فصل هفتم

زبان‌های مستقل از متن

(Context- Free Languages)

7-1: مقدمه

در این فصل رده زبان‌های مستقل از متن را معرفی می‌کنیم. پیش از این دیدیم که تعداد زیادی از زبان‌ها را می‌توان با عبارات منظم توصیف نمود. اما برخی زبان‌های ساده دیگر مانند $\{0^n1^n \mid n \geq 0\}$ وجود دارند که با آن روش‌ها توصیف شدنی نیستند. گرامرهای مستقل از متن روش قدرتمندی برای توصیف زبان‌ها می‌باشد. این گرامرها ابتدا برای مطالعه زبان‌های طبیعی مورد استفاده قرار گرفتند. یک کاربرد مهم گرامرهای مستقل از متن در بیان خصوصیات و کامپایل کردن زبان‌های برنامه‌نویسی می‌باشد. طراحان کامپایلرها و مفسرها برای زبان‌های برنامه‌نویسی معمولاً با طراحی گرامر آن زبان شروع می‌کنند. اکثر کامپایلرها و مفسرها دارای قسمتی به نام تجزیه کننده هستند که معنی یک برنامه را قبل از تبدیل به کد برنامه یا اجرای آن، بدست می‌آورد. به مجموعه زبان‌هایی که می‌توان آنها را با گرامرهای مستقل از متن نشان داد، زبان‌های مستقل از متن می‌گویند. این زبان‌ها شامل تمام زبان‌های منظم و تعداد دیگری از زبان‌ها هستند. زبان‌های مستقل از متن شامل زبان‌هایی هستند که نوعی ساختار بازگشتی دارند. موضوع را با گرامرهای مستقل از متن آغاز می‌کنیم. این گرامرها برای تعریف قوانین دستوری زبان‌های برنامه نویسی و کامپایل کردن آنها به کار می‌رود. سپس مفهوم اتوماتای پشته‌ای (Pushdown automata) که از نوع نامعین هستند، را بیان می‌کنیم و نشان می‌دهیم که این اتوماتاها قدرتی مشابه گرامرهای مستقل از متن دارند. در واقع اتوماتای پشته‌ای نوعی از ماشین‌ها هستند که زبان‌های مستقل از متن را می‌توانند تشخیص دهند. "اتوماتای پشته‌ای" را "ماشین پایین فشردنی" نیز گوییم.

گرامرهای مستقل از متن Context- free grammars با یک مثال شروع می‌کنیم. قوانین (جایگزینی‌ها) زیر را در نظر می‌گیریم:

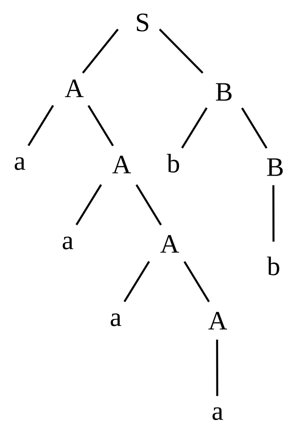
$$S \rightarrow AB$$
$$A \rightarrow a$$
$$A \rightarrow a\,A$$
$$B \rightarrow b$$
$$B \rightarrow bB$$

در این جا، S، A و B متغیرها هستند. S متغیر شروع و a و b پایانه‌ها می‌باشند. با این قوانین رشته‌های شامل پایانه‌ها بدست می‌آوریم یا به عبارتی از آنها مشتق می‌کنیم. این کار را به ترتیب زیر انجام می‌دهیم.

1. در آغاز رشته‌ای را که به عنوان متغیر شروع است، می‌نویسیم.
2. سپس هر متغیری را که در داخل رشته‌ی جاری است برمی‌داریم و همچنین هر قانونی که این متغیر را در سمت چپ خود دارد انتخاب می‌کنیم.
3. مرحله 2 را تا زمانی تکرار می‌کنیم که رشته‌ی جاری تنها شامل متغیرها شود. برای مثال، رشته‌ی aaaabb می‌تواند با روش زیر اشتقاق یابد.

$$\begin{aligned} S &\Rightarrow AB \\ &\Rightarrow a\,AB \\ &\Rightarrow a\,AbB \\ &\Rightarrow aa\,AbB \\ &\Rightarrow aaa\,AbB \\ &\Rightarrow aaaa\,bB \\ &\Rightarrow aaaa\,b\,b \end{aligned}$$

این اشتقاق را می‌توان با درخت تجزیه، در نمودار زیر، نشان داد.



پنج قانون بالا یک گرامر مستقل از متن می‌سازند.

زبان این گرامر، مجموعه‌ی همه رشته‌هایی است که

- بتواند از متغیر شروع اشتقاق یابد و

- تنها شامل پایانه‌ها باشد.

برای این مثال، زبان $\{a^m b^n \mid m \geq ۱, n \geq ۱\}$ بیان می‌شود، زیرا هر رشته‌ای به شکل $a^m b^n$ ، برای برخی $m \geq ۱$ و $n \geq ۱$ ، می‌تواند از متغیر شروع مشتق شود. در حالی که هر رشته‌ای غیر از اینها بر روی الفبای $\{a, b\}$ نمی‌تواند با این متغیر شروع اشتقاق یابد.

7-2: تعریف گرامر مستقل از متن

هر گرامر مستقل از متن عبارتست از چهارتایی $G = (V, \Sigma, R, S)$ که در آن،

1. V یک مجموعه‌ی متناهی است و عناصر آن متغیرها نامیده می‌شوند،
2. Σ یک مجموعه‌ی متناهی است و عناصر آن پایانه‌ها نام دارند،
3. $V \bigcap \Sigma = \phi$ ،
4. S عنصری از V است، آن را متغیر شروع نامند،
5. R یک مجموعه‌ی متناهی است، عناصرش را قوانین گویند، هر قانون شکل $A \to W$ را دارد که $A \in V$ و $W \in (V \bigcap \Sigma)^*$.

