

تجزیه و تحلیل عبارات

اسمها و افعال

مفاهیم پایه

$|w| = abea = 4$
 $w = a$ ①
 $w = ab$ ②
 $w = abea$ ③
 $w =$ ④

الفبا: مجموعه ای است متناهی از عناصر ساده و تجزیه ناپذیر که عموماً با حرف Σ نمایش داده می شود. (طول هر عنصر در الفبا یک است).
 رشته: دنباله ای است متناهی از عناصر الفبایی که متعلق به یک مجموعه الفبا می باشد.
 طول رشته: تعداد عناصر تشکیل دهنده رشته، طول رشته نامیده می شود.
 رشته λ : رشته به طول صفر، رشته λ نامیده می شود.

کلیه رشته هایی که از عناصر مجموعه Σ قابل تولید باشند، به صورت Σ^* نمایش داده می شود.
 Σ^* دارای سه خاصیت زیر است:
 الف) $\lambda \in \Sigma^*$

ب) اگر $\omega \in \Sigma^*$ و $a \in \Sigma$ آنگاه $a\omega \in \Sigma^*$.
 (ا) یک عنصر الفبا و ω یک رشته از عناصر الفبا می باشد.)

ج) $\omega \in \Sigma^*$ است اگر و فقط اگر ω قابل حصول از عنصر λ با استفاده از عملیات های خاصیت (ب) باشد.

نکته: اگر Σ شامل n عنصر باشد، رشته n^k عنصری (بطول k) در Σ^* وجود دارد.

$\Sigma = \{a, b\}$
 $\Sigma^* = \{\lambda, a, b, aa, aab, \dots\}$

$X \cdot Y \neq Y \cdot X$

اتحاد دو مجموعه از رشته ها (Unions)

اتحاد دو مجموعه X و Y از زیر مجموعه های Σ^* به صورت زیر می باشد:

$X \cup Y = Y \cup X = \{\omega \mid \omega \in X \text{ or } \omega \in Y\}$

اتصال دو مجموعه از رشته ها (Concat)

اتصال دو مجموعه X و Y از زیر مجموعه های Σ^* به صورت زیر می باشد:

$X \cdot Y = \{\omega = \omega_1 \omega_2 \mid \omega_1 \in X, \omega_2 \in Y\}$
 and

اتصال دو رشته

اگر u, v متعلق به Σ^* باشند و دو رشته باشند، اتصال این دو رشته با استفاده از قوانین زیر تعریف می شود:
 الف- اگر طول v برابر صفر باشد همان λ است و

$uv = u\lambda = \lambda u = u$

ب- اگر طول v بیشتر از صفر باشد در این صورت:

$a \in \Sigma, \omega \in \Sigma^*$

موجود است، به طوری که $v = \omega a$ و طول ω برابر است با طول v منهای 1 و در نتیجه داریم:

$v - 1 = \omega$

$uv = u(\omega a) = (u\omega)a$

نکته: برای دو مجموعه رشته X, Y که از دو مجموعه الفبای متفاوت ساخته شده اند، اتحاد و اتصال تعریف نشده است.

نظریه زبانها و ماشینها

علیرضا قوامی

نکته: $(ab)^* \neq a^* b^*$

$(ab)^* = \lambda \mid ab \mid a^2 b^2 \mid a^3 b^3 \mid \dots$

$a^* b^* = \{a^i b^j \mid i, j \ge 0\}$

عملگرهای * و +

اگر X مجموعه ای از رشته ها باشد، X^+ مجموعه ای است از رشته هایی که از اتصال پی در پی عناصر X با خودش تولید می شود و X^0 برابر:

$X^0 = X^+ \cup \{\lambda\}$

و می توان نوشت:

$X^0 = X^0 \cup X^1 \cup X^2 \cup \dots \cup X^i = \cup X^n \quad i \geq 0$

$X^+ = \cup X^i \quad i > 0$

اگر X مجموعه ای از رشته ها باشد، با توجه به مطالب گفته شده، داریم: $X^n \subset X^*$ است که عناصر آن دارای طول n می باشد.

$\Sigma^+ = \cup_{n>0} \Sigma^{(n)}$

$\Sigma^* = \cup_{n \geq 0} \Sigma^{(n)}$

نکته: در مواقعی ممکن است در X^+ هم λ وجود داشته باشد و آن صورتی است که اجتماع $\lambda \cup X^n$ را تولید نماید که در این صورت: $X^* = X^+$ که X^* با X^+ برابر شد.

عبارت با قاعده (Regular Expression)

$S \rightarrow a^i b^j$

اگر Σ مجموعه الفبا باشد، عبارت با قاعده روی Σ به صورت بازگشتی توسط ۳ قانون زیر تعریف می شود:

۱. λ و $\forall a \in \Sigma$ عبارات با قاعده هستند.

۲. اگر u, v دو عبارت با قاعده باشند آنگاه $u \cup v$ و $u \cdot v$ و u^* و u^+ و $u-v$ و $v-u$ عبارات با قاعده هستند.

۳. u یک عبارت با قاعده است، اگر بتوان آن را از عبارات های ابتدایی با انجام تعداد محدودی از عملیات های قسمت ۲ بدست آورد.

$[(a^+ \cup b^+)^* \cdot (a^+ b^+)^+]$

نکته: تفاضل دو عبارت با قاعده مانند u, v که به صورت $u-v$ نوشته می شود، مجموعه رشته هایی را توصیف می نماید که در u می باشند ولی در v وجود ندارد.

$S \rightarrow \lambda$
 $S \rightarrow ab$
 $S \rightarrow aab$
 $S \rightarrow a^2 b^2$

مثال ۱: اگر $\Sigma = \{a, b, c\}$ و $X_1 = \{a, ab, abc\}$ و $X_2 = \{a, aa, aaa\}$ حاصل $X_1 X_2$ کدام است؟

- ۱. $\{a, aa, ab, abc, aaa\}$
- ۲. $\{aa, aab, aabc, aaa, aaab, aaabc, aaaa, aaaab, aaaabc\}$
- ۳. $\{aa, aaa, aaaa, aba, abaa, abaaa, abca, abcaa, abcaaa\}$ ✓
- ۴. $\{aa, abaa, abcaaa\}$

$\{a^n b^n \mid n \geq 0\}$

مثال ۲: اگر $\Sigma = \{a, b\}$ باشد، کدامیک از عبارات زیر نادرست است؟

- ۱. $\{a\}^+$ مجموعه رشته هایی را نشان می دهد که فقط از a تشکیل شده و حداقل طول ۱ دارند.
- ۲. $\{a\}^*$ مجموعه رشته هایی را نشان می دهد که از a تشکیل شده و حداقل طول صفر دارند.

- ① $\{a\}^+ / \{a\}^*$
- ② $\{a\}^+ / \{a\}^*$
- ③ $\{a\}^+ / \{a\}^*$
- ④ $\{a\}^+ / \{a\}^*$

MANSOOR

$$\{a, b\}^* = (a+b)^* = (a|b)^* = (a+b)^*$$

۳. $\{a\}^* \{b\}^*$ مجموعه رشته هایی را نشان می دهد که از تعدادی a (حداقل ۱) و به دنبال آن تعدادی b (حداقل صفر) تشکیل می شود.

۴. Σ^* همه رشته هایی را شامل می شود که از a و b تشکیل شده و طول ۱ یا ۲ دارد. طول صفر تا ∞

$$\Sigma^* = \{ \epsilon, a, b, aa, ab, ba, bb, \dots \}$$

مثال ۳: $\Sigma^* \{aa\}$ چه رشته هایی را نشان می دهد؟

۱. این مجموعه همان Σ^* است.

۲. این مجموعه همان Σ^+ است.

۳. رشته های این مجموعه قطعا به aa ختم می شوند.

۴. λ متعلق به این مجموعه از رشته هاست.

ولی

$$\Sigma^* = \{ \epsilon, a, b, aa, ab, ba, bb, \dots \}$$

ولی λ نمی شود

مثال ۴: با توجه به اینکه $\Sigma = \{0,1,2\}$ کدامیک از عبارات با قاعده زیر دارای یک صفر بوده و به هر تعداد می تواند ۲ یا ۳ به هر ترتیبی می توانند واقع شوند.

