)=\{(q_0,1)\}$
$d(q_0,a,1)=\{(q_0,1)\}$
$d(q_0,b,1)=\{(q_0,2)\}$
$d(q_0,a,2)=\{(q_0,2)\}$
$d(q_0,l,2)=\{(q_f,2)\}$

مثال: آیا dfa ای وجود دارد که همان زبان pda زیر را پذیرش کند؟

$M=(\{q_0,q_1\},\{a,b\},\{z\},d,q_0,z,\{q_1\})$
$d(q_0,a,z)=\{(q_1,z)\}$
$d(q_0,b,z)=\{(q_0,z)\}$
$d(q_1,a,z)=\{(q_1,z)\}$
$d(q_1,b,z)=\{(q_0,z)\}$

حل: بله- در واقع این pda از پشته استفاده نمی کند و در عمل مانند یک dfa عمل می کند. بنابراین می توان انتقالات حالت را از pda گرفته و روبط زیر را بدست آورد.

$d(q_0,a)=q_1$
$d(q_0,b)=q_0$
$d(q_1,a)=q_1$
$d(q_1,b)=q_0$

اتوماتای پشته ای برای زبان های مستقل از متن

می خواهیم نشان دهیم که به ازای هر زبان مستقل از متن یک npda وجود دارد که آنرا پذیرش می کند. برای ساده تر شدن بحث، فرض می کنیم که گرامر تولید کننده زبان به فرم گریباخ باشد.

مثال: pda ای بسازید که زبان تولید شده بوسیله گرامر زیر را پذیرش کند.

$S \rightarrow aSA \mid a$
$A \rightarrow bB$
$B \rightarrow b.$

حل: اتومات دارای سه حالت $\{q_0, q_1, q_2\}$است که q_0 حالت شروع و q_2 حالت پایانی می باشد. ابتدا متغیر شروع را روی z در پشته قرار می دهیم: $d(q_0, l, z) = \{(q_1, Sz)\}$

شبیه سازی قوانین $S \rightarrow aSA \mid a$: $d(q_1, a, S) = \{(q_1, SA), (q_1, l)\}$

شبیه سازی قانون $A \rightarrow bB$: $d(q_1, b, A) = \{(q_1, B)\}$

شبیه سازی قانون $B \rightarrow b$: $d(q_1, b, B) = \{(q_1, l)\}$

با وقوع z در ته پشته، کار اشتقاق تمام شده و pda بوسیله دستور زیر در حالت پایانی قرار می گیرد:

$d(q_1, l, z) = \{(q_2, l)\}$

مثال: pda ای بسازید که زبان تولید شده بوسیله گرامر زیر را پذیرش کند.

$S \rightarrow aA$
$A \rightarrow aABC \mid bB \mid a$
$B \rightarrow b$
$C \rightarrow c$

حل:

$d(q_0, l, z) = \{(q_1, Sz)\}$
$d(q_1, l, z) = \{(q_f, z)\}$
$d(q_1, a, S) = \{(q_1, A)\}$
$d(q_1, a, A) = \{(q_1, ABC), (q_1, l)\}$
$d(q_1, b, A) = \{(q_1, B)\}$
$d(q_1, b, B) = \{(q_1, l)\}$
$d(q_1, c, C) = \{(q_1, l)\}$

قضیه: برای همه زبانهای مستقل از متن L ، یک npda به نام M وجود دارد بطوریکه $L = L(M)$.

قضیه: اگر در یک npda مفروض M داشته باشیم L=L(M)، آنگاه L یک زبان مستقل از متن است.

مثال: npda ای بسازید که زبان تولید شده بوسیله گرامر $S \to aSbb \mid aab.$ را پذیرش کند.

حل: می توان از زبان مربوطه $\{a^{n+2}b^{2n+1} : n \geq 0\}$ استفاده کرد:

$$d(q_0, a, z) = \{(q_1, z)\},$$
$$d(q_1, a, z) = \{(q_2, z)\},$$
$$d(q_2, a, z) = \{(q_2, 11z)\},$$
$$d(q_2, a, 1) = \{(q_2, 111)\},$$
$$d(q_2, b, 1) = \{(q_3, 1)\},$$
$$d(q_3, b, 1) = \{(q_3, l)\},$$
$$d(q_3, l, z) = \{(q_f, z)\}.$$

مثال: npda ای بسازید که زبان تولید شده بوسیله گرامر $S \to aSSS \mid ab$ را پذیرش کند.

حل: ابتدا گرامر را به فرم نرمال گریباخ تبدیل می کنیم:

$$S \to aSSS \mid aB$$
$$B \to b$$

سسپس اتومات را می سازیم:

$$d(q_0, l, z) = \{(q_1, Sz)\},$$
$$d(q_1, a, S) = \{(q_1, SSS), (q_1, B)\},$$
$$d(q_1, b, B) = \{(q_1, l)\},$$
$$d(q_1, l, z) = \{(q_f, z)\}.$$

مثال: یک npda دو حالتی برای زبان $L = \{a^n b^{n+1} : n \geq 0\}$ پیدا کنید.

حل: ابتدا گرامر به فرم گریباخ برای این زبان را به صورت زیر بدست می آوریم:

$$S \to aSB \mid b$$
$$B \to b$$

سپس یک npda با سه حالت q_0, q_1, q_f ایجاد کرده و برای حذف حالت q_1 از یک سمبل ویژه پشته مانند z_1 برای نشانه گذاری آن استفاده می کنیم. جواب نهایی به صورت زیر می باشد:

$$d(q_0, l, z) = \{(q_0, Sz_1)\},$$
$$d(q_0, a, S) = \{(q_0, SB)\},$$
$$d(q_0, b, S) = \{(q_0, l)\},$$
$$d(q_0, b, B) = \{(q_0, l)\},$$
$$d(q_0, l, z_1) = \{(q_f, l)\}.$$

اتوماتای پشته ای معین

اتومات پشته ای $M = (Q, \sum, \Gamma, d, q_0, z, F)$ معین گفته می شود اگر هم در تعریف npda صدق کند و همچنین دارای محدودیت هایی به این شرح باشد که به ازای هر $a \in \sum \mathbf{U}\{l\}, b \in \Gamma$ و $q \in Q$ ، $M = (Q, \sum, d, q_0, F)$ ،

1- $d(q, a, b)$ حداکثر یک عضو داشته باشد.

2- اگر $d(q, l, b)$ تهی نباشد، آنگاه $d(q, c, b)$ باید به ازای هر $c \in \Sigma$ تهی باشد.

اولین شرط فوق صرفا مستلزم آن است که به ازای هر سمبل ورودی مفروض و هر عنصر بالای پشته، حداکثر یک حرکت قابل انجام باشد. بر اساس شرط دوم، چنانچه در یکی از پیکربندی های مفروض به یک انتقال l برخورد کنیم، آنگاه هیچ حرکتی برای جلو بردن و مصرف ورودی امکان پذیر نمی باشد.

زبان مستقل از متن معین

زبان L یک زبان مستقل از متن معین نامیده می شود اگر و تنها اگر یک dpda به نام M وجود داشته باشد که L=L(M).

- تمامی زبان های منظم، مستقل از متن معین می باشند. چون هر زبان منظمی را می توان با یک dfa پذیرفت و چنین dfa یک dpda با پشته استفاده نشده است.
- اجتماع یک زبان مستقل از متن معین با یک زبان منظم، مستقل از متن معین می باشد.
- زبان های مستقل از متن معین، هیچگاه ذاتا مبهم نیستند.

مثال: آیا اتومات پشته ای زیر معین است؟

$$d(q_0, a, a) = \{(q_0, aa)\}$$
$$d(q_0, b, a) = \{(q_0, ba)\}$$
$$d(q_0, a, b) = \{(q_0, ab)\}$$
$$d(q_0, b, b) = \{(q_0, bb)\}$$
$$d(q_0, a, z) = \{(q_0, az)\}$$
$$d(q_0, b, z) = \{(q_0, ba)\}$$
$$d(q_0, l, a) = \{(q_1, a)\}$$
$$d(q_0, l, b) = \{(q_1, b)\}$$
$$d(q_1, a, a) = \{(q_1, l)\}$$
$$d(q_1, b, b) = \{(q_1, l)\}$$
$$d(q_1, l, z) = \{(q_2, z)\}$$

حل: خیر- چون انتقالات زیر در شرط 2 تعریف اتوماتای پشته ای معین، صدق نمی کند:

$$d(q_0, a, a) = \{(q_0, aa)\},$$
$$d(q_0, l, a) = \{(q_1, a)\}$$

این npda پذیرنده زبان $L = \{ww^R : w \in \{a, b\}^+\}$ می باشد.■

- هیچ هم ارزی بین اتوماتای پشته ای معین و نامعین وجود ندارد. (بالعکس اتوماتای متناهی)

مثال: یک dpda برای زبان مستقل از متن معین $L=\{a^n b^n : n \geq 0\}$ بسازید.

حل: جواب dpda ی $M=(\{q_0,q_1,q_2\},\{a,b\},\{0,1\},d,q_0,0,\{q_0\})$ با انتقالات زیر می باشد:

$$d(q_0,a,0)=\{(q_1,10)\},$$
$$d(q_1,a,1)=\{(q_1,11)\},$$
$$d(q_1,b,1)=\{(q_2,l)\},$$
$$d(q_2,b,1)=\{(q_2,l)\},$$
$$d(q_2,l,0)=\{(q_0,l)\}.$$

مثال: آیا زبان مستقل ازمتن $L=\{a^n b^n : n \geq 0\} \cup \{a^n b^{2n} : n \geq 0\}$ ، معین است؟

حل: خیر- چون باید یک یا دو b را با هر یک از a ها تطابق داده و هرگاه که رشته ورودی عضو $a^n b^n$ یا $a^n b^{2n}$ باشد، آنرا بعنوان اولین انتخاب مطرح کند. اطلاعات موجود در ابتدای رشته چنان نیست که بتوان به کمک آن انتخاب خود را قطعی کرد. ■

مثال: آیا زبان مستقل از متن $L=\{a^n b^n : n \geq 1\} \cup \{a\}$ ، معین است؟

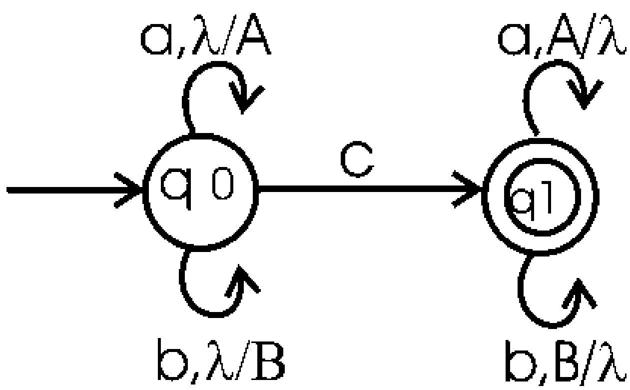
حل: بله- چون می توان یک dpda به صورت زیر برای آن ساخت. اگر اولین سـمبل ورودی a باشـد، ماشـین بـه یکـی از حـالات پایانی می رود و اگر سمبل بعدی وجود داشته باشد، از این حالت خارج شده و سپس $a^n b^n$ را می پذیرد:

$$d(q_0,a,z)=\{(q_3,1z)\},$$
$$d(q_3,a,1)=\{(q_1,11)\},$$
$$d(q_1,a,1)=\{(q_1,11)\},$$
$$d(q_1,b,1)=\{(q_1,l)\},$$
$$d(q_1,l,z)=\{(q_2,z)\}.$$

حالات پایانی q_2 و q_3 می باشد. ■

مثال: آیا زبان $\{wcw^R \mid w \in \{a,b\}^*\}$ معین است؟

حل: بله- چون می توان یک dpda به صورت زیر برای آن ساخت:



در این ماشین با رسیدن به نماد c مشخص می‌شود که به وسط رشته رسیده‌ایم و از حالت ذخیره به حالت تطابق می رویم.