در مثالی که ارایه کردیم، داریم $V = \{S, A, B\}$ ، $\Sigma = \{a, b\}$ ، و

$$R = \{S \to AB, A \to a, A \to aA, B \to b, B \to bB\}.$$

تعریف 7-3: فرض می‌کنیم $G = (V, \Sigma, R, S)$ یک گرامر مستقل از متن و A عنصری از V و u، v و W رشته‌های در $(V \bigcup \Sigma)^*$ باشند، به طوری که $A \to W$ قانونی در R باشد. گوییم رشته‌ی uWv می‌تواند با یک گام از رشته‌ی uAv اشتقاق یابد و می‌نویسیم:

$$uAv \Rightarrow uWv$$

به بیان دیگر، با اعمال قانون $A \to W$ به رشته‌ی uAv ، رشته‌ی uWv را بدست می‌آوریم. در مثال قبل می‌بینیم که $aaAbb \Rightarrow aaaAbb$.

تعریف 7-4: فرض کنیم $G=(V,\Sigma,R,S)$ یک گرامر مستقل از متن و u و v رشته‌هایی در $(V\bigcup\Sigma)^*$ باشند. گوییم v می‌تواند از u مشتق شود و می‌نویسیم $u\overset{*}{\Rightarrow}v$، اگر یکی از دو شرط زیر برقرار باشد:

1. $u=v$ یا
2. عدد صحیح $k\geq ۲$ و دنباله $u_۱,u_۲,...,u_k$ از رشته‌های $(V\bigcup\Sigma)^*$ طوری موجود باشند که:

   (a) $u=u_۱$،

   (b) $v=u_k$، و

   (c) $u_۱\Rightarrow u_۲\Rightarrow...\Rightarrow u_k$.

به عبارت دیگر، با شروع از رشته‌ی u و اعمال قوانین، صفر بار یا دفعات بیشتر، رشته‌ی v را بدست می‌آوریم. از مثال قبل می‌بینیم که:

$$aaAbB\Rightarrow aaaabbbB$$

تعریف 7-5: فرض کنیم $G=(V,\Sigma,R,S)$ یک گرامر مستقل از متن باشد. زبان گرامر G به صورت مجموعه‌ی همه رشته‌های در Σ^* تعریف می‌شود که بتوانند از متغیر شروع S اشتقاق یابند، می‌نویسیم،

$$L(G)=\{W\in\Sigma^*\mid S\overset{*}{\Rightarrow}W\}$$

تعریف 7-6: یک زبان L مستقل از متن نامیده می‌شود، اگر یک گرامر مستقل از متن G طوری موجود باشد که $L(G)=L$.

7-7: مثال‌هایی از گرامرهای مستقل از متن

7-7-1: پرانتزهایی که به خوبی لانه‌ای (تو در تو) شده‌اند.

گرامر مستقل از متن $G=(V,\Sigma,R,S)$ را در نظر می‌گیریم، که در آن $V=\{S\}$ ، $\Sigma=\{a,b\}$ ، و $R=\{S\to\varepsilon, S\to aSb, S\to SS\}$.

سه قانون R را به صورت زیر می‌نویسیم:

$$S\to\varepsilon\,|\,aSb\,|\,SS$$

به علامت "|" به عنوان "یا" نگاه کنید.

با اعمال قوانین مربوط و با شروع توسط متغیر S، برای مثال داریم:

$$\begin{aligned}
S &\Rightarrow SS\\
&\Rightarrow aSbS\\
&\Rightarrow aSbSS\\
&\Rightarrow aSSbSS\\
&\Rightarrow aaSbSbSS\\
&\Rightarrow aabSbSS\\
&\Rightarrow aabbSS\\
&\Rightarrow aabbaSbS\\
&\Rightarrow aabbabS\\
&\Rightarrow aabbabaSb\\
&\Rightarrow aabbabab
\end{aligned}$$

زبان $L(G)$ در گرامر مستقل از متن G کدام است؟

اگر به a به عنوان پرانتز چپ یعنی،")" و به b به صورت پرانتز راست یعنی، "(" نگاه کنیم، آنگاه $L(G)$ زبانی شامل همه رشته‌هایی است که به خوبی پرانتز گذاری شده‌اند. حال آن را تعبیر می‌کنیم: هر رشته‌ای که به خوبی پرانتز گذاری تودرتو شده است، یکی از موارد زیر می‌تواند باشد:

- تهی (که از S با قانون $S \rightarrow \varepsilon$ مشتق کردیم)،
- شامل یک پرانتز چپ، و به دنبال آن یک رشته‌ی دلخواه که به خوبی پرانتز گذاری تودرتو شده، و در آخر به یک پرانتز راست ختم شود (اینها از S با شروع به کارگیری قانون $S \rightarrow aSb$ مشتق می‌شوند)، یا
- شامل یک رشته‌ی دلخواه از پرانتزها به خوبی تودرتو شده، و به دنبال آن یک رشته‌ی دلخواه از پرانتزهای به خوبی تودرتو شده، است (اینها از S با شروع به کارگیری قانون $S \rightarrow SS$ مشتق می‌شوند).

2-7-7: گرامر مستقل از متن برای زبان‌های نامنظم

زبان $L_1 = \{0^n 1^n \mid n \geq 0\}$ را در نظر می‌گیریم. بیش از این دیدیم که L_1 یک زبان نامنظم است، گوییم L_1 یک زبان مستقل از متن می‌باشد. برای اثبات این ادعا، باید یک گرامر مستقل از متن بسازیم که $L(G_1) = L_1$.

مشاهده کنید که هر رشته L_1 یا

- خالی است و یا
- شامل یک 0، دنبال شده بوسیله یک رشته‌ی دلخواه در L_1، دنبال شده بوسیله 1 می‌باشد.

این به گرامر مستقل از متن $G_1 = (V_1, \Sigma, R_1, S_1)$ منجر می‌شود که در آن $V_1 = \{S_1\}$، $\Sigma = \{0, 1\}$ و R_1 شامل قوانین $S_1 \rightarrow \varepsilon \mid 0S_1 1$ می‌باشد. از این روی $R_1 = \{S_1 \rightarrow \varepsilon, S_1 \rightarrow 0S_1 1\}$.

برای اشتقاق رشته‌ی $0^n 1^n$ از متغیر شروع S_1 به ترتیب زیر عمل می‌کنیم:

- با S_1 شروع می‌کنیم، قانون $S_1 \rightarrow 0S_1 1$ را به دقیقاً n بار به کار می‌بریم. این رشته‌ی $0^n S_1 1^n$ را حاصل می‌کند.
- قانون $S_1 \rightarrow \varepsilon$ را به کار می‌بریم. این رشته‌ی $0^n 1^n$ را بدست می‌دهد.

به آسانی می‌توان دید که اینها تنها رشته‌های مشتق شده از متغیر شروع S_1 هستند. پس $L(G_1) = L_1$.