۱. $(12)^* 0 (12)^*$

۲. $0(1U2)^*$

۳. $(1U2)^* 0$

۴. $(1U2)^* 0 (1U2)^*$

کامل: $abab$
تکرار a و b

مثال ۵: اگر $\Sigma = \{a, b\}$ عبارت $(ab)^+ ab(a \cup b)^*$

۱. با قاعده است و کوتاهترین رشته این عبارت $abab$ است.

۲. با قاعده نیست و نمی توان رشته ای از آن استخراج کرد.

۳. با قاعده است و رشته ab را نشان می دهد.

۴. با قاعده است و مجموعه ای نامنظم از عناصر a را نشان می دهد.

مثال ۶: مجموعه رشته هایی به صورت کلی $\{a\}^* \{b\}^+$ تعریف می شوند، کدامیک از گزینه های زیر متعلق به این

مجموعه است:

۱. bbb

۲. $ababab$ ← یکی در میان است

۳. $aaaabbb$

۴. 3 و 1

گرامر

$$S \rightarrow asb$$

$$S \rightarrow ab$$

$$S \rightarrow \lambda$$

$$S \rightarrow \lambda$$

$$S \rightarrow ab$$

$$S \rightarrow asb$$

$$w = a^i b^j$$

$$w = a^i b^j$$

$$w = a^i b^j$$

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

منظم است

زمانی تولید می شود که فقط الفبا باشد

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

منظم است

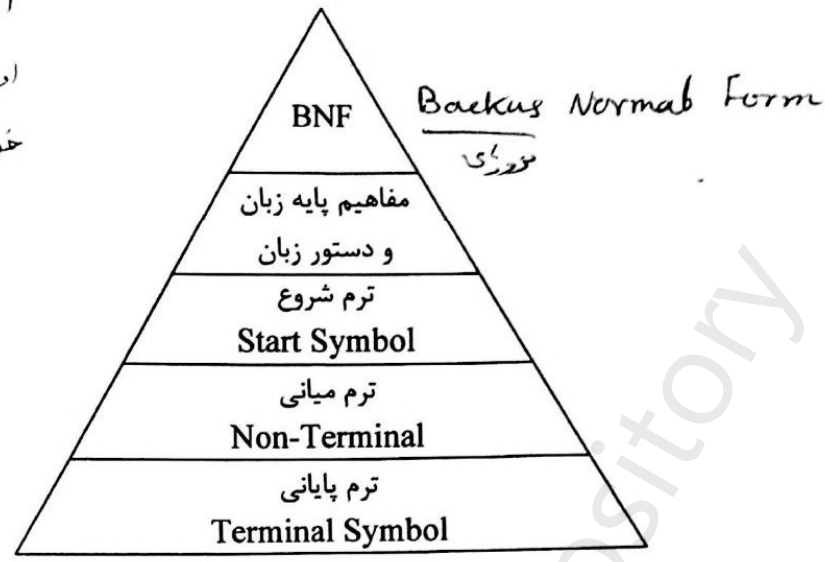
نظریه زبانها و ماشینها
علیرضا قوامی

تولید زبان
حدا پایه از اساس مشخص
شروع نیمه اول
Start Symbol
هر بار در هر بار
Symbol

product
عین اسم

زبان (گرامری)

زبان و دستور زبان



Back US Normal Form

یک زبان مجموعه ای است از جملات که هر جمله نیز خود مجموعه ای از لغات یا سمبولهاست. برای بیان قواعد زبانهای برنامه سازی از زبانهایی به نام *meta language* استفاده می گردد. به عنوان نمونه فعل، فاعل و مفعول لغاتی هستند که معمولا برای بیان دستور زبان فارسی بکار برده می شوند. به همین ترتیب زبانی خاص برای بیان فرم ظاهری و نه مفهومی زبانها ایجاد گردیده، این فرم یا زبان مافوق به *BNF* مشهور است که برای اولین بار در سال ۱۹۶۰ برای بیان دستور زبان الگول (*Algol*) به کار گرفته شده که شامل مجموعه ای از علائم به کار گرفته شده می باشد که تا امروز به صورت زبانی استاندارد برای توصیف فرم ظاهری گرامر زبانها به کار گرفته می شود. در اصطلاح *BNF* به هر سمبول (نماد یا لغات) یک ترم گفته می شود.

ترم ها به سه دسته تقسیم می شوند:

۱. ترم های پایانی (*Terminal Symbol*): یک ترم پایانی به ترمی گفته می شود که براساس ترمی دیگر در گرامر یا دستور زبان قابل تعریف نباشد.
۲. ترمهای میانی یا ترم های واسطه (*Non-Terminal*): به ترمهایی اطلاق می شود که قابل توضیح براساس یا حسب ترمهای دیگر باشند و یا ترمهای دیگر براساس آنها تعریف شده باشند به ترمهای میانی واسطه نیز می گویند، زیرا که واسطه ای برای بیان و مشخص نمودن قوانین گرامرها می باشد.
۳. ترم شروع یا سرترم (*Start Symbol*): هر دستور زبان یا گرامر شامل تنها یک سرترم می باشد. در اصطلاح *BNF* برای بیان گرامرها و قواعد آنها یکسری علائم استفاده می گردد. به مثال زیر توجه کنید:

$G_1: \langle \text{Sentence} \rangle \rightarrow \langle \text{Verb} \rangle | \langle \text{Subject} \rangle$ I قاعده
 $\langle \text{Verb} \rangle ::= \text{go} | \text{Come}$ II قاعده
 $\langle \text{Subject} \rangle ::= I | \text{You}$ III قاعده
 هست \rightarrow به معنی $::=$ یا \rightarrow

یا "۱"
 ع-۱: هجی
 ترمینال

MANSOOR

همانگونه که مشاهده می شود هر گرامر از یکسری قواعد تشکیل یافته (سه قاعده) در گرامر فوق *Sentence* سرترم (*Start Symbol*) گرامر می باشد. $\langle Verb \rangle$ و $\langle Subject \rangle$ ترمهای میانی گرامر و *I, You, go, Come* ترمهای پایانی اند. ترمهای میانی بین دو علامت $\langle \rangle$ قرار گرفته اند. علامت \rightarrow یا $=$: به مفهوم « هست » می باشد. علامت $|$ به مفهوم یا می باشد. به طور کلی برای تفکیک ترمهای پایانی از میانی و سرترم گرامر از بقیه ترمها معمولا یا فرم بکار رفته شده دو مثال فوق به کار می رود و یا اینکه:

۱. سرترم معمولا با علامت S مشخص می شوند.
۲. ترمهای میانی معمولا با حروف بزرگ مشخص می شوند.
۳. ترمهای پایانی معمولا با حروف کوچک مشخص می شوند.

تعریف گرامر زبان

هر زبان توسط مکانیزمی به نام گرامر توصیف می شود که خود براساس چهار پارامتر $\langle S, T, V, P \rangle$ تعریف می شود و به صورت $G: \langle S, T, V, P \rangle$ نشان داده می شود. به طوری که G بیانگر گرامر زبان و T یا Σ بیانگر الفبای زبان و V مجموعه ای متناهی از عناصر بنام ترمهای میانی و S عنصری خاص از V بنام ترم شروع و P *Product* که قاعده یا *Rule* نامیده می شود مجموعه ای متناهی است از قوانین جایگزینی مانند: $u \rightarrow v$

به عنوان نمونه برای گرامر روبرو می توان نوشت:

<p>Non Terminal ساخته شده Terminal ترم شروع</p>	$\left. \begin{array}{l} S \rightarrow AMP \\ A \rightarrow aA \\ M \rightarrow mM \\ P \rightarrow b \end{array} \right\}$	$V_N = \{A, M, P\}$ $T = \Sigma = \{a, b, m\}$	<p style="text-align: right;">product</p> $G = \langle S, V_N, V_T, P \rangle$ <p style="text-align: center;"> \downarrow start symbol vocabulary vocabulary Non-Terminal Terminal </p>
---	---	--	---

اگر بتوان با اعمال یک یا چند قانون به رشته خاص رسید، نگاه رشته مورد نظر متعلق به زبان است و اگر رشته ای متعلق به زبان باشد قطعا می توان با شروع از S و اعمال یک یا چند عدد از قوانین به آن جمله رسید.