تذکر: در زبان $\{ww^R \mid w \in \{a,b\}^*\}$، وسط رشته باید با آزمایش و خطا حدس زده شود و معین نمی باشد.■

مجموعه تست

۱- زبان های زیر مفروضند. کدام عبارت صحیح است؟

$L_1 = \{w_1 w_2 \mid w_1, w_2 \in (a+b)^*, |w_1| = |w_2|, w_2 \neq w_1^R\}$

$L_2 = \{a^n w w^R b^n \mid w \in (a+b)^*\}$

1) L_2 مستقل از متن و L_1 مستقل از متن نیست.
2) L_1 و L_2 مستقل از متن اند.
3) L_1 مستقل از متن و L_2 مستقل از متن نیست
4) هیچ یک از L_1 و L_2 مستقل از متن نیستند.

۲- زبان های منظم L_1, L_2, L_3 و L_4 مفروضند. برای چند زبان از این ۴ زبان می توان ماشین پشته ای (PDA) با حداکثر ۲ حالت ساخت؟

$L_1 = L(a^*)$
$L_2 = L((a+b)^*)$

$L_3 = \{w \in (a+b)^* \mid$ **تعداد b های w زوج باشد.** $\}$

$L_4 = \{w \in (a+b)^* \mid$ **تعداد b های w زوج و تعداد a های آن فرد باشد.** $\}$

1) 1
2) 2
3) 3
4) 4

۳- کدام گزاره در مورد گرامر زیر نادرست است؟

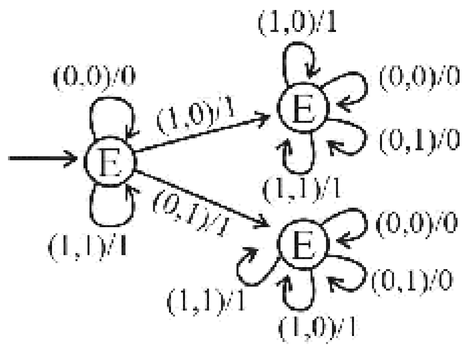
$G:$
$S \to 00S \mid X$
$X \to 11X \mid l$

1) گرامر G' با فرم نرمال چامسکی وجود دارد که زبان آن $L(G) - \{\lambda\}$ باشد.
2) G یک گرامر مستقل از متن نیست ولی زبان G مستقل از متن است.
3) یک PDA که زبانش با زبان G مساوی باشد وجود دارد.
4) G یک گرامر مستقل از متن است و زبان G هم مستقل از متن است.

۴- یک ماشین PDA با n حالت برای محاسبه کلیه ورودی ها حداکثر از 2^{n+1} سلول از حافظه پشته خود استفاده می کند. در این صورت کدام گزاره همواره درست نیست؟

1) برای زبان این ماشین یک گرامر منظم وجود دارد.
2) برای زبان این ماشین یک ماشین nondeterministic finite automaton بدون انتقال بلادرنگ وجود دارد.
3) برای زبان این ماشین یک ماشین nondeterministic finite automaton با انتقال بلادرنگ وجود دارد.
4) زبان این ماشین deterministic context free نیست.

۵- به فرض x و y دو عدد صحیح دودویی مثبت باشند کــه بــه صــورت $y = b_1b_2...b_n , x = a_1a_2...a_n$ نمــایش داده شوند، خروجی تراگذر زیر چیست؟



1) $\max(x,y)$ 2) $\min(x,y)$ 3) x 4) y

۶- کدام یک از زبانهای زیر منظم می باشند؟

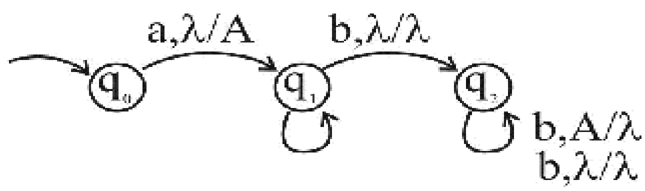
1) $L = \{a^n b^1 : n \le 1\}$

2) $L = \{c^i d^j \mid i \ge 0, j \ge 0\} \mathbf{U} \{c^i \mid i \ge 0\}$

3) $L = \{a^n b^1 a^k : n = 1 \text{ or } 1 \ne k\}$

4) $L = \{a^n b^k : n > k\} \mathbf{U} \{a^n b^k : n \ne k-1\}$

۷- در یک ماشین پشته ای با دیاگرام حالت روبرو کدام زبان پذیرفته می شود؟



$a, l / A$

1) $L = \{a^i b^{i+1}; i \ge 0\}$

2) $L = \{a^i b^j; 0 < i < j\}$

3) $L = \{(ab)^i ; i > 0\}$

4) $L = \{a^{i+1} b^i; i > 0\}$

۸- کدام یک از زبان های زیر مستقل از متن نیست؟

1) $L = \{uvwv^R : u, v, w \in \{a,b\}^+, |u| = |w| = 2\}$

2) $L = \{w_1 c w_2 : w_1, w_2 \in \{a,b\}^+, w_1 \ne w_2^R\}$ با فرض آنکه $\sum = \{a,b,c\}$ می‌باشد.

3) $L = \{a^n b^j a^k b^1 : n \le k, j \le 1\}$

4) $L = \{a^n b^m : n \ne m\}$

۹- کدام یک از زبان های زیر مستقل از متن نیست؟

1) $L = \{uvu^R v^R \mid u, v \in \{a,b\}^+\}$

2) $\Sigma = \{a,b\}\ ,\ L = \{a^n ww^R b^n \mid w \in \Sigma^*\ ,\ n \ge 1\}$

3) $\Sigma = \{a,b,c\}\ ,\ L = \{w_1 c w_2 \mid w_1, w_2 \in \{a,b\}^+\ ,\ w_1 \ne w_2^R\}$

4) $L = \{uvwv^R \mid u,v,w \in \{a,b\}^+, |u| = |w| = 2\}$

۱۰- کدام یک از زبان های زیر مستقل از متن است؟

1) $L = \{a^{2^n} : 3n = k\}$

2) $L = \{w \in \{a,b,c\}^* : n_a(w) = n_b(w) = n_c(w)\}$

3) $L = \{a^n b^n c^n : n \ge 0\}$

4) $L = \{a^n \mid n \ge 100$ or عدد اول $n\}$

۱۱- کدام یک از عبارت زیر صحیح می باشد؟

1) اگر G یک گرامر مستقل از متن باشد، آنگاه هر $w \in L(G)$ یک اشتقاق چپ و یک اشتقاق راست دارد.

2) با فرض $L = \{a^n b^n : n \ge 0\}$ ، L^2 مستقل از متن نمی باشد.

3) زبان $L = \{uvwv^R : u,v,w \in \{a,b\}^+, |u| = |w| = 2\}$ مستقل از متن نمی باشد.

4) زبان $L = \{w_1 c w_2 : w_1, w_2 \in \{a,b\}^+\ , w_1 \ne w_2^R\}$ با $\sum = \{a,b,c\}$ مستقل از متن نمی باشد.

۱۲- کدام یک از زبان های زیر مستقل از متن است؟

1) $L = \{a^{n!} : n \ge 0\}$

2) $L = \{a^n b^j : n = j^2\}$

3) $L = \{a^n b^n c^n : n \ge 0\}$

4) $L = \{a^{n^2} : n \ge 0\}$

۱۳- چه زبانی توسط NPDA زیر با حرکت های داده شده پذیرفته می شود؟

$M = (\{q_0, q_1, q_2\}, \{a,b\}, \{a,b,z\}, \delta, q_0, z, \{q_2\})$

$\delta(q_0, a, z) = \{(q_1, a), (q_2, \lambda)\}$
$\delta(q_1, b, a) = \{(q_1, b)\}$
$\delta(q_1, b, b) = \{(q_1, b)\}$
$\delta(q_1, a, b) = \{(q_2, \lambda)\}$

1) $L = \{a^n ba : n \ge 1\} \cup \{b\}$

2) $L = \{ab^n a : n \ge 1\}$

3) $L = \{ab^n a : n \ge 1\} \cup \{a\}$

4) $L = \{ab^n a : n \ge 0\}$

پاسخ تشریحی

۱–۲) رشته های هر دو زبان داده شده را می توان به کمک یک پشته بررسی کرد، بنابراین هر دو زبان مستقل از متن می باشند.

۲–۴) هر چهار زبان داده شده، منظم هستند، بنابراین مستقل از متن نیز می باشند. از آنجا که برای هر زبان مستقل از متن یک PDA با حداکثر دو حالت وجود دارد، گزینه 4 صحیح می باشد.

۳–۲) گرامر داده شده مستقل از متن است، بنابراین گزینه 2 نادرست است.

گزینه 1 درست است، چون برای هر گرامر مستقل از متن که شامل l نباشد، یک گرامر معادل در فرم نرمال چامسکی وجود دارد.

۴–۴) ماشینPDA ای با n حالت که برای محاسبه کلیه ورودی ها حداکثر از 2^{n+1} سلول از حافظه پشته خود استفاده می کند، در واقع نوعی ماشین متناهی می باشد که پذیرنده زبانهای منظم است که می تواند قطعی یا غیر قطعی باشد.

۵–۱) ماشین داده شده max(x,y)را مشخص می کند.

۶–۲) زبانهای گزینه های 1و3و4 مستقل از متن می باشند. زبان گزینه 2 را می توان به صورت $c^*d^* \cup c^* = c^*d^*$ نیز نشان داد.

۷–۲) ماشین داده شده زبان $L = \{a^i b^j ; 0 < i < j\}$ را می پذیرد.

۸–۳) رشته های زبان زیر را نمی‌توان توسط یک پشته تشخیص داد:

$L = \{a^n b^j a^k b^1 : n \le k, j \le 1\}$

۹–۱) رشته های زبان زیر را نمی‌توان توسط یک پشته تشخیص داد:

$L = \{uvu^R v^R \mid u, v \in \{a,b\}^+\}$

۱۰–۴) رشته های زبان های زیر را نمی‌توان توسط یک پشته تشخیص داد:

$L = \{a^{2^n} : 3n = k\}$

$L = \{w \in \{a,b,c\}^* : n_a(w) = n_b(w) = n_c(w)\}$

$L = \{a^n b^n c^n : n \ge 0\}$

۱۱–۱) اگر G یک گرامر مستقل از متن باشد، آنگاه هر $w \in L(G)$ یک اشتقاق چپ و یک اشتقاق راست دارد.

۱۲–۲) زبان $L = \{a^n b^j : n = j^2\}$ مستقل از متن است، چون رشته های این زبان را می توان توسط یک پشته تشخیص داد.

۱۳–۳) زبان $L = \{ab^n a : n \ge 1\} \cup \{a\}$ توسط npda داده شده ، پذیرفته می شود.

فصل نهم: خواص زبان های مستقل از متن

در این فصل، بسته بودن تحت انواع عملگرها، الگوریتمهای مربوط به تصمیم گیری، ویژگی های اعضای یک خانواده و نتایج مهمی را بررسی می کنیم.