با روشی متقارن می‌توان دید که گرامر مستقل از متن $G_2 = (V_2, \Sigma, R_2, S_2)$ که در آن $V_2 = \{S_2\}$، $\Sigma = \{0,1\}$، و R_2 شامل قوانین $S_2 \to \varepsilon \mid 1S_2 0$، دارای این خاصیت است که $L(G_2) = L_2$، و در آن $L_2 = \{1^n 0^n \mid n \geq 0\}$. بنابراین، L_2 یک زبان مستقل از متن است.

زبان $L = L_1 \bigcup L_2$ را تعریف می‌کنیم. یعنی:

$$L = \{0^n 1^n \mid n \geq 0\} \bigcup \{1^n 0^n \mid n \geq 0\}$$

گرامر مستقل ازمتن $G = (V, \Sigma, R, S)$ که $V = \{S, S_1, S_2\}$، $\Sigma = \{0,1\}$، و R شامل قوانین

$$S \to S_1 \mid S_2$$
$$S_1 \to \varepsilon \mid 0S_1 1$$

است، دارای این خاصیت است که $L(G) = L$. از این روی L یک زبان مستقل از متن است.

3-7-7: گرامر مستقل از متن برای مکمل یک زبان نامنظم

فرض کنیم L یک زبان نامنظم به صورت $L = \{0^n 1^n \mid n \geq 0\}$ باشد. می‌خواهیم ثابت کنیم که مکمل L یعنی $\overline{L}$، یک زبان مستقل از متن است. پس باید یک گرامر مستقل از متن G را بسازیم که زبان آن برابر $\overline{L}$ شود. ملاحظه می‌کنیم که یک رشته در $\overline{L}$ است، اگر و تنها اگر:

1. $W = 0^m 1^n$، برای اعداد صحیح m و n ای که، $0 \leq m < n$، یا
2. $W = 0^m 1^n$، برای اعداد صحیح m و n ای که، $0 \leq n < m$، یا
3. W شامل 10 به عنوان یک زیر رشته است.

بنابراین می‌توانیم $\overline{L}$ را به صورت اجتماع زبان‌های همه رشته‌های از نوع 1، نوع 2، و نوع 3، بنویسیم.

هر رشته از نوع 1، یا

- رشته‌ی 1 است.
- شامل یک رشته از نوع 1 است که با 1 دنبال می‌شود، یا
- شامل یک ۰ است که با یک رشته‌ی دلخواه از نوع 1 دنبال می‌شود و به دنبال آن یک 1 می‌آید.

پس با به کارگیری قوانین

$$S_1 \to 1 \mid S_1 1 \mid 0 S_1 1$$

می‌توانیم از S_1، همه رشته‌های نوع 1 را مشتق کنیم.

مشابهاً با قوانین

$$S_2 \to 0 \mid 0 S_2 \mid 0 S_2 1$$

می‌توانیم از S_2، همه رشته‌های نوع 2 را بدست آوریم.

هر رشته از نوع 3

- شامل یک رشته‌ی دودویی دلخواه است که به دنبال آن رشته‌ی ۱۰ می‌آید و سپس با یک رشته‌ی دلخواه دودویی دنبال می‌شود.

با به کارگیری قوانین

$$x \to \varepsilon \mid 0x \mid 1x,$$

می‌توانیم از x، همه رشته‌های دودویی را مشتق کنیم. پس با ترکیب کردن اینها با قانون

$$S_3 \to x10x,$$

می‌توانیم از S_3، همه رشته‌های نوع 3 را مشتق نماییم.

حال گرامر مستقل از متن $G = (V, \Sigma, R, S)$ را بدست آورده‌ایم که در آن $V = \{S, S_1, S_2, S_3, x\}$، $\Sigma = \{0, 1\}$، و R قوانین زیر است:

$$S \rightarrow S_1 \mid S_2 \mid S_3$$
$$S_1 \rightarrow 1 \mid S_1 1 \mid 0 S_1 1$$
$$S_2 \rightarrow 0 \mid 0 S_2 \mid 0 S_2 1$$

خلاصه مطالب چنین است،

به ازای همه اعداد صحیح m و n ای که $0 \leq m < n$، $S_1 \overset{*}{\Rightarrow} 0^m 1^n$،

به ازای همه اعداد صحیح m و n ای که $0 \leq n < m$، $S_2 \overset{*}{\Rightarrow} 0^m 1^n$،

برای هر رشته u در $\{0,1\}^*$، $X \overset{*}{\Rightarrow} u$.

و

برای هر رشته‌ی دودویی w که 10 را به عنوان زیر رشته دارد، $S_3 \overset{*}{\Rightarrow} w$.

از مشاهدات بالا نتیجه می‌شود که $L(G) = \bar{L}$.

4-7-7: گرامر مستقل از متن که درستی جمع را بررسی می‌کند.

زبان $L = \{a^n b^m c^{n+m} \mid n \geq 0, m \geq 0\}$ را در نظر می‌گیریم. با استفاده از لم تزریق که برای زبان‌های منظم بیان کردیم، می‌توان نشان داد که L یک زبان منظم نیست. یک گرامر مستقل از متن G را درست می‌کنیم که زبان آن L باشد، پس نتیجه می‌شود که L یک زبان مستقل از متن است.

ابتدا مشاهده می‌کنیم که $\varepsilon \in L$. در نتیجه، $S \rightarrow \varepsilon$ را به عنوان یکی از قوانین در گرامر اختیار می‌کنیم. حال به بینیم که چگونه می‌توان همه رشته‌های موجود در L را از متغیر شروع S مشتق کرد:

1. هر بار که یک a اضافه می‌کنیم، همچنین یک c می‌افزاییم. بدین طریق همه رشته‌های به شکل $a^n c^n$ که $n \geq 0$، را بدست می‌آوریم.

2. رشته‌ی $a^n c^n$ را در نظر می‌گیریم و افزودن b ها را آغاز می‌کنیم.