نکته: برای هر زبان حداقل یک گرامر می توان نوشت.

نکته: هر گرامر تمامی جملات زبان را تولید می کند، نه بیشتر و نه کمتر.

طبقه بندی زبانها (سلسله مراتب چامسکی)

زبانها براساس قوانین بیانگر گرامر آنها به چهار طبقه تقسیم می گردد و چنانچه برای یک زبان حداقل یک گرامر از یک طبقه وجود داشته باشد زبان متعلق به آن طبقه است.

۱. زبانهای با قاعده (Regular Language)

قوانین گرامر در این زبان به صورت زیر می باشد:

$$\begin{aligned} A &\rightarrow a & A \in V, a \in \Sigma \\ A &\rightarrow aA & A, A' \in V, a \in \Sigma \\ S &\rightarrow \lambda & (\lambda \in L \text{ اگر و فقط اگر}) \end{aligned}$$

۲. زبانهای مستقل از متن، فارغ از متن و یا مستقل از زمینه (Content Free Language)

قوانین گرامر در این زبان به صورت زیر می باشد:

$$\begin{aligned} A &\rightarrow u & A \in V, u \in (V \cup \Sigma)^+ \\ S &\rightarrow \lambda & (\lambda \in L \text{ اگر و فقط اگر}) \end{aligned}$$

۳. زبانهای وابسته به متن یا وابسته به زمینه (Content Sensitive Language)

$$\begin{aligned} u &\rightarrow v & u \in (V \cup \Sigma)^+ & \text{رابطه} & v \in (V \cup \Sigma)^+ \\ & & & \text{همچنین} & & \text{مثال: } S \rightarrow \alpha \beta \alpha \\ & & & & & \text{مثال: } uv \rightarrow v^k u \\ & & & & & \text{مثال: } a^n \end{aligned}$$

$$\text{Length}(u) \leq \text{Length}(v)$$

مثال: $S \rightarrow \alpha \beta \alpha$ با n تکرار α و β یکسان باشد.

تذکر: برای زبانهای با قاعده و زبانهای مستقل از متن رابطه زیر برقرار است:

زبانهای وابسته به متن \subset زبانهای مستقل از متن
 $\phi =$ زبانهای وابسته به متن \cap زبان مستقل از متن

۴. زبانهای بدون محدودیت

قوانین گرامر در این زبانها بصورت زیر می باشد:

$$\begin{aligned} u &\rightarrow v & u \in (V \cup \Sigma)^+ \\ & & v \in (V \cup \Sigma)^+ \end{aligned}$$

در نتیجه می توان نوشت:

زبانهای بدون محدودیت \subset زبانهای مستقل از متن \subset بانهای با قاعده

و نیز:

زبانهای بدون محدودیت \subset زبانهای وابسته به متن

نکته: هر زبانی یک زبان بدون محدودیت است.

مثال ۱: فرم کلی زبان گرامر روبه رو کدامست؟

$$S \rightarrow AaaA$$

$$A \rightarrow aA$$

$$A \rightarrow bA$$

$$A \rightarrow cA$$

$$A \rightarrow \lambda$$

$$.1 \quad a^n b^n c^n$$

$$.2 \quad b^n aac^n$$

$$.3 \quad (a \cup c)^* aa (b \cup c)^*$$

$$.4 \quad (a \cup b \cup c)^* aa (a \cup b \cup c)^*$$

مثال ۲: گرامر رو برو جزو کدامیک از طبقه هاست؟ ✓

$$S \rightarrow ASBB$$

$$S \rightarrow abb$$

$$B \rightarrow b$$

$$A \rightarrow a$$

.1 مستقل از متن ✓

.2 وابسته به متن ✓

.3 بدون محدودیت

.4 ۳ و ۱ ✓

مثال ۳: در گرامر زیر کدامیک از قوانین را حذف نماییم تا گرامر وابسته به متن گردد؟ ✓

$$S \rightarrow a$$

$$S \rightarrow bSBC$$

$$T \rightarrow bTB$$

$$TCB \rightarrow bT$$

$$CT \rightarrow CB$$

$$T \rightarrow C$$

$$B \rightarrow C$$

$$B \rightarrow bB$$

$$.1 \quad T \rightarrow CB$$

$$.2 \quad TCB \rightarrow bT, CT \rightarrow CB \leftarrow \text{بجایگزینی}$$

$$.3 \quad TCB \rightarrow bT \checkmark$$

$$.4 \quad \text{هیچکدام}$$

مثال ۴: یک گرامر دارای قانونی بصورت $Aa \rightarrow a$ است کدامیک صحیح است؟

۱. این گرامر یک گرامر با قاعده است.

۲. مستقل از متن است.

۳. وابسته به متن است.

۴. بدون محدودیت است. ✓

راست
بدون محدودیت

مثال ۵: گرامر مستقل از متنی بنویسید که زبان زیر را تولید کند.

$$L = \{a^i b^j \mid i = 0, 1, 2, \dots \quad j = 0, 1, 2, \dots \quad j \geq i\}$$

جواب:

$$\begin{cases} S \rightarrow aSb \\ S \rightarrow Sb \\ S \rightarrow \lambda \end{cases}$$

گرامرهای با قاعده گرامرهایی هستند که یا خود بازگشتی چپ دارند یا خود بازگشتی راست.

$$A \rightarrow aA$$

خود بازگشتی راست

$$A \rightarrow Aa$$

خود بازگشتی چپ

اشتقاق (Derivation)

Derivation

اشتقاق: دنباله ای از قوانین جایگزینی که جمله را از نماد آغازگر تولید می نماید اشتقاق نامیده می شود. به عبارت (عمودی) دیگر اشتقاق مکانیزم تولید جملات زبان از گرامر است. (مکانیزم تولید زبان (گرامر))

شبه جمله: عبارتهای موجود در عمل اشتقاق را شبه جمله می گویند. (Sentencial form)

جمله: شبه جمله ای که فقط از عناصر الفبا تشکیل شده باشد. (sentence)

زنجیره اشتقاق: دنباله قوانین جایگزینی یک زنجیره اشتقاق نامیده می شود. (Derivation chain)

نکته: برای تولید یک جمله از زبان ممکن است زنجیره های اشتقاق متفاوتی وجود داشته باشد.

نکته: در زبانهای مستقل از متن در مراحل اشتقاق متغیرهای یک شبه جمله مستقل از یکدیگر جایگزین می شوند.

روشهای اشتقاق

1. اشتقاق چپ (چپ ترین) (Left Most Derivation)

سمت چپ ترین متغیر برای جایگزینی انتخاب می شود.

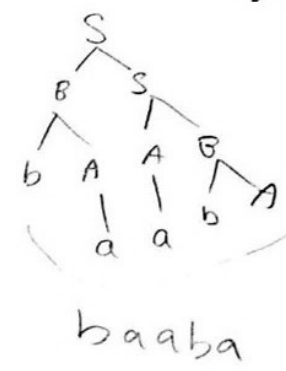
2. اشتقاق راست (راست ترین) (Right Most Derivation)

سمت راست ترین متغیر برای جایگزینی انتخاب می شود.

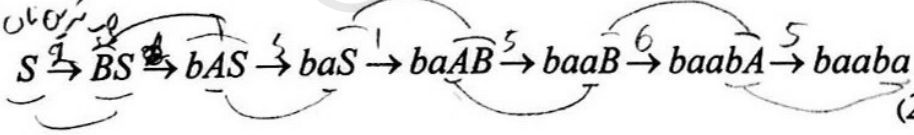
نکته: برای اثبات اینکه جمله ای به زبان تعلق دارد کافیست که جمله ای از S توسط اشتقاق تولید شود.

به عنوان مثال در گرامر مستقل از متن زیر جمله baaba با استفاده از اشتقاق چپ و اشتقاق راست به صورت زیر تولید می شود.