لم تزریق برای زبان های مستقل از متن

فرض کنید L یک زبان مستقل از متن نامتناهی باشد. آنگاه عدد صحیح و مثبت m وجود دارد، بطوریکه هر w متعلق به L با فرض $|w| \geq m$ را می توان به صورت $w = uvxyz$ با شرایط $|vxy| \leq m$ و $|vy| \geq 1$ چنان تجزیه کرد که به ازای هر $i = 0,1,2,...$ داشته باشیم: $uv^i xy^i z \in L$

تذکر: مطالب بیشتر در رابطه با لم تزریق در کتاب مرجع همین مولف" فرشید شیرافکن" آورده شده است.

تذکر: با اعمال قوانین لم تزریق روی زبان $L = \{a^n b^n \mid n > 0\}$، متوجه می‌شویم که به ازای هر مقدار i ، رشته تزریق شده در L است. از این موضوع نمی‌توان نتیجه گرفت که L مستقل از متن است و فقط می‌توان گفت که از لم تزریق نتوانستیم نتیجه‌ای بگیریم. برای اثبات مستقل از متن بودن یک زبان باید از طریق ساختن یک گرامر مستقل از متن بهره گرفت.

مثال: توسط لم تزریق می‌توان نشان داد که زبانهای زیر مستقل از متن نمی‌باشند:

$\{ww : w \in \{a,b\}^*\}$, $\{a^{n!} \mid n > 0\}$, $\{a^n b^j : n = j^2\}$

$\{ww^R w : w \in \{a,b\}^*\}$, $\{a^{n^2} : n \geq 0\}$, $\{a^n b^j : n \leq j^2\}$

$\{a^n b^m \mid n \geq (m-1)^3\}$, $\{a^n b^m c^k \mid k = mn\}$, $\{a^n b^n c^m , n \neq m\}$

$\{a^n b^m c^k \mid k > n, k > m\}$, $\{a^n b^m c^k \mid n < m, n \leq k \leq m\}$, $\{w \mid n_a(w) < n_b(w) < n_c(w)\}$

$\{w \mid n_a(w) / n_b(w) = n_c(w)\}$, $\{w \mid n_a(w) = n_b(w) = n_c(w)\}$, $\{a^n b^j a^n b^j : n \geq 0, j \geq 0\}$

$\{a^n b^m :$ هر دو اول هستند m,n $\}$, $\{a^{nm} :$ هر دو اول هستند m,n $\}$

$\{a^n b^m :$ n یااول استmاول است $\}$, $\{a^n b^m :$ n واول استmاول نیست $\}$

$\{a^n :$ یک عدد اول است $n\}$, $\{w \in \{a,b,c\}^* : n_a^2(w) + n_b^2(w) = n_c^2(w)\}$

بسته بودن زبان های مستقل از متن

چند قضیه:

1- خانواده زبان های مستقل از متن تحت اجتماع، الحاق و بستار ستاره ای بسته است.

2- خانواده زبان های مستقل از متن تحت اشتراک و مکمل گیری بسته نیست.

3- اگر L_1 مستقل از متن و L_2 یک زبان منظم باشد، آنگاه $L_1 \cap L_2$ مستقل از متن است. این خاصیت، بسته بودن تحت اشتراک منظم خوانده می شود.

مثال: نشان دهید که زبان $L = \{a^n b^n : n \geq 0, n \neq 3\}$ مستقل از متن است.

حل: فرض کنید $L_1 = \{a^3b^3\}$، آنگاه چون این زبان متناهی است، بنابراین منظم نیز هست و چون زبان های منظم تحت مکمل گیری بسته هستند، $\overline{L_1}$ نیز منظم است. از طرفی زبان L را به صورت زیر نشان می دهیم و بر اساس بسته بودن زبان های مستقل از متن تحت اشتراک منظم، زبان L مستقل از متن می باشد.

$L = \{a^nb^n : n \geq 0\} \cap \overline{L_1}$

■

مثال: آیا زبان $L = \{w \in \{a,b,c\}^* : n_a(w) = n_b(w) = n_c(w)\}$ مستقل از متن است؟

حل:خیر- اشتراک این زبان با زبان منظم $L(a^*b^*c^*)$ برابر $\{a^nb^nc^n : n \geq 0\}$ می باشد که می دانیم مستقل از متن نمی باشد. بنابراین L مستقل از متن نیست. (با توجه به قضیه اشتراک منظم)■

مثال: آیا مکمل زبان $L = \{w \in \{a,b,c\}^* : n_a(w) = n_b(w) = n_c(w)\}$ مستقل از متن است؟

حل: بله – مکمل زبان L از دو حالت تشکیل شده است:1- $n_a(w) \neq n_b(w)$ -2 $n_a(w) \neq n_c(w)$

حالت 1 را می توان به دو حالت $n_a(w) < n_b(w)$ و $n_a(w) > n_b(w)$ تقسیم کرد.

حالت 2 را می توان به دو حالت $n_a(w) < n_c(w)$ و $n_a(w) > n_c(w)$ تقسیم کرد.

تمامی این چهار حالت مستقل از متن است و بنابراین اجتماع این ها نیز مستقل از متن است. (زبان های مستقل از متن تحت اجتماع بسته هستند.)■

مثال : آیا اشتراک L_1 و L_2 مستقل از متن است؟

$L_1 = \{a^nb^nc^m : n \geq 0, m \geq 0\}$, $L_2 = \{a^nb^mc^m : n \geq 0, m \geq 0\}$

حل: خیر. اشتراک این دو زبان برابر است با $\{a^nb^nc^n : n \geq 0\}$ که با لم تزریق نشان دادیم که مستقل از متن نیست. بنابراین زبان های مستقل از متن تحت اشتراک بسته نیست.■

نکات :

1- خانواده زبان های مستقل از متن تحت معکوس و هم ریختی بسته است.

2-خانواده زبان های مستقل از متن تحت تفاضل بسته نیست، اما تحت تفاضل منظم بسته است. یعنی اگر L_1 مستقل از متن و L_2 منظم باشد، آنگاه $L_1 - L_2$ مستقل از متن است.

3- خانواده زبان های مستقل از متن معین، تحت تفاضل منظم بسته است.

4- خانواده زبان های مستقل از متن معین، تحت اجتماع و اشتراک بسته نیست.

5- خانواده زبان های مستقل از متن غیرمبهم، تحت اجتماع و اشتراک بسته نیست.

6- خانواده زبان های خطی تحت اجتماع و هم ریختی بسته است.

7- خانواده زبان های خطی تحت اشتراک و الحاق بسته نیست.

8- اگر L_1 خطی و L_2 منظم باشد، آنگاه L_1L_2 یک زبان خطی است.

با فرض اینکه G یک گرامر مستقل از متن باشد:

1- یک الگوریتم برای تصمیم گیری در مورد اینکه $w \in L(G)$ هست یا نه، وجود دارد؟ (الگوریتم عضویت)

2- یک الگوریتم برای تصمیم گیری در مورد تهی بودن یا نبودن $L(G)$ وجود دارد.

3-یک الگوریتم برای تعیین نامتناهی بودن یا نبودن $L(G)$ وجود دارد.

به کمک هیچ الگوریتمی نمی توان تعیین کرد که آیا دو گرامر مستقل از متن مفروض یک زبان یکسان را تولید می کنند یا خیر.

می توان بوسیله الگوریتمی، درستی یا نادرستی $l \in L(G)$ را تعیین کرد. (G :گرامر مستقل ازمتن)

مجموعه تست

۱- ثابت Pumping Lemma برای زبان های مستقل از متن با گرامر G=(S,V,T,P) کدام است؟

1) تعداد واژه های زبان در T (Terminals)

2) تعداد واژه های نحوی در V (Nonterminals)

3) تعداد قواعد تولید در P (Production rules)

4) هیچکدام

۲- $L = \{a^m cb^n : m \neq n\} \cup \{a^m db^{2m} : m \geq 0\}$ **کدام گزینه نادرست است؟**

1) هر هومورفیسم L با یک PDA معین شناسائی می شود.

2) یک گرامر غیرمبهم برای زبان L موجود است.

3) یک PDA نامعین برای شناسائی L موجود است.

4) همه موارد

۳- اگر $L = \{0^n 1^n 0^n / n \in N\}$ **آنگاه:**

1) L^c مستقل از متن است.

2) L^c منظم است.

3) L مستقل از متن است.

4) L^c و L مستقل از متن هستند.

۴- همه زبان های زیر مستقل از متن هستند به جز؟

1) $L = \{a^n b^n c^m \mid n \geq 0, m \geq 0\} \cap \{a^{2n} b^{2n} c^{2m} \mid n \geq 0, m \geq 0\}$

2) $L = \{a^n b^{2n} c^m \mid n \geq 0, m \geq 0\} \cap \{a^n b^n c^{2m} \mid n \geq 0, m \geq 0\}$

3) $L = \{a^{2n} b^n c^m \mid n \geq 0, m \geq 0\}$

4) $L = \{a^{2m} b^n c^m \mid n \geq 0, m \geq 0\}$

۵- کدام یک از گزینه های زیر نادرست است؟

1) $L = \{w \in \{a,b\}^* : n_a(w) \neq n_b(w)\}$، یک زبان مستقل از متن معین نیست.

2) هر زبان منظم، یک زبان مستقل از متن معین است.

3) اگر L_1 زبان مستقل از متن معین و L_2 یک زبان منظم باشد، آنگاه $L_1 \cup L_2$ یک زبان مستقل از متن معین است.

4) اگر L_1 زبان مستقل از متن معین و L_2 یک زبان منظم باشد، آنگاه $L_1 \cap L_2$ یک زبان مستقل از متن معین است.

۶- کدام یک از عبارات زیر صحیح نمی باشد؟

1) اگر گرامر مستقل از متن $G=(V,T,S,P)$ را داشته باشیم، الگوریتمی برای تصمیم گیری در مورد اینکه $L(G)$ تهی است یا نه وجود دارد.

2) خانواده زبان های مستقل از متن تحت همریختی بسته است.

3) خانواده زبان های خطی تحت همریختی بسته است.

4) اگر گرامر مستقل از متن $G=(V,T,S,P)$ را داشته باشیم، الگوریتمی برای تعیین اینکه $L(G)$ متناهی است یا نه وجود ندارد.

۷- کدام یک از عبارات زیر صحیح نمی باشد؟

1) الگوریتمی وجود دارد که مشخص کند زبان تولید شده توسط یک گرامر مستقل از متن شامل کلماتی با طول کمتر از n می‌باشد.

2) خانواده زبان های مستقل از متن غیر گنگ، تحت اشتراک بسته نیست.

3) اگر L_1 خطی و L_2 منظم باشد، آنگاه L_1L_2 یک زبان خطی است.

4) با فرض آن که L_1 یک زبان مستقل از متن و L_2 یک زبان منظم می باشد، الگوریتمی برای تعیین اینکه آیا L_1 و L_2 عنصر مشترکی دارند یا نه، وجود ندارد.

پاسخ تشریحی

۱-۴) بر طبق لم پمپاژ، با فرض مستقل از متن بودن زبان L ، یک ثابت m وجود دارد بـه طـوری کـه اگـر w متعلـق بـه Lباشـد و $|w| \geq m$ و داشته باشیم $w = xyzuv$ به طوری که $|yu| \geq 1$ و $|yzu| \leq m$ آنگاه به ازای $i \geq 0$ داریم: $xy^i zu^i v \in L$.