هر بار که یک b اضافه می‌کنیم، همچنین یک c نیز می‌افزاییم. ملاحظه کنید که هر b باید بین a ها و c ها اضافه شود. در نتیجه متغیر B را به عنوان"اشاره‌گری" به مکان در رشته‌ی جاری، که یک b بتواند افزوده گردد، به کار می‌بریم. به جای اشتقاق $a^n c^n$ از S، رشته‌ی $a^n Bc^n$ را مشتق می‌کنیم. سپس از B همه رشته‌های به شکل $b^m c^m$، $m \geq 0$، را استخراج می‌نماییم. گرامر $G = (V, \Sigma, R, S)$ را بدست می‌آوریم که در آن $V = \{S, A, B\}$، $\Sigma = \{a, b, c\}$ و R قوانین زیر است:

$$S \to \varepsilon \mid A$$
$$A \to \varepsilon \mid aAc \mid B$$

این واقعیت‌ها که

- $A \overset{*}{\Rightarrow} a^n Bc^n$، برای هر $n \geq 0$،
- $B \overset{*}{\Rightarrow} b^m c^m$، برای هر $m \geq 0$،

نتیجه می‌دهد که رشته‌های زیر می‌توانند از متغیر شروع S مشتق شوند:

- $m \geq 0$ و $n \geq 0$، $S \overset{*}{\Rightarrow} a^n Bc^n \overset{*}{\Rightarrow} a^n b^m c^m c^n = a^n b^m c^{n+m}$

در واقع هیچ رشته‌ی دیگر در $\{a, b, c\}^*$ نمی‌تواند از S مشتق شود. پس $L(G) = L$.

چون $S \Rightarrow A \Rightarrow B \Rightarrow \varepsilon$،

می‌توانیم این گرامر G را با حذف قانون $S \to \varepsilon$ و $A \to \varepsilon$ ساده سازی کنیم. این گرامر مستقل از متن $G' = (V, \Sigma, R', S)$ را ارایه می‌کند که در آن $V = \{S, A, B\}$، $\Sigma = \{a, b, c\}$ و R' شامل قوانین زیر است:

$$S \to A$$
$$A \to aAc \mid B$$

سرانجام مشاهده می‌کنیم که به S نیاز نداریم و به جای آن از A به عنوان متغیر شروع استفاده می‌کنیم. بدین ترتیب گرامر مستقل از متن نهایی $G''=(V,\Sigma,R'',S)$ را داریم که در آن $V=\{A,B\}$ ، $\Sigma=\{a,b,c\}$ و R'' شامل قوانین زیر است:

$$A \rightarrow aAc \mid B$$
$$B \rightarrow \varepsilon \mid bBc$$

7-8: زبان‌های منظم، مستقل از متن هستند.

پیش از این اشاره کردیم که رده زبان‌های مستقل از متن شامل رده زبان‌های منظم می‌باشد. در این بخش این ادعا را ثابت می‌کنیم.

قضیه 7-8-1: فرض کنیم Σ یک الفبا و $L \subseteq \Sigma^*$ یک زبان منظم باشند. آنگاه L یک زبان مستقل از متن است.

اثبات: چون L یک زبان منظم است، پس یک اتوماتای متناهی معین $M=\{Q,\Sigma,\delta,q,F\}$ وجود دارد که L را پذیرش می‌کند.

برای اثبات مستقل از متن بودن L، باید یک گرامر مستقل از متن $G=(V,\Sigma,R,S)$ را طوری تعریف کنیم که $L=L(M)=L(G)$. بنابراین G باید خواص زیر را داشته باشد:

به ازای هر رشته $w \in \Sigma^*$ ،

$w \in L(M)$ اگر و تنها اگر $w \in L(G)$

که می‌تواند به صورت زیر فرمول‌بندی شود:

M، w را می‌پذیرد اگر و تنها اگر $S \overset{*}{\Rightarrow} w$.

گرامر مستقل از متن G را طوری تعریف می‌کنیم که تناظر زیر برای هر رشته‌ی $w = w_1 w_2 ... w_n$ برقرار شود:

- فرض می‌کنیم M پس از خواندن زیر رشته‌ی $w_1 w_2 ... w_i$ ، درست در حالت A قرار گیرد.
- آنگاه در گرامر مستقل از متن G، داریم $S \overset{*}{\Rightarrow} w_1 w_2 ... w_i A$.

در گام بعدی، M نماد w_{i+1} را می‌خواند و از A به مثلاً B می‌رود، پس $\delta(A, w_{i+1}) = B$. به منظور تضمین این که تناظر بالا باز هم برقرار است، باید قانون $A \to w_{i+1} B$ را به G اضافه کنیم.

حال زمانی را در نظر می‌گیریم که M همه رشته‌ی w را خوانده است. فرض کنیم A حالت آن لحظه‌ی M باشد. با استفاده از تناظر بالا داریم:

$$S \overset{*}{\Rightarrow} w_1 w_2 ... w_n \, A = wA$$

یادآوری می‌کنیم که G باید دارای خاصیت زیر باشد:

M رشته‌ی w را پذیرش می‌کند اگر و تنها اگر $S \overset{*}{\Rightarrow} w$ ،

که معادل است با $A \in F$ اگر و تنها اگر $S \overset{*}{\Rightarrow} w$.

این خاصیت با افزودن قانون $A \to \varepsilon$ که A هر حالت پذیرش از M است، به G، تضمین می‌شود.

حال آماده‌ایم تا تعریف رسمی گرامر مستقل از متن $G = (V, \Sigma, R, S)$ را بیان کنیم.

- $V = Q$ ، یعنی متغیرهای شروع G، همان حالت‌های M هستند.
- $S = q$ ، یعنی متغیر شروع G، همان حالت شروع M است.
- R شامل قوانین زیر است:

$A \to aB$ که $A \in Q$ ، $a \in \Sigma$ ، $B \in Q$ ، و $\delta(A, a) = B$ و $A \to \varepsilon$ که در آن $A \in F$.

موارد بالا را می‌توان چنین بیان کرد:

- هر انتقال $\delta(A,a)=B$ از M (یعنی وقتی M در حالت A است و نماد a را می‌خواند، به B انتقال می‌یابد.) متناظر است با یک قانون $A \to aB$ در گرامر G.
- هر حالت پذیرش A از M متناظر با یک قانون $A \to \varepsilon$ ، در گرامر G، می‌باشد. گوییم $L(G)=L$. برای اثبات باید نشان دهیم که: $L \subseteq L(G), L(G) \subseteq L$.