- 1) $S \rightarrow AB$
- 2) $S \rightarrow BS$
- 3) $A \rightarrow aA$
- 4) $B \rightarrow bB$
- 5) $A \rightarrow a$
- 6) $B \rightarrow bA$
- 7) $S \rightarrow \lambda$

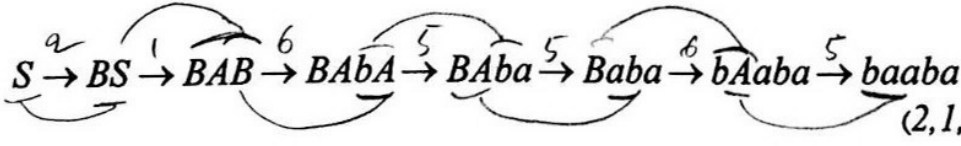


سمت چپ ترین



مراحل اشتقاق: (2,6,5,1,5,6,5)

سمت راست ترین:



مراحل اشتقاق: (2,1,6,5,5,6,5)

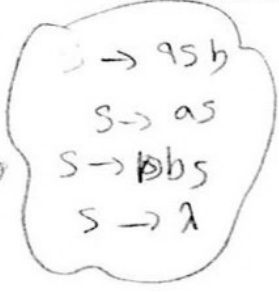
نکته: در تولید زبان از گرامر (مراحل اشتقاق) عدد زیر خط صحیح گویا محدودیت ندارد. 1- ترتیب 2- تکرار 3- اولیت

درخت اشتقاق چپ و راست

اگر برای گسترش همواره سمت چپ ترین گره از درخت مستقل از سطح (level) انتخاب شود درخت حاصل اشتقاق چپ و اگر همواره سمت راست ترین انتخاب شود درخت حاصل اشتقاق راست نامیده می شود.

گرامر مبهم (Ambiguous) هرگاه بتوان از یک گرامر یک جمله را در درخت گسترش متفاوت تولید نمود که امر را مبهم می گویند. اگر برای حداقل یک جمله از زبان تولید شده بوسیله گرامری بیشتر از یک درخت اشتقاق وجود داشته باشد، در این صورت گرامر مبهم نامیده می شود. به عنوان مثال دو درخت اشتقاق زیر برای تولید جمله $\omega = ab$ از گرامر

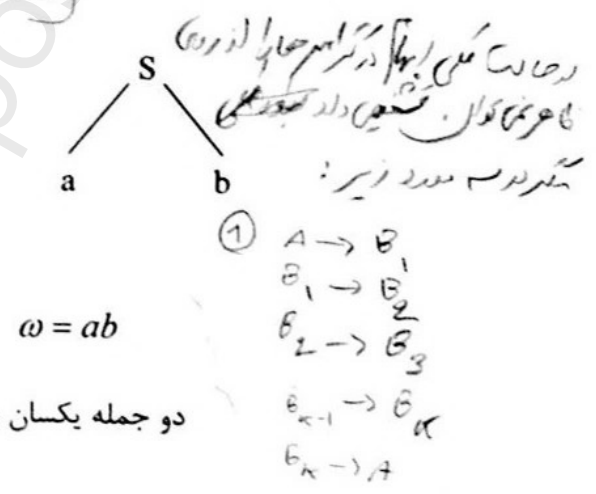
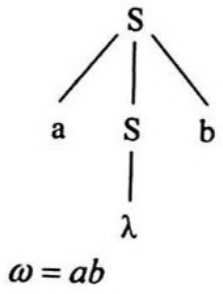
- $S \rightarrow aSb$
- $S \rightarrow ab$
- $S \rightarrow \lambda$



- ② $A \rightarrow Ax \mid \alpha A$
 - ③ $A \rightarrow A\alpha \mid \alpha A\beta$
- self-embeding
خودگردان

متصور است.

می توان با دستکاری گرامر
لذا در این روش گرامر را



دو جمله یکسان با دو درخت متفاوت با یک گرامر = گرامر مبهم

نکته: تعریف گرامر مبهم در مورد زبانهای مستقل از متن نیز صادق است.

نکته: فقط در زبانهای با قاعده می توان در مورد اینکه در هر گسترش از چه قانونی استفاده می شود، صراحتاً سخن گفت.

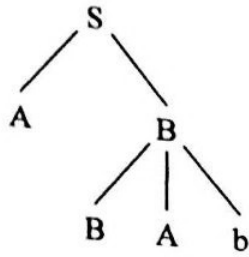
مثال 1: در گرامر مستقل از متن زیر کدامیک از مراحل اشتقاق جمله $aabbbb$ را تولید می کند؟

- 1) $S \rightarrow ASBB$
- 2) $S \rightarrow abb$
- 3) $B \rightarrow b$
- 4) $A \rightarrow a$

- 1. (1,3,2,4,3)
- 2. (1,4,2,3,3)
- 3. (1,3,3,2,4)
- 4. هر سه

هرگاه دو گرامر غیر یکسان و متفاوت یکسان تولید کنند گرامرهای معادل Equivalent می گویند

مثال ۲: بخشی از یک درخت اشتقاق حاصل از یک گرامر در شکل زیر آمده است، کدامیک از قوانین زیر قطعاً در این گرامر وجود دارد؟



- ۱. $S \rightarrow A$
- ۲. $B \rightarrow A$
- ۳. $B \rightarrow Bb$
- ۴. ۱ و ۲

مثال ۳: کدام گزینه یک گرامر مبهم را نشان می دهد؟

- ۱. $S \rightarrow \lambda$
- ۲. $S \rightarrow SS$
- ۳. $S \rightarrow ab$

- ۱) $S \rightarrow T$
- $S \rightarrow Sa$
- ۱. $S \rightarrow a$
- $T \rightarrow ab$
- $T \rightarrow \lambda$

- $S \rightarrow \lambda$
- $S \rightarrow SB$
- ۳. $S \rightarrow a$
- $A \rightarrow b$
- $T \rightarrow ab$

۴. ۱ و ۲

تمرین ۱: گرامر نوع دومی بنویسید که زبان زیر را تولید کند؟

$$L = \{a^i b^j a^i b^j \mid i = 0, 1, 2, \dots \quad j = 0, 1, 2, \dots\}$$

جواب:

- $S \rightarrow aSb$
- $S \rightarrow B$
- $B \rightarrow bBa$
- $B \rightarrow \lambda$

$$S \rightarrow aSb \rightarrow a^{10} S b^{10} \rightarrow a^{10} B b^{10} \rightarrow a^{10} b^{100} a^{100} b^{10}$$

تمرین ۲: گرامر نوع دومی بنویسید که زبان زیر را تولید کند؟

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

جواب:

$$S \rightarrow SC$$

$$S \rightarrow AC$$

$$A \rightarrow aAb$$

$$A \rightarrow ab$$

تمرین ۳: گرامری از نوع دوم بنویسید که زبان زیر را تولید کند؟

$$L = \{a^k b^m c^n \mid k = m, \text{ or } n = m\}$$

جواب:

$$S \rightarrow A \mid B$$

$$A \rightarrow Ac$$

$$A \rightarrow T$$

$$T \rightarrow aTb$$

$$T \rightarrow \lambda$$

$$B \rightarrow aB$$

$$B \rightarrow R$$

$$R \rightarrow bRC$$

$$R \rightarrow \lambda$$

تمرین ۴: گرامری از نوع دوم بنویسید که زبان زیر را تولید کند؟

$$L = \{a^i b^j c^{i+j} \mid i = 0, 1, 2, \dots \quad j = 0, 1, 2, \dots\}$$

جواب:

$$S \rightarrow a$$

$$S \rightarrow D$$

$$D \rightarrow bDc$$

$$D \rightarrow \lambda$$

الگوریتم پارس (Pars)

الگوریتمی که با شبیه سازی اشتقاق چپ اقدام به تولید زبان بنماید. پیشوند جمله (*Prefix*): α پیشوند جمله ω است به شرطی که $\omega = \alpha_1\alpha_2$ باشد. به عنوان مثال در جمله $\omega = \bar{a}abb$ مجموعه *Prefix* پیشوند های جمله ω را نشان می دهد.

$$prefix = \{\lambda, a, aa, aab, aabb\}$$

نکته: هر جمله به طول n دارای $n+1$ پیشوند است.