بنابراین ثابت m در تعریف لم پمپاژ، هیچکدام از موارد گفته شده در گزینه ها نمی باشد.

۲-۱) گزینه 1 نادرست است، چون زبان داده شده، مستقل از متن نامعین است و هر همومورفیسم آن ممکن است مستقل از مـتن نامعین باشد. بنابراین هر همومورفیسم L با یک PDA معین شناسائی نمی شود.

گزینه 2 درست است، چون برای زبان داده شده می توان گرامر مستقل از متن غیر مبهم زیر را ارائه داد:

$$S \to A \mid B$$
$$A \to aAb \mid aA \mid Ab \mid ac \mid cb$$
$$B \to aBbb \mid d$$

گزینه 3 صحیح است. چون PDA مربوط به این زبان، نامعین است.

۳-۱) زبان $L = \{0^n 1^n 0^n / n \in N\}$ مستقل نمی باشد، اما متمم آن $L^c = \{0^n 1^m 0^k \mid n \neq m \neq k\}$ مستقل از متن است.

۴-۲) گزینه 1 را می توان به صورت $\{a^{2n} b^{2n} c^{2m} \mid n \geq 0, m \geq 0\}$ نوشت، که مستقل از متن می باشد.

گزینه 2 را می توان به صورت $\{a^n b^{2n} c^{4n} \mid n \geq 0\}$نوشت، که مستقل از متن نمی باشد.

گزینه های 3و4، مستقل از متن می باشند.

۵-۱) تمامی عبارات زیر صحیح می باشند:

الف- هر زبان منظم، یک زبان مستقل از متن معین است.

ب- اگر L_1 زبان مستقل از متن معین و L_2 یک زبان منظم باشد، آنگاه $L_1 \cup L_2$ یک زبان مستقل از متن معین است.

ج- اگر L_1 زبان مستقل از متن معین و L_2 یک زبان منظم باشد، آنگاه $L_1 \cap L_2$ یک زبان مستقل از متن معین است.

۶-۴) گزینه 4 نادرست است، چون الگوریتمی برای تعیین اینکه L(G) متناهی است یا نه وجود دارد.(G : گرامر مستقل از متن)

۷-۴) گزینه های 1و2و3 همگی صحیح هستند.

فصل دهم: ماشین های تورینگ

ماشین تورینگ کاملترین مدل برای یک ماشین محاسبه‌گر می باشد و از ماشـین‌هـای متنـاهی و پشـته‌ای کـاملتر اسـت. ماشـین تورینگ پذیرنده زبانهای منظم، مستقل از متن، وابسته به متن و بدون محدودیت می‌باشـد. در فصـل‌هـای قبـل مشـاهده شـد کـه ماشین‌های پشته‌ای قویتر از ماشین‌های متناهی می‌باشند و زبانهـای مسـتقل از مـتن را مـی‌پذیرنـد. امـا زبانهـای سـاده‌ای ماننـد $\{a^nb^nc^n\}$ یا $\{ww\}$ وجود دارند که توسط ماشین‌های پشته‌ای پذیرفته نمی‌شـوند و بایـد فراتـر از آنهـا قـدم برداشـت. وجـه تمـایز ماشین‌های متناهی و پشته‌ای در حافظه بود. ماشین‌های متناهی فاقد حافظه هستند و ماشین‌های پشته‌ای دارای حافظه‌ای از نـوع پشته می‌باشند. ماشین تورینگ یک کامپیوتر ساده است که دارای واحد پردازشی است که حافظه محدود دارد و دارای نواری اسـت که به عنوان حافظه ثانوی دارای ظرفیت نامحدود است و دستوراتی که این ماشین می‌تواند انجام دهد بسیار محدود اسـت.بر طبـق تز تورینگ، تمامی فرآیندهای محاسباتی، حتی آنها که بوسیله کامپیوترهای پیشرفته امروزی انجام می شوند، بوسیله ماشـین هـای تورینگ هم قابل انجام هستند.

ماشین تورینگ استاندارد

یک ماشین تورینگ، اتوماتی است که در آن از نوار به عنوان حافظه ذخیره سازی موقت استفاده شده است. این نوار به سلول هـایی تقسیم شده و هر سلول قادر به نگهداری فقط یک سمبل است. به این نوار یک هد خواندن-نوشتن متصل است کـه مـی توانـد بـه سمت راست یا چپ نوار حرکت کرده و در هر حرکت فقط یک سمبل را بخواند و بنویسد.

تعریف: ماشین تورینگ M یک هفت‌تایی است که به صورت $M=(Q,\sum,\Gamma,d,,q_0,F)$ تعریف می‌شود، که در آن :

Q: مجموعه حالات داخلی $\sum$: الفبای ورودی Γ :الفبای نوار δ : تابع انتقال

□: سمبل خالی(متعلق به الفبای نوار) q_0 : حالت شروع F : مجموعه حالت های پایانی

- **سمبل خالی به عنوان الفبای ورودی استفاده نمی شود. به عبارتی الفبای ورودی زیر مجموعه ای از الفبای نوار به استثنای فضای خالی است.**

- **تابع انتقال d به صورت** $(Q\times\Gamma\rightarrow Q\times\Gamma\times\{L,R\})$ **تعریف می شود.**

- **در ماشین متناهی حرکت نوار خوان فقط در یک جهت بود، اما در ماشین تورینگ نوارخوان در هر دو جهت چپ و راست می‌تواند حرکت کند.**

- وضعیت $d(q_0,a)=(q_1,d,R)$ یعنی ماشین در وضـعیت q_0 بـا خوانـدن a روی نـوار بـه وضـعیت q_1 رفتـه و a بـا d جایگزین می‌شود و نوک به سمت راست می‌رود.

مثال: عملکرد ماشین تورینگ زیر چیست؟

$$Q=\{q_0,q_1\}\quad \sum=\{a,b\}\quad \Gamma=\{a,b,B\}\quad F=\{q_1\}$$

$$d(q_0,a)=(q_0,b,R),\ d(q_0,b)=(q_0,b,R),\ d(q_0,B)=(q_1,B,L)$$

حل: این ماشین از حالت q_0 شروع به کار کرده و سمبل a، زیر هد خواندن– نوشتن قرار گرفته و یک b را جایگزین a کرده و هد به سمت راست حرکت می کند. ماشین در حالت q_0 باقی می ماند و a های بعدی نیز با b جایگزین می شوند، اما b ها تغییری نمی کنند. ماشین با رسیدن به اولین سلول خالی (blank)، یک سلول به عقب برمی‌گردد و در حالت نهایی q_1 متوقف می‌شود.

با فرض اینکه رشته aa بر روی نوار باشد، عملکرد این ماشین را می توان به صورت زیر نمایش داد:

$$q_0aa \;\mathbf{a}\; bq_0a \;\mathbf{a}\; bbq_0B \;\mathbf{a}\; bq_1b$$

مثال: عملکرد ماشین تورینگ زیر چیست؟

$$Q=\{q_0,q_1\}, \sum=\{a,b\}, \Gamma=\{a,b,\square\}, F=\{\ \}$$

$$\delta=(q_0,a)=(q_1,a,R), \delta(q_0,b)=(q_0,b,R), \delta(q_0,\square)=(q_1,\square,R)$$
$$,\delta=(q_1,a)=(q_0,a,l), \delta=(q_1,b)=(q_0,b,l), \delta(q_1,\square)=(q_0,\square,L)$$

حل: این ماشین تورینگ در یک حلقه بی‌نهایت می‌افتد و هرگز متوقف نمی‌شود. مثلا اگر نواری حاوی ...ab باشد و نوک روی a باشد، ماشین a را خوانده و به وضعیت q_1 می‌رود و a تغییر نمی‌کند و هد به سمت راست می‌رود، سپس b خوانده می‌شود و تغییری نمی‌کند و ماشین به وضعیت q_0 رفته و هد به سمت چپ حرکت می‌کند. یعنی دوباره به حالت اولیه رسیده‌ایم و این حرکات مجددا تکرار می‌شود. می‌توان وضعیت ماشین را روی baa یا هر ورودی دیگر نیز بررسی کرد. این وضعیت را به شکل $x_1qx_2 \overset{*}{\mathbf{a}} \infty$ نمایش می دهیم. فرم فوق به این معنی است که ماشین با راه اندازی از پیکربندی شروع x_1qx_2، وارد یک حلقه شده و هرگز متوقف نمی شود.

ویژگی های ماشین تورینگ استاندارد :

1- نامحدود بودن نوار ماشین از دو طرف (ممکن بودن حرکت به راست یا چپ به هر تعداد)

2- معین بودن (به ازای هر پیکربندی فقط یک حرکت تعریف می‌شود.)

3- عدم وجود هیچ فایل ورودی خاصی و همچنین عدم وجود هیچ وسیله خروجی خاصی.

ماشین های تورینگ در نقش پذیرنده های زبان

زبانی که ماشین تورینگ می‌پذیرد به صورت زیر تعریف می‌شود:

$$L(M)=\{w\in \sum{}^{+} : q_f \in F, x_1, x_2 \in \Gamma^* \qquad q_0w \overset{*}{\mathbf{a}} x_1q_fx_2\}$$

بر اساس این تعریف؛ ورودی w روی نوار نوشته شده و در هر یک از طرفین آن از سمبل فضای خالی استفاده می شود. اگر w عضو L(M) نباشد، یکی از دو حالت زیر اتفاق می افتد:

1- ماشین در یک حالت غیر پایانی متوقف می شود.

2- ماشین به یک حلقه بی نهایت وارد شده و هرگز متوقف نمی شود.

بنابراین، هر رشته ای که باعث توقف M نشود، عضو L(M) نمی باشد.

مثال: ماشین تورینگی برای پذیرش زبان $\{a^n b^n \mid n \geq 1\}$ ، طراحی کنید.

حل: در این ماشین با خواندن یک a به جای آن x قرار داده می‌شود و a های بعدی را رد می‌کند تا به یک b برسـد، سـپس آن را با y جایگزین کرده و دوباره به سمت چپ‌ترین a رفته و آنرا با x جایگزین می‌کند و دوباره چپ‌ترینb را با y جایگزین مـی‌کنـد و این حرکت پاندولی را ادامه می دهیم و تک تک a ها را با b نظیر آن تطابق می دهیم. اگر a یا b ای باقی نمانده بود، آنگـاه رشـته حتما عضو زبان خواهد بود.

جواب به صورت زیر می باشد (منظور از B ، سمبل خالی است)

$$\Sigma = \{a,b\}, \Gamma = \{a,b,x,y,B\}, F = \{q_4\}$$
$$d(q_0,a) = (q_1,x,R)$$
$$d(q_1,a) = (q_1,a,R)$$
$$d(q_1,y) = (q_1,y,R)$$
$$d(q_1,b) = (q_2,y,L)$$
$$d(q_2,y) = (q_2,y,L)$$
$$d(q_2,a) = (q_2,a,L)$$
$$d(q_2,x) = (q_0,x,R).$$
$$d(q_0,y) = (q_3,y,R)$$
$$d(q_3,y) = (q_3,y,R)$$
$$d(q_3,B) = (q_4,B,R)$$

بعنوان مثال، ورودی aabb پیکربندی های متوالی زیر را ایجاد می کند:

$$q_0aabb \;\mathbf{a}\; xq_1abb \;\mathbf{a}\; xaq_1bb \;\mathbf{a}\; xq_2ayb \;\mathbf{a}\; q_2xayb \;\mathbf{a}\; xq_0ayb \;\mathbf{a}\; xxq_1yb$$
$$\mathbf{a}\; xxyq_1b \;\mathbf{a}\; xxq_2yy \;\mathbf{a}\; xq_2xyy \;\mathbf{a}\; xxq_0yy \;\mathbf{a}\; xxyq_3y \;\mathbf{a}\; xxyyq_3B \;\mathbf{a}\; xxyyBq_4B.$$

مثال: ماشین تورینگی برای پذیرش زبان $L = \{a^n b^n c^n : n \geq 1\}$، طراحی کنید.