ثابت می‌کنیم $L \subseteq L(G)$. فرض کنیم $w = w_1 w_2 ... w_n$ یک رشته‌ی دلخواه در L باشد. زمانی که اتوماتای متناهی M رشته‌ی w را می‌خواند، حالات $r_0, r_1, ..., r_n$ را می‌بینند که:

- $r_0 = q$ ، و
- $r_{i+1} = \delta(r_i, w_{i+1})$ برای $i = 0,1,...,n-1$.

چون $w \in L = L(M)$ ، معلوم است که $r_n \in F$.

از روش تعریف کردن گرامر G چنین بر می‌آید که:

- برای هر $i = 0,1,...,n-1$ ، قانون $r_i \to w_{i+1} r_{i+1}$ در R وجود دارد و
- $r_n \to \varepsilon$ یک قانون در R است.

بنابراین، داریم:

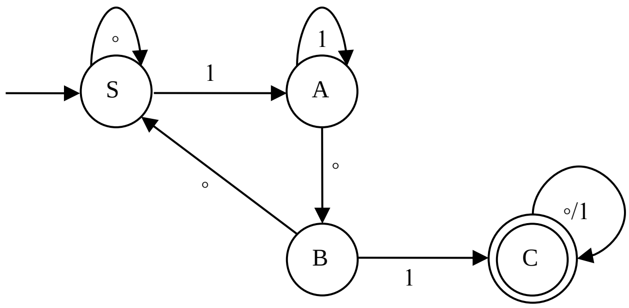
$$S = q = r_0 \Rightarrow w_1 r_1 \Rightarrow w_1 w_2 r_2 \Rightarrow ... \Rightarrow w_1 w_2 ... w_n r_n \Rightarrow w_1 w_2 ... w_n = w$$

و این ثابت می‌کند که $w \in L(G)$. اثبات $L(G) \subseteq L$ را به عنوان تمرین واگذار می‌کنیم. پیش از این دیدیم که $\{0^n 1^n \mid n \ge 0\}$ منظم نیست، اما مستقل از متن است. پس، راه همه زبان‌های مستقل از متن، به خوبی شامل رده زبان‌های منظم می‌باشد.

مثال 7-8-2: فرض کنیم L زبان تعریف شده زیر باشد:

$$L = \{w \in \{0,1\}^* \mid \text{یک زیر رشته از } w \text{ است } 101\}$$

دیدیم که L یک زبان منظم است و اتوماتای متناهی معین M که L را پذیرش می‌کند به صورت نمودار زیر نشان دادیم:



می‌توان اثبات کرد که M می‌تواند به یک گرامر مستقل از متن G با زبان L، قابل تبدیل است (موضوع این نوع تبدیل کردن‌ها در این کتاب آورده نشده است و تنها به پذیرفتن آن اکتفا می‌کنیم).

بر طبق ساختار تبدیل یافته داریم $G = (V, \Sigma, R, S)$ که $V = \{S, A, B, C\}$ ، $\Sigma = \{۰, ۱\}$ ، متغیر شروع S همان حالت شروع M است و R قوانین زیر می‌باشد:

$$S \rightarrow ۰S \mid ۱A$$
$$A \rightarrow ۰B \mid ۱A$$
$$B \rightarrow ۰S \mid ۱C$$

رشته‌ی ۰۱۰۰۱۱۰۱۱ که عضوی از L است، را در نظر می‌گیریم، وقتی اتوماتای متناهی M این رشته را می‌خواند، حالات زیر را می‌بیند:

$$S, S, A, B, S, A, A, B, C, C.$$

در گرامر G، این متناظر با اشتقاق زیر است:

$$
\begin{aligned}
S &\Rightarrow ۰S \\
&\Rightarrow ۰۱A \\
&\Rightarrow ۰۱۰B \\
&\Rightarrow ۰۱۰۰S \\
&\Rightarrow ۰۱۰۰۱A \\
&\Rightarrow ۰۱۰۰۱۱A
\end{aligned}
$$

پس اشتقاق

$$S \overset{*}{\Rightarrow} ۰۱۰۰۱۱۰۰۱۱$$

بیان می‌کند که رشته‌ی ۰۱۰۰۱۱۰۰۱۱ در زبان $L(G)$ از گرامر مستقل از متن G قرار دارد.

رشته‌ی ۱۰۰۱۱ در زبان L قرار ندارد. زمانی که اتوماتای متناهی M این رشته را می‌خواند. حالات زیر را مشاهده می‌کند:

$$S, A, B, S, A, A$$

یعنی وقتی رشته خوانده شده، M در حالت عدم پذیرش A است. در گرامر G، خواندن رشته‌ی ۱۰۰۱۱ متناظر است با اشتقاق زیر:

$$
\begin{aligned}
S &\Rightarrow ۱A \\
&\Rightarrow ۱۰B \\
&\Rightarrow ۱۰۰S
\end{aligned}
$$

چون A یک حالت پذیرش در M نیست، گرامر G قانون $A \to \varepsilon$ را در خود ندارد. این بیان می‌کند که رشته‌ی ۱۰۰۱۱ نمی‌تواند از متغیر شروع S اشتقاق یابد. بنابراین، رشته‌ی ۱۰۰۱۱ در زبان $L(G)$ از G، نیست.

7-9: فرم نرمال چامسکی (Chomsky normal form)

دیدیم قوانین گرامر مستقل از متن $G = (V, \Sigma, R, S)$ به صورت $A \to w$ ، هستند که در آن A یک متغیر و w رشته‌ای روی الفبای $V \bigcup \Sigma$ می‌باشد. در این بخش نشان می‌دهیم که هر گرامر مستقل از متن G را می‌توان به گرامر مستقل از متن G' تبدیل کرد. به طوری که $L(G) = L(G')$ و قوانین G' به فرم‌های مشخصی هستند. این موضوع در تعریف زیر بیان می‌گردد.

تعریف 7-9-1: یک گرامر مستقل از متن $G = (V, \Sigma, R, S)$ را به فرم نرمال چامسکی گویند، اگر هر قانون در R یکی از سه فرم زیر را داشته باشد:

1. $A \to BC$ ، که A، B و C اعضای V هستند و $B \neq S$ ، $C \neq S$.
2. $A \to a$ ، که A عضوی از V است و a عنصری از Σ می‌باشد.
3. $S \to \varepsilon$ ، که S متغیر شروع است.

می‌توان در نظر داشت که برای چنین گرامری، R، قانون $S \to \varepsilon$ را شامل می‌شود، اگر و تنها اگر $\varepsilon \in L(G)$.