پیشوند شبه جمله: α_1 پیشوند شبه جمله u است، اگر $u = \alpha_1A\alpha_2$ و α_1 متعلق به Σ^* است. به عنوان مثال برای شبه جمله های زیر می توان گفت:

$$u = AaB \Rightarrow prefix(u) = \{\lambda\}$$

$$u = aaAaB \Rightarrow prefix(u) = aa$$

نکته: هر شبه جمله فقط یک پیشوند دارد.

پسوند شبه جمله: α_2 پسوند شبه جمله u است، اگر $u = \alpha_1\alpha_2$ و α_1, α_2 متعلق به $(\Sigma \cup V)^*$.

الگوریتم پارسر (تجزیه کننده)

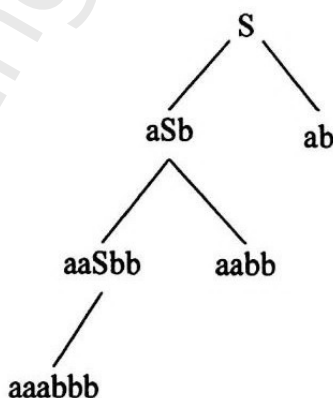
الگوریتم تولید کننده جملات زبان و تشخیص تعلق جملات به زبان پارسر نامیده می شود.

گراف پارس

ساختاری که پارسر برای تولید جملات زبان و یا تشخیص تعلق یک جمله به زبان تولید می نماید گراف پارس نامیده می شود. در گراف پارس گره ورودی گراف (نماد آغازگر) هر گره از گراف یک شبه جمله و گره های پایانی گراف جملات زبان و هر لبه از گراف معرف قانونی است که جهت جایگزینی استفاده می شود. اگر در طی تولید گراف پارس جمله مورد نظر در گراف ظاهر شود، در این صورت جمله متعلق به زبان است و اگر عملیات تولید گراف متوقف شود و جمله مورد نظر ظاهر نشود جمله متعلق به زبان نیست. به عنوان مثال در گراف پارس حاصل از گرامر زیر مشاهده می شود که:

$$S \rightarrow aSb$$

$$S \rightarrow ab$$



گراف پارس

به عنوان مثال در گراف پارس بالا $aaabbb, aabb, ab$ متعلق به زبان هستند.

عبارت (A, a, ab, abA) را بنویسید

تولید کننده (ab)
تکراری

$w = abA$

Non Terminal

پارسرهای بالا به پائین

الگوریتم پارس ۱ (تولید کننده گراف در پهنا)

در این الگوریتم با برداشتن هر شبه جمله از صف کلیه شبه جمله هایی که براساس قوانین موجود توسط شبه جمله فعلی قابل تولید هستند در صف قرار می گیرند. این عملیات با قرار دادن نماد آغازگر S در صف به عنوان اولین عنصر آغاز می شود و تا خالی شدن و یا پیدا شدن جمله مورد نظر ادامه می یابد.

نکته: در مواردی ممکن است در مورد جملاتی که به زبان تعلق ندارند الگوریتم در حلقه نامتناهی قرار گیرد.

نکته: اگر الگوریتم پارس ۱ متوقف شود جمله ممکن است به زبان متعلق باشد و یا نباشد (بسته به اینکه رشد گراف به اتمام رسیده و یا به دلیل پیدا کردن جمله مورد نظر توقف صورت گرفته باشد). ولی اگر متوقف نشود جمله متعلق به زبان نیست.

نکته: از شرط پیشوند برای توقف رشد گراف استفاده می شود، یعنی گره ای از گراف که پیشوند مشترک با جمله مورد آزمایش دارد فقط گسترش می یابد.

مثال: تعلق جمله aab به زبان گرامر زیر را با استفاده از الگوریتم پارس ۱ بررسی کنید.

1) $S \rightarrow AB$

2) $A \rightarrow aa$

3) $B \rightarrow b$

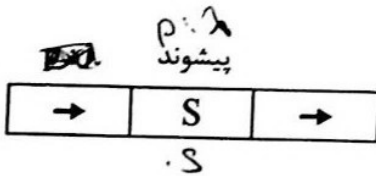
4) $A \rightarrow a$

$w = aab$

$S \rightarrow AB \rightarrow aAB \rightarrow aab \checkmark$ *تعلق دارد*

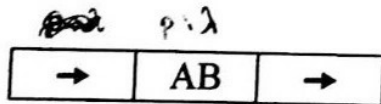
جواب:

۱.



S

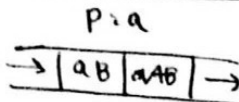
۲.



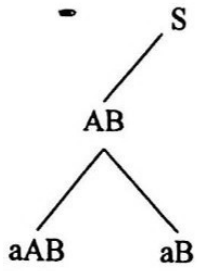
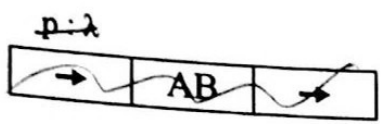
در جای S که است

همه گره ها را در صف قرار می دهیم

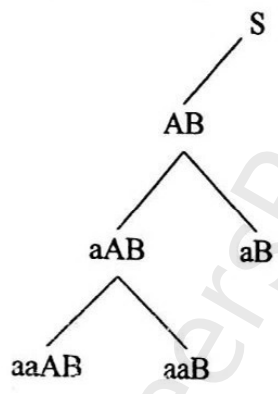
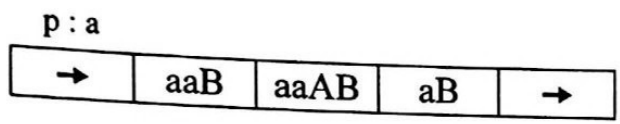
صورت دیگر (معمولی)



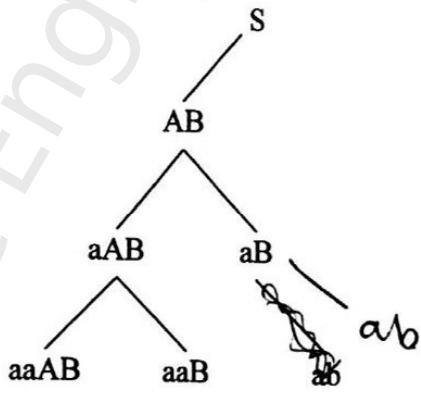
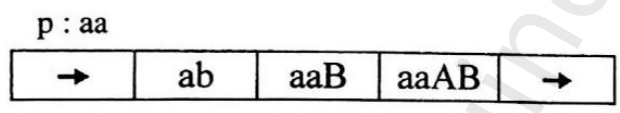
۳