حل: حرف a را با x ، b را با y و c را با z علامت‌گذاری می‌کنیم.

$\delta(q_0,\not{b}) = (q_1,\not{b},R)$, $\delta(q_1,a) = (q_2,x,R)$, $\delta(q_2,a) = (q_2,a,R)$

$\delta(q_2,y) = (q_2,y,R)$, $\delta(q_2,b) = (q_3,y,R)$, $\delta(q_3,b) = (q_3,b,R)$

$\delta(q_3,z) = (q_3,z,R)$, $\delta(q_3,c) = (q_4,z,L)$, $\delta(q_4,a) = (q_4,a,L)$

$\delta(q_4,y) = (q_4,y,L)$, $\delta(q_4,b) = (q_4,b,L)$, $\delta(q_4,z) = (q_4,z,L)$

$\delta(q_4,x) = (q_4,x,R)$, $\delta(q_1,y) = (q_5,y,R)$, $\delta(q_5,\not{b}) = (q_6,\not{b},L)$

$\delta(q_5,z) = (q_5,z,R)$, $\delta(q_5,y) = (q_5,y,R)$, $\delta(q_6,x) = (q_6,a,L)$

$\delta(q_6,y) = (q_6,b,L)$, $\delta(q_6,z) = (q_6,z,L)$

مشاهده کردید که ماشین تورینگ قادر به شناخت زبان $L = \{a^n b^n c^n : n \geq 1\}$ که مستقل از متن نمی باشـد، بـود. ایـن نشـانه قوی بودن ماشین تورینگ از ماشین پشته ای می باشد.

مثال: یک ماشین تورینگ با حداکثر سه حالت طراحی کنید که زبان $L(a(a+b)^*)$ را روی الفبای $\{a,b\}$ پذیرش کند.

حل:

$$d(q_0,a)=(q_1,a,R)$$
$$d(q_1,a)=(q_1,a,R)$$
$$d(q_1,b)=(q_1,a,R)$$
$$d(q_1,B)=(q_2,B,R)$$

با ضابطه $F=\{q_2\}$.

مثال: یک ماشین تورینگ طراحی کنید که زبان $L(aba^*b)$ را روی الفبای $\{a,b\}$ پذیرش کند.

حل:

$$d(q_0,a)=(q_1,a,R)$$
$$d(q_1,a)=(q_2,b,R)$$
$$d(q_2,a)=(q_2,a,R)$$
$$d(q_2,a)=(q_3,b,R)$$

با ضابطه $F=\{q_3\}$.

مثال: یک ماشین تورینگ طراحی کنید که زبان $L=\{w:$ زوج است $|w|\}$ را روی الفبای $\{a,b\}$ پذیرش کند.

حل:

$$d(q_0,a)=(q_1,B,R)$$
$$d(q_0,b)=(q_1,B,R)$$
$$d(q_0,B)=(q_2,B,R)$$
$$d(q_1,a)=(q_0,B,R)$$
$$d(q_1,b)=(q_0,B,R)$$

با ضابطه $F=\{q_2\}$. ■

برای همه ماشین های تورینگ، یک ماشین دیگر با تنها یک حالت پایانی وجود دارد که همان زبان را می پذیرد. در واقع اگر چند حالت پایانی داشته باشیم، یک حالت پایانی جدید معرفی می کنیم و برای همه $q\in F$ و $a\in\Gamma$ انتقالات $d(q,a)=(q_f,a,R)$ را انجام می دهیم.

ماشین تورینگ به عنوان مترجم ها

ماشین تورینگ علاوه بر دارا بودن خاصیت پذیرش زبان‌ها، یک مدل ساده انتزاعی برای کامپیوترهای رقمی می‌باشد. در واقع تمامی توابع ریاضی معمولی توسط ماشین تورینگ، محاسبه پذیر بوده و میزان پیچیدگی آنها، تاثیری بر این امر نخواهد داشت. این موضوع در کتاب مرجع همین مولف شرح داده شده است.

مدل های دیگر ماشین های تورینگ

طبق تز تورینگ، پیچیده کردن ماشین های تورینگ استاندارد از طریق تجهیز آنها به ابزار ذخیره سازی پیچیده تر، تاثیری بر قدرت ماشین ندارد. چون هر نوع محاسبه ای که با این ماشین های جدید قابل انجام باشد، مدلی از محاسبه مکانیکی محسوب شده و به همین علت بوسیله یک مدل استاندارد هم قابل انجام است. انواع مدل های ماشین تورینگ عبارتند از:

۱- اعمال تغییرات جزئی در تعریف ماشین تورینگ (سکون دار، با نوار نیمه متناهی و آف لاین)

۲- ماشین های تورینگ با حافظه پیچیده تر(چند نواره و چند بعدی)

۳- ماشین های تورینگ نامعین

۴- ماشین تورینگ عمومی

۵- اتوماتای کراندار خطی

ماشین های تورینگ سکون دار

هد در این نوع ماشین ها می تواند پس از بازنویسی محتوای سلول، در جای خود باقی بماند و به جلو یا عقب حرکت نکند.

گنجاندن این انتخاب جدید برای حرکت هد، قدرت ماشین را افزایش نمی دهد.

قضیه: دسته ماشین های تورینگ سکون دار، هم ارز با دسته ماشین های تورینگ استاندارد می باشند.

اگر هر یک از سلول های نوار به چند بخش به نام شیار تقسیم شود به ماشین حاصل، ماشین تورینگ چند شیاری می گویند. علائم نوار این ماشین، چندتایی های الفبای ساده تر دیگری می باشند. این تغییر در تعریف ماشین تورینگ، آن را گسترش نمی دهد، چون فقط لازم است، Γ را الفبایی فرض کنیم که هر یک از سمبل های آن از چندین بخش تشکیل شده است.

ماشین های تورینگ با نوار نیمه نامتناهی

نوار در این ماشین فقط از یک طرف نامحدود است. وقتی هد در انتها قرار می گیرد، حرکت به چپ مجاز نیست.

این محدودیت هیچ تاثیری بر قدرت ماشین نمی گذارد.

ماشین های تورینگ آف لاین(off-line)

در ماشین تورینگ offline ، علاوه بر نوار شامل یک فایل ورودی فقط خواندنی نیز می‌باشد. در این نوع ماشین ها، تمامی حرکت ها توسط موارد زیر تصمیم گیری می شود:

الف- حالت درونی

ب- سمبلی که در حال حاضر از فایل ورودی خوانده می شود.

ج- آنچه که بوسیله هد خواندن-نوشتن مشاهده می شود.

یک ماشین تورینگ آف لاین با ویژگی های زیر متناظر با یک ماشین متناهی است:

1- ورودی فقط یک مرتبه قابل خواندن باشد.

2- ورودی از چپ به راست حرکت کند.

3- ورودی قابل بازنویسی نباشد.

4- حداکثر فقط از n سلول اضافی نوار، بعنوان فضای کاری استفاده کند.(n برای تمام ورودی ها ثابت است.)

ماشین های تورینگ با حافظه پیچیده تر

می توان ابزار ذخیره سازی ماشین تورینگ استاندارد را پیچیده تر کرد، اما این عمل قدرت ماشین را افزایش نمی دهد. با ذکر دو مثال(چند نواره و چند بعدی)، این موضوع را نشان می دهیم.

ماشین های تورینگ چند نواره

ماشین تورینگی با چند نوار که هر نوار، دارای هد خواندن- نوشتن می‌باشد که به طور مستقل کنترل می شود.

مثال: در یک ماشین 2 نواره، تابع انتقال $d(q_0,a,e)=(q_1,x,y,L,R)$، به این معنی است که ماشین در حالت q_0 بوده و اولین هد یک a و دومین هد یک e را می بیند. سپس سمبل روی اولین نوار با x جایگزین شده و هد به سمت چپ حرکت خواهد کرد. در عین حال، سمبل روی نوار دوم به y تغییر یافته و هد به سمت راست حرکت می کند. پس از این کار واحد کنترل به q_1 تغییر حالت می دهد.

مثال: روش طولانی و خسته کننده پذیرش زبان $\{a^nb^n\}$ را به کمک ماشین تورینگ استاندارد در فصل قبل مشاهده کردید. به کمک ماشین تورینگ دو نواره کار بسیار ساده تر می شود. در ابتدا رشته a^nb^n روی نوار 1 قرار دارد. سپس تمامی a ها را از نوار 1 خوانده و به نوار 2 کپی می کنیم. با رسیدن به اولین b روی نوار 1، آنها را با a های نوار 2 تطبیق می دهیم و به سادگی تعیین می کنیم آیا تعداد a ها و b ها برابر هستند یا خیر. (بنابراین بدون نیاز به جابجایی متوالی هد به راست و چپ، این عمل انجام شد.) ■

یک ماشین تورینگ چند نواره، قدرتمندتر از یک ماشین تورینگ استاندارد نیست.

ماشین های تورینگ چند بعدی

در این ماشین ها، نوار به صورت نامتناهی در بیش از یک بعد گسترش یافته است.

در یک ماشین تورینگ دو بعدی، هد علاوه بر حرکت به چپ و راست، می تواند به بالا و پایین نیز حرکت کند.

ماشین های تورینگ نامعین

ماشین تورینگ نامعین مشابه تورینگ معین است با این تفاوت که دارای تغییر وضعیت‌های متفاوتی می‌باشد.

مثال: ماشین تورینگ با انتقالاتی به فرم $d(q_0,a)=\{(q_1,b,R),(q_2,c,L)\}$ نامعین است.

■ قضیه : دسته ماشین های تورینگ معین و دسته ماشین های تورینگ نامعین هم ارز هستند.

- یک ماشین تورینگ نامعین به هیچ وجه قدرتمندتر از نوع معین خود نیست.
- هر ماشین تورینگ نامعین را می توان بوسیله یک ماشین تورینگ معین شبیه سازی کرد.
- وقتی که بیش از یک حرکت ممکن باشد، ماشین به تعداد لازم کپی از ماشین تهیه می‌کند و بـه هرکـدام کـاری را واگـذار می‌کند.
- یک آتاماتای پشته‌ای، مانند یک ماشین تورینگ نامعین است که نوار آن به صورت پشته استفاده می‌شود.

ماشین تورینگ عمومی

ماشین های تورینگ را نمی توان هم ارز با کامپیوترهای دیجیتال همه منظوره در نظر گرفت. با طراحی یک ماشین تورینـگ قابـل برنامه ریزی این موضوع را نقض می کنیم. ماشین تورینگ عمومی M_u، اتوماتی است که با در اختیار داشتن توصـیف هـر ماشـین تورینگ M بعنوان ورودی و رشته w ، قادر به شبیه سازی محاسبه M روی w می باشد.

آتاماتای کراندار خطی(LBA)

یک اتومات کراندار خطی(Linear Bounded Automata) ،یک ماشین تورینگ نامعین است، ولی با ایـن محـدودیت کـه مقدار نواری که می‌تواند استفاده کند تابعی از ورودی است. همچنین قسمت قابل استفاده نوار به سلول‌هایی که حاوی ورودی است محدود می‌باشند و برای حفاظت از این محدوده از دو علامت کروشه],[استفاده می‌شود.