قضیه 7-9-2: فرض کنیم Σ یک الفبا و $L \subseteq \Sigma^*$ یک زبان مستقل از متن باشند. آنگاه یک گرامر مستقل از متن در فرم نرمال چامسکی وجود دارد، که زبان آن L است.

اثبات: چون L یک زبان مستقل از متن است، پس یک گرامر مستقل از متن $G = (V, \Sigma, R, S)$ طوری وجود دارد که $L(G) = L$.

حال G را به گرامری تبدیل می‌کنیم که به فرم چامسکی باشد و زبان آن برابر $L(G)$ شود. این انتقال شامل پنج گام زیر است:

گام 1: متغیر شروع را از سمت راست قوانین حذف می‌کنیم. گرامر $G_1 = (V_1, \Sigma, R_1, S_1)$ را تعریف می‌کنیم که در آن S_1 متغیر شروع (متغیر جدیدی است)، $V_1 = V \bigcup \{S_1\}$، و $R_1 = R \bigcup \{S_1 \to S\}$. این گرامر خواص زیر را دارد:

- متغیر شروع S_1 در سمت راست هیچ یک از قوانین در R_1، ظاهر نمی‌شود، و
- $L(G_1) = L(G)$.

گام 2: یک $\varepsilon-$ قانون، قانونی است که به شکل $A \to \varepsilon$ باشد که در آن A یک متغیری است که با متغیر شروع برابر نمی‌باشد. در گام 2، همه $\varepsilon-$ قوانین‌ها را از G_1 حذف می‌کنیم. همه $\varepsilon-$ قوانین‌ها را یکی پس از دیگری بررسی می‌کنیم. فرض کنیم $A \to \varepsilon$ یکی از این قوانین باشد که $A \in V_1$ و $A \neq S_1$. حال G_1 را به صورت زیر اصلاح می‌کنیم:

1. قانون $A \to \varepsilon$ را از مجموعه R_1 حذف می‌کنیم.
2. برای هر قانون در مجموعه جاری R_1 به شکل

(a) $B \to A$، قانون $B \to \varepsilon$ را به R_1 می‌افزاییم، مگر این که این قانون پیش از این از R_1 حذف شده باشد، مشاهده کنید که بدین طریق، اشتقاق دوگامی $B \Rightarrow A \Rightarrow \varepsilon$ را با اشتقاق یک گامی $B \Rightarrow \varepsilon$ جایگزین می‌نماییم.

(b) $B \to uAv$ (که u و v رشته‌هایی هستند که هر دو با هم تهی نیستند)، قانون $B \to uv$ را به R_1 می‌افزاییم، مشاهده کنید که بدین طریق، اشتقاق دو گامی $B \Rightarrow uAv \Rightarrow uv$ را با اشتقاق یک گامی $B \to uv$ جایگزین می‌نماییم.

(c) $B \to uAvAw$ (که u، v و w رشته‌ها هستند)، قوانین $B \to uvw$، $B \to uAvw$ و $B \to uvAw$ را به R_1 می‌افزاییم، اگر $u = v = w = \varepsilon$ و قانون $B \to \varepsilon$ پیش از این از R_1 حذف شده است، آنگاه قانون $B \to \varepsilon$ را اضافه نمی‌کنیم.

(d) قوانین را با همین روش، در آنجا که A پیش از دو بار در سمت راست رخ دهد، ادامه می‌دهیم.

این روند را تا زمانی تکرار می‌کنیم که همه‌ی ε – قوانین‌ها، حذف شده باشند: فرض کنیم R_2 مجموعه‌ی همه قوانین، پس از حذف همه ε – قوانین‌ها، باشد. گرامر $G_2 = (V_2, \Sigma, R_2, S_2)$ را تعریف می‌کنیم که در آن $V_2 = V_1$ و $S_2 = S_1$. این گرامر دارای خواصی است که در آن:

- متغیر شروع S_2 در سمت راست هیچ قانونی در R_2، ظاهر نمی‌شود.
- R_2 شامل هیچ ε – قانون نیست (ممکن است قانون $S_2 \rightarrow \varepsilon$ را شامل شود)، و
- $L(G_2) = L(G_1) = L(G)$.

گام 3: یک واحد- قانون، قانونی است که به فرم $A \rightarrow B$ می‌باشد که در آن A و B متغیرها هستند. در گام 3، همه‌ی واحد- قانون‌ها را از G_2 حذف می‌کنیم.

همه واحد- قانون‌ها را یکی پس از دیگری در نظر می‌گیریم. فرض کنیم $A \rightarrow B$ یکی از این قانون‌ها باشد که A و B اعضای V_2 هستند. می‌دانیم که $B \neq S_2$، حال G_2 را به صورت زیر اصلاح می‌کنیم:

1. قانون $A \rightarrow B$ را از مجموعه‌ی جاری R_2 حذف می‌کنیم.
2. به ازای هر قانون در مجموعه‌ی جاری R_2 که به شکل $B \rightarrow u$ است، و $u \in (V_2 \cup \Sigma)^*$، قانون $A \rightarrow u$ را به مجموعه‌ی جاری R_2 اضافه می‌کنیم، مگر این که یک واحد- قانون باشد که پیش از این حذف شده است.

مشاهده کنید که بدین ترتیب، اشتقاق دو گامی $A \Rightarrow B \Rightarrow u$ با اشتقاق یک گامی $A \Rightarrow u$ جایگزین شده است.

این روند را تا زمانی تکرار می‌کنیم که همه واحد- قانون‌ها حذف شده باشند. فرض کنیم R_3 مجموعه‌ی قوانین، پس از انجام حذف‌های واحد- قانون‌ها باشد. حال گرامر $G_3 = (V_3, \Sigma, R_3, S_3)$ را تعریف می‌کنیم که در آن $V_3 = V_2$ و $S_3 = S_2$. این گرامر دارای خواصی است که در آن:

- متغیر شروع S_3 در سمت راست هیچ قانونی در R_3 ظاهر نمی‌شود.

- R_3 شامل هیچ ε-قانون نیست (ممکن است شامل قانون $S_3 \to \varepsilon$ باشد)،
- R_3 هیچ واحد- قانونی را در بر ندارد و
- $L(G_3) = L(G_2) = L(G_1) = L(G)$.