۴

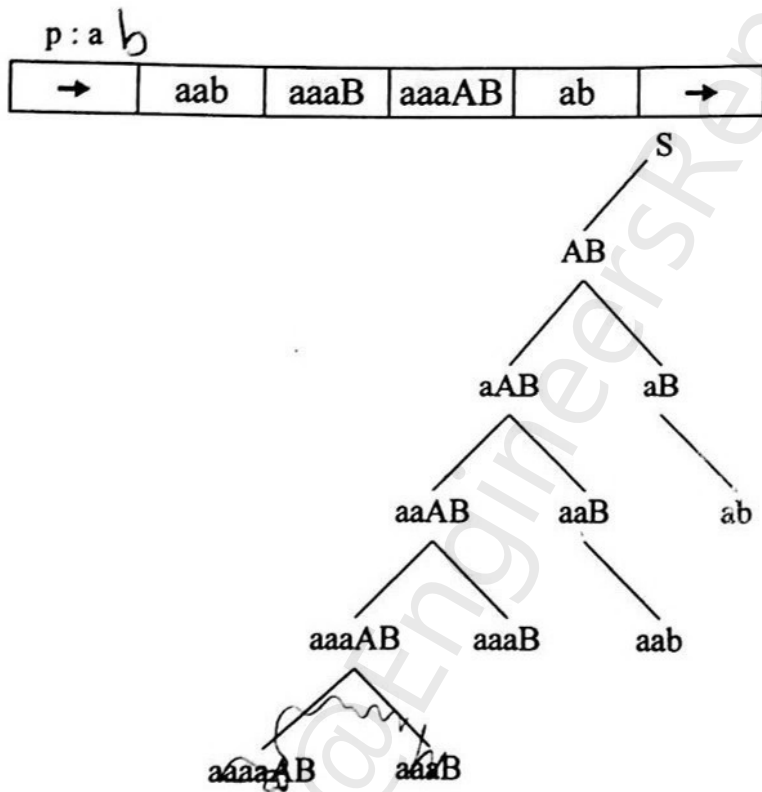
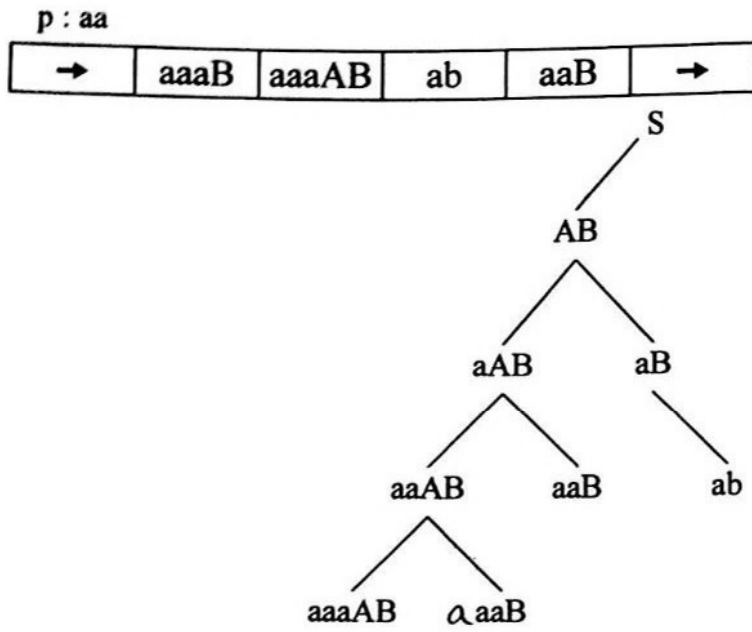


۵



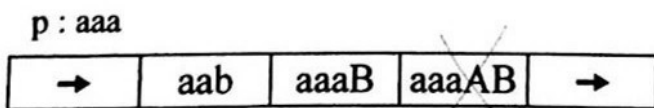
MANSOOR

۶



۷

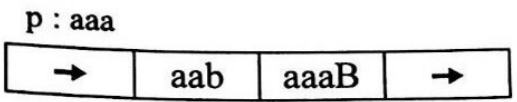
۸



گراف بدون تغییر

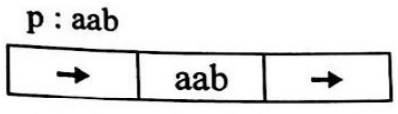
انرا افصح من

۹



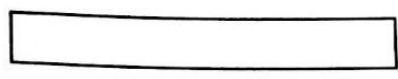
گراف بدون تغییر

۱۰



گراف بدون تغییر

۱۱



گراف بدون تغییر

الگوریتم پارس ۲ (تولید کننده گراف عمق)

در این الگوریتم به جای صف، از پشته استفاده می شود و علاوه بر ذخیره خود شبه جمله در پشته مجموعه قوانینی که برای گسترش قابل استفاده هستند، ذخیره می شود و هر شبه جمله که در پشته قرار گیرد و به دنبال آن شبه جمله ای براساس یکی از قوانین ذخیره شده قابل گسترش هستند، روی پشته قرار می گیرند. هر شبه جمله ای که غیر قابل گسترش تشخیص داده شد، (براساس قوانین باقی مانده و یا پیشوند شبه جمله) از پشته برداشته می شود، به این ترتیب هر شاخه گراف با استفاده از اشتقاق چپ، در عمق پیش می رود و بعد از متوقف شدن رشد یک شاخه در شاخه کناری پیش می رود. به این ترتیب گراف گسترش می یابد. شروع عملیات با قرار دادن k به عنوان نماد آغازگر و کلیه قوانین قابل استفاده در پشته صورت می گیرد.

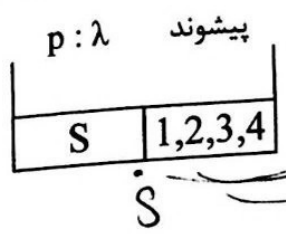
نکته: اگر الگوریتم پارس ۲ متوقف شود جمله می تواند به زبان متعلق باشد و یا نباشد، ولی اگر متوقف نشود در مورد تعلق جمله به زبان هیچ چیز نمی توان گفت.

نکته: از شرط پیشوند برای گسترش بیشتر و یا توقف گسترش در هر شاخه استفاده می شود.

مثال: تعلق جمله aab را به زبان گرامر زیر با استفاده از الگوریتم پارس ۲ بررسی کنید.

- 1) $S \rightarrow AB$
- 2) $A \rightarrow aA$
- 3) $B \rightarrow b$
- 4) $A \rightarrow a$

جواب:
۱

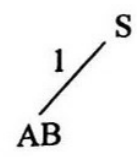


S

.۲

p : λ

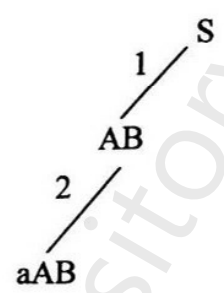
AB	1,2,3,4
S	2,3,4



.۳

p : a

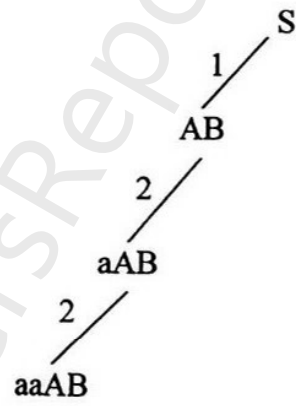
aAB	1,2,3,4
AB	1,3,4
S	2,3,4



.۴

p : aa

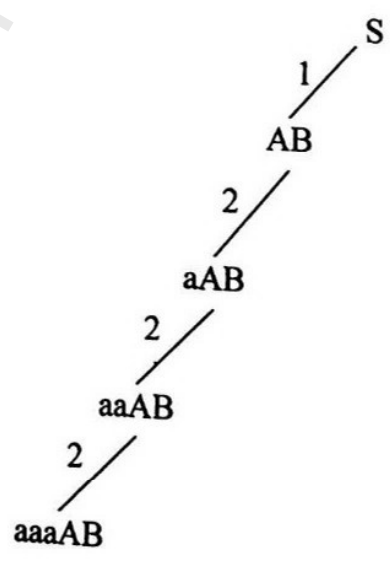
aaAB	1,2,3,4
aAB	1,2,3
AB	1,3,4
S	2,3,4



.۵

p : aaa

aaaB	1,2,3,4
aaAB	1,2,4
aAB	1,2,3,4
AB	1,3,4
S	2,3,4



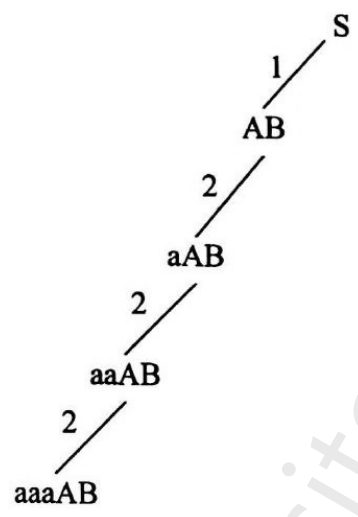
عدد سلام را با کمترین عدد
عبارت با حذف متناهی