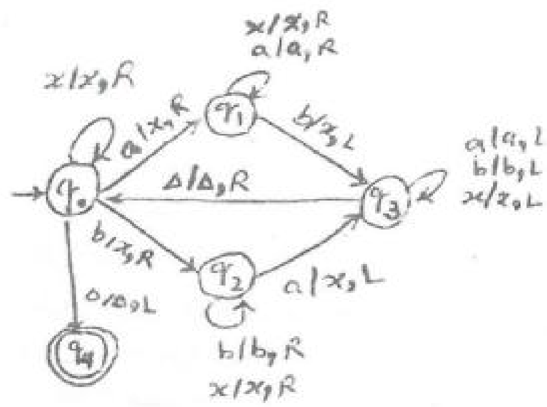
مثال: زبان $\{a^n b^n c^n \mid n \geq 1\}$ توسط یک LBA پذیرفته می‌شود. چون محاسباتی که برای پذیرش این زبان نیاز است، احتیـاجی به فضائی خارج از ورودی اولیه ندارد.

- اتوماتای کراندار خطی، قدرتمندتر از اتوماتای پشته ای هستند.
- هر زبان مستقل از متن فاقد l را می توان بوسیله یک LBA پذیرفت.
- هر چند LBA های معین هم وجود دارد، ولی هیچ روشی برای اثبات تناظر آنها با نسخه نامعین خود وجود ندارد.

مجموعه تست

۱- ماشین تورینگ مقابل چه زبانی را می پذیرد؟

(Δ نماد خنثی ماشین تورینگ است و مقصود از $n_a(w)$ تعداد a های موجود در w است.)



1) $\{a^n b^n : n \geq 0\}$

2) $\{w \in (a+b)^* : w = w^R\}$

3) $\{w \in (a+b+x)^* : n_a(w) = n_b(w)\}$

4) $\{ax : x \in (a+b)^*\} \mathbf{U} \{bx : x \in (a+b)^*\}$

۲- ماشین تورینگ M با دستورات زیر مفروض است که در آن q_0 حالت شروع، q_f حالت پایانی و B علامت خانه های خالی دو طرف نوار است. منظور از $d(q,a)=(P,X,R)$ این است که اگر M در حالت q و سر آن مقابل حـرف a روی نوار باشد، آنگاه به حالت P رفته و a را با X عوض کرده و سر را به اندازه یک خانه به راست می برد(اگر به جـای R ، L باشد آنگاه به چپ می رود.) اگر در شروع کار M (یعنی حالت q_0 و سر در ابتدای ورودی روی نوار) محتوی نـوار برابـر رشته aaabbb باشد، پس از دقیقا ۱۱ حرکت،محتوی کدام است؟

$$d(q_0,a)=(q_1,X,R)$$
$$d(q_1,a)=(q_1,a,R)$$
$$d(q_1,b)=(q_2,Y,L)$$
$$d(q_2,a)=(q_2,a,L)$$
$$d(q_2,X)=(q_1,X,R)$$
$$d(q_0,B)=(q_f,B,R)$$
$$d(q_1,Y)=(q_1,Y,R)$$
$$d(q_2,Y)=(q_2,Y,L)$$
$$d(q_1,B)=(q_f,B,R)$$

1) XaaYYb
2) XXaYYb
3) XXaYbb
4) XXXYYY

۳- برای تشخیص زبان $L=\{a^n b^n \mid n \ge 0\}$ یک ماشین تورینگ ساخته ایم. حداقل هزینه تشخیص $w \in L$ بـا ایـن ماشین تورینگ در چه حدی است؟

1) $o(n)$ 2) $o(n^2)$ 3) $O(n^3)$ 4) $o(2^n)$

۴- $\delta(q,a)=(q',X,L)$, $\delta(q,a)=(q',X.R)$ قواعد نمونه یک ماشین تورینگ می باشند که اگر ماشین در حالت q باشد و سر آن حرف a را روی نوار ببیند ماشین به حالت q' رفته ، حرف a با x عوض شده و سر ماشین به ترتیب به راست (R) و یا چپ (L) می رود. زبان ماشین تورینگ با قواعد زیر کدام است؟(q_4 حالت نهایی ، B علامت جای خالی روی نوار و $\Sigma=\{a,b\}$ مجموعه واژه های زبان است)

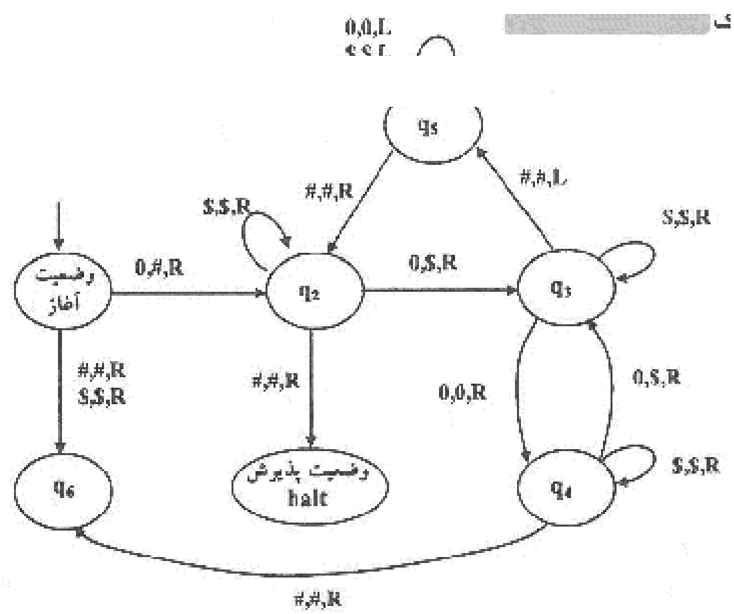
$\delta(q_\circ,a)=(q_1,X,R)$, $\delta(q_\circ,y)=(q_3,y,R),\delta(q_1,a)=(q_1,a,R),\delta(q_1,y)=(q_1,y,R),\delta(q_1,b)=(q_2,y,L),$

$\delta(q_2,a)=(q_2,a,L),\delta(q_2,y)=(q_2,y,L),\delta(q_2,x)=(q_0,x,R),\delta(q_3,y)=(q_3,y,R),\delta(q_3,B)=(q_4,B,R)$

1) $\{a^n b^n \mid n \ge 0\}$ 2) $\{a^n b^n a^n \mid n \ge 1\}$

3) $\{w \in (a+b)^* \mid$ تعداد a ها با تعداد b ها برابر است $\}$ 4) هیچکدام

۵- در ماشین تورینگ زیر نماد # نشان دهنده خانه خالی، و نماد $ یکی از حروف نوار ماشین است. منظور از انقال بـه شکل x,y,R این است که حرف y به جای x جایگزین شده و هد به سمت راست حرکت می کند. در انتقـال بـه شـکل x,y,L نیز حرف y به جای x جایگزین شده و هد به سمت چپ حرکت می کند. زبان ماشین تورینگ روبرو عبارت است:



ع)

1) $\{0^{2n} \mid x \ge 0\}$ 2) $\{0^{2n} \mid n \ge 0\}$ 3) $\{0^{n^2} \mid n \ge 0\}$ 4) $\{0^{n^n} \mid n \ge 0\}$

۶- فضای زبان هایی که با مدل تورینگ مشخص می شود، با کدام یک از تغییرات زیر در تعریف ماشین تورینگ تغییر خواهد کرد؟

1) امکان استفاده از انتقال بلادرنگ در اتوماتون ماشین

2) عدم امکان حرکت هد ماشین به سمت چپ

3) استفاده از بیش از یک نوار ولی یک طرفه

4) محدود کردن حروف الفبای ماشین به $\{B,0,1\}$

۷- کدام یک از موارد زیر صحیح نیست؟

1) قدرت یک ماشین متناهی قطعی و یک ماشین متناهی غیرقطعی برابر است.

2) قدرت یک ماشین پشته ای غیرقطعی، بیش از یک ماشین پشته ای قطعی است.

3) قدرت یک ماشین تورینگ غیرقطعی، بیش از یک ماشین تورینگ قطعی است.

4) اینکه قدرت یک ماشین کراندار خطی غیرقطعی بیش از یک ماشین کراندار خطی قطعی هست یا خیر، تصمیم ناپذیر است.

پاسخ تشریحی

۱-۳) ماشین تورینگ داده شده زبان زیر را می پذیرد:

$$\{w \in (a+b+x)^* : n_a(w) = n_b(w)\}$$

۲-۱) در زیر 11 حرکت آورده شده و در انتها XaaYYb روی نوار خواهد بود:

$$\underline{a}aabbb \to X\,\underline{a}abbb \to Xa\underline{a}bbb \to Xaa\underline{b}bb \to Xa\underline{a}Ybb \to X\,\underline{a}aYbb$$

$$\underline{X}aaYbb \to X\,\underline{a}aYbb \to Xa\underline{a}Ybb \to Xaa\underline{Y}bb \to Xaa\underline{Y}Yb$$

۳-۲) در متن درس نحوه تشخیص زبان $L = \{a^n b^n \mid n \geq 0\}$ توسط ماشین تورینگ آورده شده است. مشاهده کردید که هد ماشین به ازای دیدن هر a به ابتدای حروف b رفته و دوباره به ابتدای رشته برمی گردد. بنابراین حداقل هزینه تشخیص $w \in L$ با این ماشین در حد $o(n^2)$ است.

۴-۴) زبان این ماشین $\{a^n b^n \mid n \geq 1\}$ می باشد. (رشته l را نمی پذیرد.)

۵-۲) زبان های گزینه 1و3 شامل رشته λ می باشند در حالی که توسط ماشین پذیرفته نمی شود.

2-6) فضای زبان هایی که با مدل تورینگ مشخص می شود، با تغییر عدم امکان حرکت هد ماشین به سمت چپ در تعریف ماشین تورینگ، تغییر خواهد کرد.

۷-۳)یک ماشین تورینگ غیرقطعی، به هیچ وجه قدرتمندتر از نوع قطعی(معین) خود نیست.

فصل یازدهم: سلسله مراتب اتوماتا و زبان های صوری

در این فصل زبان های صوری را مطالعه می کنیم. در اولین قدم، زبان های مرتبط با ماشین های تورینگ و محدودیت های آنها را بررسی می کنیم. از آنجا که ماشین های تورینگ قادر به انجام انواع محاسبات الگوریتمی می باشند، خانواده زبان های مرتبط با آنها هم منطقا باید بسیار گسترده باشند. سئوالی که مطرح است این است که آیا زبانی وجود دارد که توسط هیچ ماشین تورینگی پذیرفته نشود؟ برای پاسخ به این پرسش نشان می دهیم که تعداد زبان ها، بیشتر از ماشین های تورینگ است، بنابراین به ازای برخی زبان ها، هیچ ماشین تورینگی وجود ندارد.

زبان های بازگشتی و شمارش پذیر بازگشتی

قبل از آشنایی با چند اصطلاح در مورد زبانهای مرتبط با ماشین تورینگ، باید بین زبان هایی که یک ماشین تورینگ پذیرنده به ازای آن وجود دارد و زبان هایی که یک الگوریتم عضویت به ازای آن وجود دارد، تمایز قائل شویم. بدلیل آنکه ماشین های تورینگ لزوما برای ورودی که آنرا نمی پذیرند، توقف نمی کنند، بنابراین وجود یک ماشین تورینگ پذیرنده بطور ضمنی به معنای وجود الگوریتم عضویت مربوطه نیست.