گام 4: حذف همه‌ی قوانینی که بیشتر از دو نماد در سمت راست دارند. به ازای هر قانون در مجموعه‌ی جاری R_3 که به شکل $A \to u_1u_2...u_k$، $k \geq 3$ است و هر u_i که عضوی از $V_3 \bigcup \Sigma$ می‌باشد، G را به صورت زیر اصلاح می‌کنیم:

1. قانون $A \to u_1u_2...u_k$ را از مجموعه‌ی جاری R_3 حذف می‌کنیم.
2. قوانین زیر را به مجموعه‌ی R_3 اضافه می‌نماییم.

$$\begin{aligned} &A \to u_1A_1 \\ &A_1 \to u_2A_2 \\ &A_2 \to u_3A_3 \\ &\quad\vdots \end{aligned}$$

که در آن $A_{k-2}, ..., A_2, A_1$ متغیرهای جدیدی هستند که به مجموعه‌ی جاری V_3 افزوده شده‌اند. مشاهده کنید که بدین ترتیب، اشتقاق یک گامی $A \to u_1u_2...u_k$ با اشتقاق $k-1$ گاهی جایگزین شده است.

$$A \Rightarrow u_1A_1 \Rightarrow u_1u_2A_2 \Rightarrow ... \Rightarrow u_1u_2...u_{k-2}A_{k-2} \Rightarrow u_1u_2...u_K$$

فرض کنیم R_4 مجموعه‌ی قوانین و V_4 مجموعه‌ی متغیرها، پس از حذف همه‌ی قوانینی که بیش از دو نماد در سمت راست دارند، باشند. حال تعریف می‌کنیم: $G_4 = (V_4, \Sigma, R_4, S_4)$ که $S_4 = S_3$. این گرامر دارای خواصی است که در آن:

- متغیر S_4 در سمت راست هیچ قانونی در R_4 ظاهر نمی‌شود،
- R_4 شامل هیچ ε-قانونی نیست (ممکن است قانون $S_4 \to \varepsilon$ را داشته باشد)،

- R_4 شامل هیچ واحد- قانون نیست،
- R_4 هیچ قانونی را که بیش از دو نماد در سمت راست دارد، در بر ندارد و
- $L(G_4) = L(G_3) = L(G_2) = L(G_1) = L(G)$.

گام 5: حذف همه‌ی قوانینی که به شکل $A \to u_1u_2$ هستند که در آن u_1 و u_2 هر دو با هم متغیر نیستند.

به ازای هر قانون در مجموعه‌ی جاری R_4 که به شکل $A \to u_1u_2$ است که در آن u_1 و u_2 اعضای $V_4 \bigcup \Sigma$ می‌باشند، اما u_1 و u_2 هر دو با هم در V_4 نیستند، گرامر G_3 را به صورت زیر اصلاح می‌کنیم:

1. اگر $u_1 \in \Sigma$ و $u_2 \in V_4$، آنگاه در مجموعه‌ی جاری R_4، قانون $A \to u_1u_2$ را با دو قانون $A \to U_1u_2$ و $U_1 \to u_1$، جایگزین می‌کنیم که در آن U_1 یک متغیر جدید است که به مجموعه‌ی جاری V_4 افزوده شده است.

مشاهده کنید که به این ترتیب، یک اشتقاق یک گامی $A \Rightarrow u_1u_2$ با یک اشتقاق دو گامی $A \Rightarrow U_1u_2 \Rightarrow u_1u_2$ جایگزین شده است.

2. اگر $u_1 \in V_4$ و $u_2 \in \Sigma$، آنگاه در مجموعه‌ی جاری R_4، قانون $A \to u_1u_2$ را با دو قانون $A \to u_1U_2$ و $U_2 \to u_2$، جایگزین می‌کنیم که در آن U_2 یک متغیر جدید است که به مجموعه‌ی جاری V_4 اضافه شده است.

مشاهده می‌کنید که به این ترتیب، یک اشتقاق یک گامی $A \to u_1u_2$ با یک اشتقاق دو گامی $A \Rightarrow u_1U_2 \Rightarrow u_1u_2$ جایگزین شده است.

3. اگر $u_1 \in \Sigma$، $u_2 \in \Sigma$ و $u_1 \neq u_2$، آنگاه قانون $A \to u_1u_2$ را در مجموعه‌ی جاری R_4، با سه قانون $A \to U_1U_2$، $U_1 \to u_1$ و $U_2 \to u_2$ جایگزین می‌کنیم که در آن U_1 و U_2 متغیرهای جدید هستند که به مجموعه‌ی جاری V_4 اضافه شده‌اند.

مشاهده می‌کنید که به این ترتیب، اشتقاق یک گامی $A \to u_1u_2$ با یک اشتقاق سه گامی $A \Rightarrow U_1U_2 \Rightarrow u_1U_2 \Rightarrow u_1u_2$ جایگزین شده است.

4. اگر $u_1 \in \Sigma$، $u_2 \in \Sigma$، و $u_1 = u_2$، آنگاه در مجموعه‌ی جاری R_4، قانون $A \to u_1u_2 = u_1u_1$ را با دو قانون $A \to U_1U_1$ و $U_1 \to u_1$ جایگزین می‌کنیم، که در آن U_1 یک متغیر جدید است که به مجموعه‌ی جاری V_4 اضافه شده است.

مشاهده می‌کنید که به این ترتیب، اشتقاق یک گامی $A \Rightarrow u_1u_2 = u_1u_1$ با اشتقاق سه گامی $A \Rightarrow U_1U_1 \Rightarrow u_1U_1 \Rightarrow u_1u_1$ جایگزین شده است.

فرض کنیم R_5 مجموعه‌ی قوانین و V_5 مجموعه‌ی متغیرها، پس از انجام کامل گام 5، باشند. حال گرامر $G_5 = (V_5, \Sigma, R_5, S_5)$ را تعریف می‌کنیم که در آن $S_5 = S_4$. این گرامر دارای خواصی است که در آن:

- متغیر S_8 در سمت راست هیچ قانونی در R_8 ظاهر نمی‌شود،
- R_8 شامل هیچ ε - قانون نیست (ممکن است شامل قانون $S_8 \to \varepsilon$ باشد)،
- R_8 هیچ واحد- قانونی را در بر ندارد،
- R_8 هیچ قانونی که دارای دو نماد در سمت راست باشد را در بر ندارد،
- R_5 هیچ قانونی به شکل $A \to u_1u_2$ را در بر ندارد که u_1 و u_2 هر دو با هم متغیرهای V_8 نیستند و
- $L(G_5) = L(G_4) = L(G_3) = L(G_2) = L(G_1) = L(G)$.