بهرد نمی شود در صورت
تکرار a دلید
طبقه نام کامل حذف متناهی

۶

p : aa

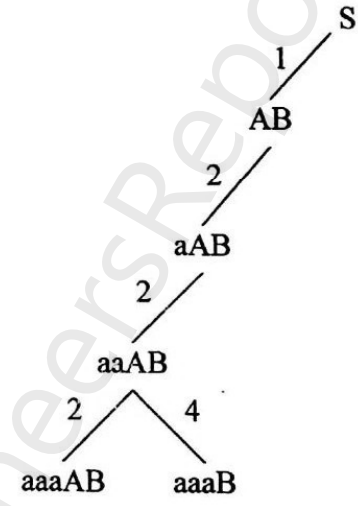
aaAB	1,3,4
aAB	1,3,4
AB	1,3,4
S	2,3,4



۷

p : aa_a

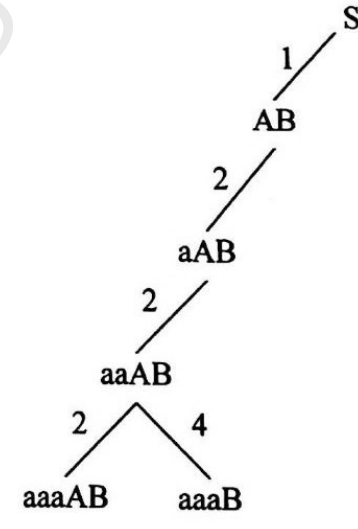
aaaAB	1,2,3,4
aaAB	1,3,4
aAB	1,3,4
AB	1,3,4
S	2,3,4



۸

p : aa

aaAB	1,3
aAB	1,3,4
AB	1,3,4
S	2,3,4



گراف علی

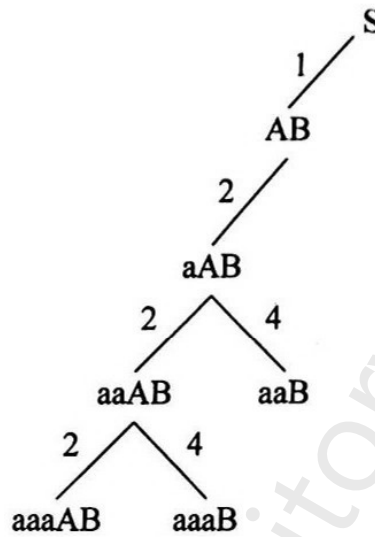
MANSOOR

بہر در تمام ضروری
یک طبقہ یا سہ ماہی

۹

p : aa

aaB	1,2,3,4
aAB	1,3
AB	1,3,4
S	2,3,4

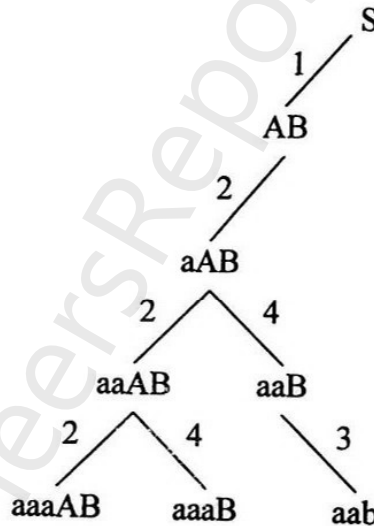


۱۰

مکمل بودن
aab

p : aab

aab	1,2,3,4
aaB	1,2,4
aAB	1,3
AB	1,3,4
S	2,3,4



پارسرهای پائین به بالا

از این پارسرها برای حل مشکل قرار گرفتن الگوریتم های پارس در حلقه نامتناهی استفاده می شود. در این روش گراف پارس از جمله ورودی تولید می شود. قوانین گرامر در این روش بصورت معکوس استفاده می شوند و زمانی می توان گفت جمله متعلق به زبان است که نماد آغازگر به عنوان یک گره از گراف به وجود آمده باشد.

الگوریتم پارس ۳ (تولید کننده گراف در سطح)

این الگوریتم نیز مانند الگوریتم پارس ۱ براساس صف کار می کند. در اینجا جمله مورد نظر در ابتدا در صف قرار می گیرد. هر شبه جمله که از ابتدای صف برداشته می شود، در عوض کلیه حالت های ممکن کاهش روی زیر رشته های شبه جمله در انتهای صف اضافه می شود تا وقتی که یا صف خالی شود و یا نماد آغازگر در صف تولید شود که در صورت تولید نماد آغازگر جمله ابتدایی متعلق به زبان است.

نکته: در صورتی که جمله ورودی متعلق به زبان نباشد و $\lambda \rightarrow S$ یکی از قوانین گرامر باشد قطعاً الگوریتم در حلقه نامتناهی قرار می گیرد که البته در صورتی که قانون یاد شده فقط برای تولید λ بکار رود می توان آن را نادیده گرفت تا الگوریتم در حلقه نامتناهی واقع نشود.

الگوریتم پارس ۴ (تولید کننده گراف در عمق)

در این الگوریتم مانند الگوریتم پارس ۲ از پشته استفاده می شود. در این الگوریتم عنصر به عنصر از سمت چپ جمله ورودی خوانده می شود و در صورت وجود پسوند قابل کاهش در بخش خوانده شده عملیات کاهش صورت می گیرد و دوباره وجود و یا عدم وجود بخش قابل کاهش بررسی می شود. در صورت عدم وجود بخش کاهش عنصر بعدی خوانده می شود. آنچه مهم است، اینست که برای کاهش قوانین مختلفی احتمالاً قابل استفاده خواهد بود که همه باید امتحان شوند به همین دلیل قبل از اعمال هر قانون مجموعه قوانین دیگر که امتحان نشده اند، به همراه باقیمانده جمله ورودی و بخش کاهش یافته جدا شده از ورودی برای بررسی بعدی در پشته ذخیره می شوند. این عملیات ادامه می یابد تا باقیمانده جمله ورودی به λ برسد و بخش دیگر همان جمله آغازگر می باشد. در مورد هر رکورد که از پشته برداشته می شود، قابلیت کاهش بررسی می شود و در نهایت (تمام شدن ورودی و یا بررسی همه قوانین ممکن) از رکورد صرف نظر می شود. در هر مرحله اعمال یکی از قوانین که در موقع کاهش صورت می گیرد یک شاخه به گراف و نیز یک رکورد جدید به بالای پشته اضافه می شود.

نکته: اگر $\lambda \rightarrow S$ جزء قوانین باشد ولی جمله به زبان متعلق نباشد این الگوریتم قطعاً در حلقه نامتناهی قرار می گیرد و حتی اگر جمله به زبان متعلق هم باشد الگوریتم ممکن است در حلقه نامتناهی قرار گیرد.

مثال: در گرامر زیر جمله $bbaa$ در الگوریتم پارس پائین به بالا تولید کننده گراف در عمق داده می شود.

- $S \rightarrow \lambda$
- $S \rightarrow AB$
- $A \rightarrow Aa|a$
- $B \rightarrow b$

۱. الگوریتم قطعاً در حلقه نامتناهی قرار می گیرد.
۲. الگوریتم ممکن است در حلقه نامتناهی قرار گیرد.
۳. الگوریتم قطعاً در حلقه نامتناهی قرار نمی گیرد.
۴. با حذف قانون سوم گرامر الگوریتم در حلقه نامتناهی قرار نمی گیرد.

مثال: یک الگوریتم پارس هنگام بررسی تعلق یک جمله به زبان در حلقه نامتناهی قرار گرفته است. اگر بدانیم جمله متعلق به زبان است.

۱. این الگوریتم می تواند هر نوع پارس بالا به پائین باشد.
۲. این الگوریتم پارس بالا به پائین تولید کننده گراف در پهنا نمی باشد.
۳. این الگوریتم می تواند پارس پائین به بالا تولید کننده گراف در عمق باشد.
۴. ۲ و ۳

مثال: کدامیک از محارفات زیر یک بیستونند برای حجمه 5000 می باشد؟

- ۱. 10
- ۲. 5
- ۳. 5000
- ۴. هر سه

مثال: کدامیک از محارفات زیر بیستونند برای شبه حجمه 540 می باشد؟

- ۱. 40
- ۲. 540
- ۳. 4
- ۴. 3

@EngineersRepository

تولیدات واحد یا unit-production

هر قانونی به صورت $A \rightarrow B$ یک تولید واحد نامیده می شود.