تعریف: زبان مفروض L بازگشتی شمارش پذیر خوانده می شود، اگر ماشین تورینگی وجود داشته باشد که آنرا پذیرش کند.

تعریف: زبان مفروض L روی Σ، بازگشتی خوانده می شود، اگر ماشین تورینگ M وجود داشته باشد که L را پذیرفته و روی هر w موجود در Σ^+ متوقف شود. به بیان دیگر، یک زبان بازگشتی خواهد بود اگر و تنها اگر یک الگوریتم عضویت به ازای آن وجود داشته باشد.

قضیه هایی در رابطه با زبان های بازگشتی و بازگشتی شمارش پذیر

1: اگر S یک مجموعه شمارش پذیر نامتناهی باشد، آنگاه مجموعه توانی آن، (2^S)شمارش پذیر نیست.

2 : به ازای هر Σ غیر تهی، زبان هایی وجود دارند که بازگشتی شمارش پذیر نیستند.

3 : یک زبان بازگشتی شمارش پذیر وجود دارد که مکمل آن بازگشتی شمارش پذیر نیست.

4 : اگر زبان L و مکمل $\overline{L}$ آن هر دو بازگشتی شمارش پذیر باشند، آنگاه هر دو زبان بازگشتی هستند. اگر L بازگشتی باشد، آنگاه $\overline{L}$هم بازگشتی است و در نتیجه هر دو بازگشتی شمارش پذیر هستند.

۵ : یک زبان بازگشتی شمارش پذیر وجود دارد که بازگشتی نیست. به بیان دیگر، خانواده زبان های بازگشتی یکی از زیر مجموعه های مناسب خانواده زبان های بازگشتی شمارش پذیر است.

تذکر: هر زبانی که بوسیله یک روش الگوریتمی مستقیم قابل توصیف باشد، بوسیله یک ماشین تورینگ هم قابل پذیرش بوده و بنابراین بازگشتی شمارش پذیر است. بنابراین برای توصیف زبانی که بازگشتی شمارش پذیر نباشد، باید از روش غیر مستقیم استفاده کرد. اما اینکار غیر ممکن است.

- مجموعه تمام زبان هایی که بازگشتی شمارش پذیر نیستند، قابل شمارش نمی باشند.
- اگر L یک زبان متناهی باشد، آنگاه L^+ بازگشتی شمارش پذیر است.
- اگر L یک زبان مستقل از متن باشد، آنگاه L^+ بازگشتی شمارش پذیر است.
- اگر L بازگشتی باشد، لزوما L^+هم بازگشتی است.
- خانواده زبان های بازگشتی، تحت اجتماع و اشتراک و معکوس بسته است.
- خانواده زبان های بازگشتی شمارش پذیر ، تحت اجتماع و معکوس بسته است.
- زبانهای بازگشتی شمارش پذیر ، تحت متمم بسته نمی‌باشند.
- اگر یک زبان بازگشتی شمارش پذیر نباشد، مکمل آن هم بازگشتی نیست.
- مکمل یک زبان مستقل از متن، حتما بازگشتی است.
- اگر L_1 بازگشتی و L_2 بازگشتی شمارش پذیر باشد، آنگاه $L_2 - L_1$ لزوما بازگشتی شمارش پذیر است.
- مکمل هر زبان تصمیم پذیر، تصمیم پذیر است.
- مکمل زبان بازگشتی شمارش پذیر، لزوما بازگشتی شمارش پذیر نیست.
- زبان های تصمیم پذیر، تحت اشتراک بسته هستند.
- اشتراک یک زبان بازگشتی شمارش پذیر با زبان تصمیم پذیر، لزوما تصمیم ناپذیر نیست.
- اشتراک یک زبان بازگشتی شمارش ناپذیر با زبان تصمیم پذیر، لزوما بازگشتی شمارش پذیر نیست.
- اشتراک یک زبان بازگشتی شمارش ناپذیر با زبان بازگشتی شمارش پذیر، لزوما بازگشتی شمارش پذیر نیست.

گرامرهای بدون محدودیت

گرامر مفروض $G=(V,T,S,P)$ بدون محدودیت خوانده می شود، اگر تمامی قوانین آن به فرم $u \to v$ باشند که در آن ، u عضو $(V \cup T)^+$ و v عضو $(V \cup T)^*$ می باشد.

در گرامرهای بدون محدودیت، اساسا هیچ شرط و محدودیتی برای قواعد تولید قائل نمی شویم. بعلاوه هر تعداد غیر پایانی و پایانی را می توان با هر ترتیبی در طرفین راست و چپ قرار داد. فقط یک شرط باید مورد توجه قرار گیرد: l نمی تواند در سمت چپ قواعد تولید رخ دهد.

- گرامرهای بدون محدودیت بسیار قدرتمندتر از فرم های محدود شده ای از قبیل گرامرهای منظم و مستقل از متن هستند.
- گرامرهای بدون محدودیت، متناظر با بزرگترین خانواده زبان ها بوده و بوسیله ابزار مکانیکی قابل تشخیص می باشند.
- گرامرهای بدون محدودیت، صرفا خانواده زبان های بازگشتی شمارش پذیر را ایجاد می کنند.
- تعیین $L(G)=\phi$ وقتی که G یک گرامر بدون محدودیت است، یک مسئله تصمیم ناپذیر است.

تعیین تهی بودن $L(G_1) \cap L(G_2)$، وقتی که G_1 گرامر بدون محدودیت و G_2 گرامر منظم است، یک مسئله تصمیم ناپذیر است.

قضیه هایی در رابطه با گرامرهای بدون محدودیت:

قضیه ۱: هر زبانی که بوسیله یک گرامر بدون محدودیت ایجاد شود، بازگشتی شمارش پذیر است.

قضیه ۲: برای هر زبان بازگشتی شمارش پذیر L، یک گرامر بدون محدودیت G وجود دارد، بطوریکه L=L(G).

گرامرهای حساس به متن

گرامر مفروض G=(V,T,S,P) حساس به متن خوانده می شود، اگر تمامی قوانین آن به فرم $x \to y$ باشند که در آن x و y عضو $(V \cup T)^+$ باشند و $|x| \le |y|$.

تذکر: طبق تعریف بالا، قاعده $x \to \lambda$ غیر مجاز است. بنابراین گرامرهای حساس به متن هرگز قادر به تولید زبانهای دارای رشته تهی نمی باشند.

زبان های حساس به متن و اتوماتای کراندار خطی

زبان مفروض L حساس به متن خوانده می شود، اگر گرامر حساس به متن G وجود داشته باشد، بطوریکه $L = L(G)$ یا $L = L(G) \cup \{\lambda\}$.

مثال: زبان $L = \{a^n b^n c^n : n \ge 1\}$ یک زبان حساس به متن است. برای اثبات از یک گرامر حساس به متن در زبان مذکور مانند گرامر زیر استفاده می کنیم:

$S \to abc \mid aAbc$
$Ab \to bA$
$Ac \to Bbcc$
$bB \to Bb$
$aB \to aa \mid aaA$

تذکر: چون زبان حساس به متن $\{a^n b^n c^n\}$، مستقل از متن نیست، می توان نتیجه گرفت که خانواده زبانهای مستقل از متن، یکی از زیر مجموعه های خانواده زبان های حساس به متن است.

قضیه هایی در رابطه با زبانهای حساس به متن

1: به ازای هر زبان L حساس به متن دارای λ، یک اتومات کراندار خطی M وجود دارد بطوریکه L=L(M).

2: اگر زبان L بوسیله یک اتومات کراندار خطی مفروض به نام M پذیرفته شود، آنگاه یک گرامر حساس به متن وجود دارد که L را تولید می کند.

3: تمامی زبانهای حساس به متن، بازگشتی هستند.

4: یک زبان بازگشتی وجود دارد که حساس به متن نمی باشد.

- اتوماتای کراندار خطی عملا ضعیف تر از ماشین های تورینگ بوده و فقط قادر به پذیرش یکی از زیر مجموعه هـای مناسـب زبان های بازگشتی می باشند.
- هر زبان پذیرفته شده بوسیله یک اتومات پشته ای، بوسیله یک اتومات کراندار خطی هم پذیرفته می شـود، امـا زبـان هـایی هم وجود دارند که بوسیله اتوماتای کراندار خطی پذیرفته می شوند، اما هیچ اتوماتای پشته ای به ازای آن وجود ندارد.
- هر زبان حساس به متنی بوسیله یک ماشین تورینگ پذیرفته شده و بنابراین، بازگشتی شمارش پذیر محسوب می شود.
- خانواده زبانهای حساس به متن تحت اجتماع و معکوس بسته هستند.

ارتباط بین زبان ها، گرامر ها و ماشین ها

جدول زیر ارتباط بین زبان ها، گرامرها و ماشین ها را نشان می دهد:

گرامر	زبان مربوط	ماشین معادل
منظم	منظم	متناهی(FA)
مستقل از متن	مستقل از متن	پشته ای (PDA)
حساس به متن	حساس به متن	کراندار خطی(LBA)
	بازگشتی	تورینگ تصمیم گیرنده
بدون محدودیت	بازگشتی شمارش پذیر	تورینگ تشخیص دهنده

تذکر: البته می توان گرامر خطی را نیز در جدول بالا گنجاند که زبان مربوط بـه آن خطـی یـا مسـتقل از مـتن قطعـی مـی باشـد. همچنین ماشین متناظر با زبان مستقل از متن قطعی، ماشین پشته ای قطعی(DPDA) می باشد.

سلسله مراتب چامسکی

نوام چامسکی زبان ها را در چهار گروه، از نوع صفر تا نوع سه، دسته بندی کرد.

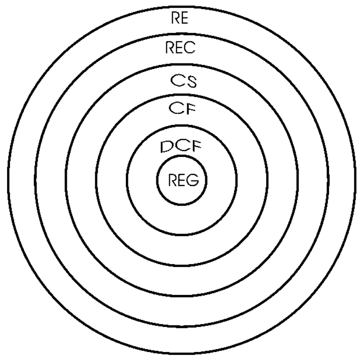
زبان های نوع صفر :بوسیله گرامرهای محدود نشده، یعنی زبان های بازگشتی شمارش پذیر، ایجاد می شوند.

زبان های نوع یک:شامل زبانهای حساس به متن می باشند.

زبان های نوع دو :شامل زبانهای مستقل از متن می باشند.

زبان های نوع سه :شامل زبانهای منظم می باشند.

هر یک از خانواده های زبان های نوع i ، یکی از زیر مجموعه های مناسب خانواده نوع i-1 محسوب مـی شـوند. نمـودار زیـر ایـن رابطه را مشخص می کند:



ارتباط بین زبانها را می‌توان به کمک رابطه زیر نمایش داد:

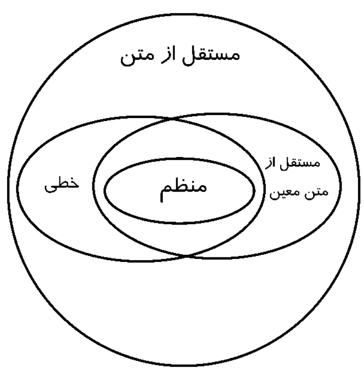
$REG \subseteq DCF \subseteq CF \subseteq CS \subseteq REC \subseteq RE$

بازگشتی شمارش پذیر $\subseteq$ بازگشتی $\subseteq$ حساس به متن $\subseteq$ مستقل از متن $\subseteq$ مستقل از متن قطعی $\subseteq$ منظم

- هر زبان منظمی، زبانی است مستقل از متن.(چون زبانهای منظم حالت خاصی از زبانهای مستقل از متن می‌باشند.)
- زبانهای بازگشتی شمارش پذیر ، دارای زیر مجموعه‌ای هستند به نام زبانهای بازگشتی.