پس G_5 در فرم نرمال چامسکی است و اثبات تمام است.

7-9-3: مثال

گرامر مستقل از متن $G = (V, \Sigma, R, A)$ را در نظر می‌گیریم که در آن $V = \{A, B\}$، $\Sigma = \{0, 1\}$، A متغیر شروع و R شامل قوانین زیر است:

$$A \to BAB \mid B \mid \varepsilon$$
$$B \to 00 \mid \varepsilon$$

این گرامر را به ترتیب زیر به یک گرامر به فرم نرمال چامسکی تبدیل می‌کنیم، زبان آن همان زبان مربوط به G است. حروف درشت متغیرها هستند.

گام 1: متغیر شروع را از سمت راست قوانین حذف می‌کنیم. متغیر شروع جدید را معرفی می‌کنیم و قانون $S \to A$ را اضافه می‌نماییم. حال گرامر زیر را داریم:

$$S \to A$$
$$A \to BAB \mid B \mid \varepsilon$$

گام 2: همه‌ی ε - قانون‌ها را حذف می‌کنیم.

ε - قانون $A \to \varepsilon$ را برداشته و آن را حذف می‌کنیم. آنگاه همه‌ی قوانینی را در نظر می‌گیریم که A را در سمت راست خود دارند. دو قانون زیر را داریم:

- $S \to A$، قانون $S \to \varepsilon$ را اضافه می‌کنیم،
- $A \to BAB$، قانون $A \to BB$ را اضافه می‌نماییم.

حال گرامر زیر را داریم:

$$S \to A \mid \varepsilon$$
$$A \to BAB \mid B \mid BB$$

ε - قانون را برداشته و آن را حذف می‌کنیم. آنگاه همه قوانینی را در نظر می‌گیریم که B را در سمت راست خود دارند. سه قانون زیر را داریم:

- $A \to BAB$، قوانین $A \to AB$ ، $A \to BA$ ، و $A \to A$ را اضافه می‌کنیم،

- $A \to B$ ، قانون $A \to \varepsilon$ را اضافه نمی‌کنیم، زیرا این قانون پیش از این حذف شده است. چون همه واحد- قانون‌ها حذف شده‌اند، پس گام 3 انجام شده است.

گام 4: همه‌ی قوانینی را که بیش از دو نماد در سمت راست دارند، حذف می‌کنیم. دو قانون زیر را داریم:

- قانون $S \to BAB$ را برداشته و آن را حذف می‌کنیم، و قوانین $S \to BA_1$ و $A_1 \to AB$ را اضافه می‌کنیم.
- قانون $A \to BAB$ را برداشته و آن را حذف می‌کنیم، و قوانین $A \to BA_2$ و $A_2 \to AB$ را اضافه می‌کنیم.

حال گرامر زیر را داریم:

$$
\begin{aligned}
&S \to \varepsilon \mid BB \mid AB \mid BA \mid \circ\circ \mid BA_1 \\
&A \to BB \mid AB \mid BA \mid \circ\circ \mid BA_2 \\
&B \to \circ\circ
\end{aligned}
$$

پس گام 4 کامل شده است.

گام 5: همه‌ی قوانینی را حذف می‌کنیم که در سمت راست آنها دقیقاً دو نماد است که هر دو با هم متغیر نیستند. سه قانون زیر را داریم.

- قانون $S \to \circ\circ$ را با قوانین $S \to A_3A_3$ و $A_3 \to \circ$ جایگزین می‌کنیم.
- قانون $A \to \circ\circ$ را با قوانین $A \to A_4A_4$ و $A_4 \to \circ$ جایگزین می‌کنیم.
- قانون $B \to \circ\circ$ را با قوانین $B \to A_5A_5$ و $A_5 \to \circ$ جایگزین می‌کنیم.

حال گرامر زیر را داریم که به فرم نرمال چامسکی است:

$$
\begin{aligned}
&S \rightarrow \varepsilon \mid BB \mid AB \mid BA \mid BA_{1} \mid A_{3}A_{3}\\
&A \rightarrow BB \mid AB \mid BA \mid BA_{2} \mid A_{4}A_{4}\\
&B \rightarrow A_{5}A_{5}\\
&A_{1} \rightarrow AB\\
&A_{2} \rightarrow AB
\end{aligned}
$$

7-10: اتوماتای پشته‌ای یا ماشین پایین فشردنی (Pushdown automata)

در این بخش نوع دیگری از مدل محاسباتی به نام اتوماتای پشته‌ای یا ماشین پایین فشردنی را معرفی می‌کنیم. این اتوماتا مانند اتوماتای متناهی نامعین می‌باشد، با این تفاوت که دارای یک جزء اضافی به نام پشته است. پشته علاوه بر حافظه محدودی که در واحد کنترل وجود دارد، حافظه‌ای اضافی به ماشین می‌دهد. خواهیم دید، رده زبان‌هایی که می‌تواند بوسیله این اتوماتاها پذیرفته شود، دقیقاً رده زبان‌های مستقل از متن است.

موضوع را با یک توصیف غیر رسمی از اتوماتای پشته‌ای معین شروع می‌کنیم. چنین اتوماتایی شامل موارد ذیل می‌باشد، همچنین شکل 7-1 را ببینید.

1. یک نوار وجود دارد که به سلول‌های حافظه‌ای تقسیم می‌شود. هر سلول یک نماد متعلق به یک مجموعه‌ی متناهی Σ را در خود ذخیره می‌کند. مجموعه‌ی Σ را الفبای نوار گویند. یک نماد مخصوص مانند □ وجود دارد که در Σ موجود نیست. این نماد را نماد خالی blank symbol گویند. اگر یک سلول شامل □ باشد، آنگاه این به این معنی است که سلول دقیقاً خالی است.
2. یک هد نوار (Tape head) وجود دارد که می‌تواند در طول نوار حرکت کند، در هر حرکت یک سلول به راست می‌رود. این هد همچنین می‌تواند سلولی که اخیراً اسکن شده است را بخواند.
3. یک استک (stack) وجود دارد که نمادهای یک مجموعه‌ی متناهی Γ در آن قرار دارند و آن را الفبای استک نامند. این مجموعه شامل نماد خاص $\$$ می‌باشد.