ارتباط بین زبانهای منظم، خطی، مستقل از متن معین و مستقل از متن

در شکل قبلی جایگاه زبانهای خطی و مستقل از متن معین(قطعی) نشان داده نشده است. این موضوع در شکل زیر نشان داده شده است:



مثال : زبان مستقل از متن $\{w \mid n_a(w)=n_b(w)\}$ معین است ولی خطی نیست.

مثال: زبان $\{a^nb^n\}\cup\{a^nb^{2n}\}$ خطی است ولی معین نیست.

جدول زیر خواص بسته بودن شش نوع زبان را تحت عملگرهای مختلف نشان می دهد:

عملگر	نوع زبان					
	منظم	مستقل از متن قطعی	مستقل از متن	حساس به متن	بازگشتی	بازگشتی شمارش پذیر
الحاق	P	-	P	P	P	P
اجتماع	P	-	P	P	P	P
اشتراک	P	-	-	P	P	P
مکمل	P	P	-	P	P	-
معکوس	P	-	P	P	P	P
همریختی	P	-	P	-	-	P
بستار ستاره	P	-	P	P	P	P

تذکر: به غیر از عملگرهای موجود در جدول، عملگرهای اشتراک منظم، همریختی بدون l ، همریختی معکوس نیز وجود دارد:

1- همه انواع زبان های جدول بالا، تحت اشتراک منظم و تحت همریختی معکوس بسته هستند.

2- همه انواع زبان های جدول بالا(به غیر از مستقل از متن قطعی)، تحت همریختی بدون l بسته هستند.

مجموعه تست

۱- با تحمیل کدام شرط بر روی تعریف ماشین تورینگ، کلاس زبانهای مشخص شده با کلاس زبانهای r.e. تفاوت خواهد داشت؟

1) نوار ماشین از یک سمت محدود شود.

2) هد ماشین بعد از هر حرکت به سمت چپ اجبارا در مرحله بعدی باید یک بار به سمت راست حرکت کند.

3) تعداد حالتها و نمادهای ماشین حداکثر 1390 است.

4) تعداد حالتهای ماشین حداکثر 1390 و تعداد نمادهای ماشین حداقل 1390 باشد.

۲- یک ماشین تورینگ غیرقطعی که در واحد کنترل آن انتقالهای مستقل از محتوای نوار (λ – Transition) هم وجود دارد، داده شده است. کدام گزینه صحیح است؟

1) درخت محاسبه چنین ماشینی برای هر ورودی لزوما یک مسیر نیست.

2) درخت محاسبه چنین ماشینی لزوما یک شاخه محاسبه نامتناهی (loop) دارد.

3) امکان حذف λ - انتقالها در چنین ماشین تورینگی یک مسئله تصمیم پذیر است.

4) درخت محاسبه چنین ماشینی لزوما برای هر ورودی، نامتناهی شاخه محاسبه دارد.

۳- مجموعه زبان های بازگشتی(Recursive) را R و مجموعه زبان های بازگشتی شمارش پذیر (Recursively Enumerable) راRE می نامیم. زبان L مفروض است. در کدام یک از حالت های زیر یک ماشین تورینگ که برای تمام رشته های L به حالت توقف برسد وجود دارد؟

1) $L \in RE$ و $\overline{L} \in RE$

2) $L \in RE$ و $\overline{L} \notin R$

3) $L \in RE$ و $\overline{L} \in RE$

4) هیچ کدام

۴- زبان L مجموعه تمامی زوج های مرتب <M,w> است که در آن M یک ماشین تورینگ و w یک رشته است به طوری که ماشین M بر ورودی w متوقف نمی شود. کدام یک از جملات زیر صحیح است؟

الف) L بازگشتی است. ب) L به طور بازگشتی شمارا است.

ج) L بازگشتی نیست. د) L به طور بازگشتی شمارا نیست.

1) ب
2) الف و ب
3) ب و ج
4) ج و د

۵- زبان $L = \{x^i \ y^j \ z^{j+2} \ w^k \ v^{i+k} \mid i, j, k \geq 0\}$ با تعریف زیر مفروض است. کدام یک از گزاره ها نادرست است؟

1) یک آتاماتای پشته ای غیرقطعی مثل A وجود دارد به قسمی که L=L(A).

2) رشته های L توسط یک اتاماتای قطعی کراندار (Linedar Bounded Automata) قابل شناسائی هستند.

3) زبان L از نوع مستقل از متن معین (DCFL) نمی باشد.

4) زبان L از نوع بازگشتی شمارش پذیر است.

۶- کدام گزاره در مورد زبان $L = \{a^i b^j c^k \mid i, j, k \geq 1, k \leq \max(i, j)\}$ نادرست است؟

1) L یک زبان مستقل از متن است.

2) L یک زبان حساس به متن است.

3) L مستقل از متن نیست ولی حساس به متن است.

4) L زبان یک اتوماتون قطعی نامتناهی روی $\{a,b,c\}^*$ است.

۷- اگر N مجموعه اعداد طبیعی باشد، کدام گزاره نادرست است؟

1) تعداد زیر مجموعه های شمارای بازگشتی شمارش پذیر (R.E)، شمارا است.

2) تعداد زیر مجموعه های شمارای N ، شمارا است.

3) مجموعه اعداد اول تصمیم پذیر است.

4) هر مجموعه تصمیم پذیر، بازگشتی شمارش پذیر(R.E) است.

۸- کدام گزاره درست است؟

1) هر مجموعه تصمیم پذیر، توسط یک PDA پذیرفته می شود.

2) هر مجموعه بازگشتی شمارش پذیر (R.E)، توسط یک DFA پذیرفته می شود.

3) ماشین تورینگی وجود دارد که به ازاء هر ورودی خروجی ندارد و به ازاء هیچ ورودی نیز در loop نمی افتد.

4) هر زبانی که توسط یک ماشین تورینگ پذیرفته می شود، تصمیم پذیر است.

۹- گرامرهای حساس به متن (context sensitive) معادل چه نوع مدلی هستند؟

1) ماشین تورینگ (Turing machine)

2) اتوماتون های کراندار خطی (linear bounded autoamta)

3) اتوماتون های پشته ای (push down autoamta)

4) اتوماتون های قطعی نامتناهی (delerministic finite autoamta)

۱۰- کدام یک از گزاره های زیر صحیح است؟

1) مجموعه همه ماشین های تورینگ (Turing Machines) روی یک الفبا، شمارش پذیر (countable) است.

2) مجموعه همه ماشین های تورینگ (Turing Machines) روی یک الفبا، شمارش ناپذیر (uncountable) است.

3) مجموعه همه زبان های نامنظم (non-regular) روی یک الفبا، شمارش پذیر (countable) است.

4) مجموعه تمامی رشته های تعریف شده روی یک الفبا، شمارش ناپذیر (uncountable) است.

پاسخ تشریحی

۱-۳) با تحمیل شرط "تعداد حالتها و نمادهای ماشین حداکثر 1390 است" ، بـر روی تعریـف ماشـین تورینـگ، کـلاس زبانهـای مشخص شده با کلاس زبانهای .r.e تفاوت خواهد داشت.

۲-۳) امکان حذف λ - انتقالها در یک ماشین تورینگ غیرقطعی که در واحد کنتـرل آن انتقالهـای مسـتقل از محتـوای نـوار هـم وجود دارد، یک مسئله تصمیم پذیر است.

۳-۳) با توجه به تعاریف زیر، گزینه 3 درست است:

الف- اگر زبان L و مکمل آن یعنی $\overline{L}$ هر دو بازگشتی شمارش پذیر باشند، آنگاه هر دو بازگشتی هستند.

ب- زبان مفروض L روی Σ، بازگشتی خوانده می شود، اگر ماشین تورینگ M وجـود داشـته باشـد کـه L را پذیرفتـه و روی هـر wموجود در Σ^+ متوقف شود.

۴-۳) زبان مفروض L روی Σ بازگشتی خوانده می شود، اگر ماشین تورینگ M وجود داشته باشد که L را پذیرفتـه و روی هـر w موجود در Σ^+ متوقف شود. بنابراین زبان مورد نظر در این تست بازگشتی نمی باشد.

از طرفی زبان مفروض L بازگشتی شمارش پذیر خوانده می شود، اگر ماشین تورینگی وجود داشته باشد که آنرا پـذیرش کنـد. کـه زبان مورد نظر در این تست این چنین نیست پس به طور بازگشتی شمارا نیست.

۵-۳) زبان داده شده، مستقل از متن معین است.

۶-۳) زبان L با شرط $k \le \max(i, j)$، از اجتماع دو زبان با شرط های $k \le i$ و $k \le j$ تشکیل شده است:

$$L = \{a^i b^j c^k \mid i, j, k \ge 1, k \le i\} \mathbf{U} \{a^i b^j c^k \mid i, j, k \ge 1, k \le j\}$$

بنابراین چون این دو زبان مستقل از متن بوده و زبان های مستقل از متن تحت اجتماع بسته هستند، زبان L نیـز مسـتقل از مـتن می باشد و گزینه 3 نادرست است.

یادآوری: هر زبان مستقل از متن، یک زبان حساس به متن نیز می باشد. بنابراین گزینه 2 درست است.

۷-۲) تعداد زیر مجموعه های شمارای یک مجموعه شمارای نامتناهی مانند N ، شمارا نمی باشد.

تذکر: مجموعه تمام زبان هایی که بازگشتی شمارش پذیر نیستند، قابل شمارش نمی باشند.

۸-۳) گزینه 3 درست است، چون می توان ماشین تورینگی ساخت که خروجی نداشته باشد و بـه ازاء هـیچ ورودی نیـز در حلقـه نیافتد، یعنی به ازای هر ورودی متوقف شود.

گزینه 1 نادرست است، چون زبان های تصمیم پذیری وجود دارند که توسط هیچ ماشین پشته ای پذیرفته نمی شوند.

گزینه 2 نادرست است، چون زبان های بازگشتی شمارش پذیری وجود دارند که توسط هیچ DFA ای پذیرفته نمی شوند.

گزینه 4 نادرست است، چون هر زبانی که توسط یک ماشین تورینگ پذیرفتـه مـی شـود، بازگشـتی شـمارش پـذیر اسـت و لزومـا تصمیم پذیر نیست.

۹-۲) طبق قضیه زیر، گرامرهای حساس به متن معادل مدل اتوماتون های کراندار خطی (LBA) هستند.

"اگر زبانL بوسیله یک اتومات کراندار خطی پذیرفته شود، آنگاه یک گرامر حساس به متن وجود دارد کهL را تولید می کند."

۱۰-۱) مجموعه همه ماشین های تورینگ روی یک الفبا، شمارش پذیر است.

منابع

1ـ مقدمه‌ای بر زبانهای رسمی و ماشین تألیف لینز

2ـ تئوری محاسبات تألیف وود

3ـ تئوری زبانهای رسمی تألیف روسز

4ـ کتاب نظریه زبان‌ها و محاسبات سیپسر