متغیر غیر مفید یا useless variable

متغیری که در تولید جملات زبان نقشی نداشته باشد، متغیر غیر مفید نامیده می شود و به عنوان مثال در گرامر زیر متغیرهای M, N, F غیر مفید هستند:

- $S \rightarrow \lambda$
- $S \rightarrow AB$
- $S \rightarrow MN$
- $B \rightarrow b$
- $A \rightarrow Aa|a$
- $F \rightarrow f$

متغیر بازگشتی چپ

اگر یک متغیر بازگشتی طی یک یا چند مرحله خود را به عنوان سمت چپ ترین عنصر شبه جمله تولید نماید، متغیر بازگشتی چپ نامیده می شود.

$A \xrightarrow{*} A...$

متغیر بازگشتی چپ مستقیم و غیر مستقیم

اگر یک متغیر بازگشتی چپ در یک گرامر طی یک مرحله خود را تولید کند به آن مستقیم و در غیر این صورت غیر مستقیم نامیده می شود.

MANSOOR

<p>مثال عمومی و یکبار واحد از گرامر نیم حذف کنند</p> <p>$S \rightarrow D$</p> <p>$S \rightarrow ABD$</p> <p>$S \rightarrow MN$</p> <p>$S \rightarrow B Aa A$</p> <p>$B \rightarrow M b$</p> <p>$N \rightarrow D nN n$</p> <p>$D \rightarrow dD d$</p> <p>$M \rightarrow m$</p>	<p>حذف تولیدات واحد</p> <p>①</p> <p>$S \rightarrow D$</p> <p>$A \rightarrow B$</p> <p>$A \rightarrow A$</p> <p>$B \rightarrow M$</p> <p>$N \rightarrow D$</p> <hr/> <p>$A \rightarrow M$</p>	<p>حذف تولیدات واحد از گرامر</p> <p>②</p> <p>$S \rightarrow ABD$</p> <p>$S \rightarrow MN$</p> <p>$A \rightarrow Aa$</p> <p>$B \rightarrow b$</p> <p>$N \rightarrow nN n$</p> <p>$D \rightarrow dD d$</p> <p>$n \rightarrow m$</p>	<p>③</p> <p>$S \rightarrow dD d$</p> <p>$A \rightarrow b$</p> <p>$A \rightarrow An$</p> <p>$B \rightarrow m$</p> <p>$N \rightarrow dD d$</p> <p>$A \rightarrow m$</p>
--	--	---	---

④ از ② و ③

- $S \rightarrow AaD|MN|dD|d$
- $A \rightarrow Aa|b|m$
- $B \rightarrow b|m$
- $N \rightarrow nN|n|dD|d$
- $D \rightarrow dD|d$
- $N \rightarrow m$

برای حذف

مراحل تولید گرامر نرمال از یک گرامر مستقل از متن

۱. حذف نماد آغازگر بازگشتی
۲. حذف قوانین λ
۳. حذف تولیدات واحد
۴. حذف متغیرهای غیر مفید
۵. حذف بازگشتی های چپ

۱- حذف نماد آغازگر بازگشتی

اگر در یک گرامر $G = (\Sigma, V, S, P)$ نماد آغازگر S بازگشتی باشد، اگر در این صورت گرامر G' وجود دارد. به طوری که نماد آغازگر G' بازگشتی نباشد و داشته باشیم:

$$L(G) = L(G')$$

در این حالت G' به صورت زیر تعریف می شود:

$$G' = \{\Sigma, V \cup \{S'\}, R \cup \{S' \rightarrow S\}, S'\}$$

تبدیل زیر مثالی از حذف نماد آغازگر بازگشتی است

$$G: \begin{cases} S \rightarrow aSb \\ S \rightarrow \lambda \end{cases} \rightarrow G: \begin{cases} S' \rightarrow S \\ S \rightarrow aSb \\ S \rightarrow \lambda \end{cases}$$

۲- حذف قوانین λ

بعد از شناسایی متغیرهای λ باید شبه جمله هایی را که از طریق قوانین λ بدست می آیند، توسط قوانین گرامر تولید شوند تا امکان حذف قوانین λ فراهم آید. (در صورتیکه λ متعلق به زبان باشد جمله $S \rightarrow \lambda$ نباید حذف شود) به عنوان مثال قوانین λ در گرامر زیر طی مراحل ذیل حذف می شوند.

$$\begin{aligned} \text{nulllabel} &= \{S, A, B, D\} \\ S \rightarrow ABD &\Rightarrow S \rightarrow ABD | BD | AD | AB | D | B | A \\ A \rightarrow Aa | \lambda &\Rightarrow A \rightarrow Aa | a \\ B \rightarrow Bb | \lambda &\Rightarrow B \rightarrow Bb | b \\ D \rightarrow Db | \lambda &\Rightarrow D \rightarrow Db | b \end{aligned}$$

مثال: گرامر مستقل از متن بدون تولید λ متناظر با گرامر زیر را بنویسید.

$$\text{nullable} = \{B, C, A\}$$

$$\begin{aligned} S \rightarrow ABaC &\Rightarrow S \rightarrow ABaC \mid AaC \mid ABa \mid BaC \mid aC \mid Ba \mid Aa \mid a \\ A \rightarrow BC &\Rightarrow A \rightarrow BC \mid B \mid C \\ B \rightarrow b \mid \lambda &\Rightarrow B \rightarrow b \\ C \rightarrow D \mid \lambda &\Rightarrow C \rightarrow D \\ D \rightarrow d &\Rightarrow D \rightarrow d \end{aligned}$$

۳- حذف تولیدات واحد (حذف زنجیرها)

برای حذف زنجیرها بعد از اینکه زنجیرهای $A \rightarrow A$ مستقیماً حذف شدند، مجموعه‌هایی مستقل از متغیرهای مرتبط به وسیله زنجیرها تشکیل شده و سپس براساس روش زیر قوانین جدید جایگزین می‌شوند.

$$\begin{aligned} A \rightarrow B \\ B \rightarrow a_1 \mid a_2 \mid a_3 \mid \dots \mid a_n \end{aligned} \Rightarrow \begin{cases} A \rightarrow a_1 \mid a_2 \mid \dots \mid a_n \\ B \rightarrow a_1 \mid a_2 \mid \dots \mid a_n \end{cases}$$

مثال: تمامی تولیدات واحد از گرامر زیر را حذف کنید:

$$\begin{aligned} S &\rightarrow Aa \mid B \\ B &\rightarrow A \mid bb \\ A &\rightarrow a \mid bC \mid B \end{aligned}$$

جواب:

I مشخص نمودن تولیدات واحد

$$\begin{cases} S \rightarrow B \\ B \rightarrow A \\ A \rightarrow B \end{cases} \quad \text{مستقیم}$$

$$S \rightarrow A \quad \text{غیر مستقیم}$$

II حذف تولیدات واحد

$$\begin{cases} S \rightarrow Aa \\ B \rightarrow bb \\ A \rightarrow a \mid bC \end{cases}$$

III جایگزینی

$$\begin{cases} S \rightarrow bb \\ B \rightarrow a \mid bC \\ A \rightarrow bb \\ S \rightarrow a \mid bC \end{cases}$$

ادغام موارد II و III:

$$\begin{aligned} S &\rightarrow Aa|bb|a|bC \\ B &\rightarrow bb|a|bC \\ A &\rightarrow a|bC|bb \end{aligned}$$

$$\begin{aligned} S &\rightarrow D \\ S &\rightarrow ABD \\ S &\rightarrow MN \\ B &\rightarrow M|b \\ N &\rightarrow D|nN|n \\ D &\rightarrow dD|d \end{aligned}$$

مثال: تمامی تولیدات واحد را از گرامر زیر حذف کنید.

جواب:

I تولیدات واحد مستقیم

$$\begin{cases} S \rightarrow D \\ B \rightarrow M \\ N \rightarrow D \end{cases}$$

$$\begin{aligned} S &\rightarrow ABD \\ S &\rightarrow MN \\ B &\rightarrow b \\ N &\rightarrow nN|n \\ D &\rightarrow dD|d \end{aligned}$$

II

$$\begin{aligned} S &\rightarrow dD|d \\ B &\rightarrow m \\ N &\rightarrow dD|d \end{aligned}$$

III

: III, II

$$\begin{aligned} M &\rightarrow m \\ S &\rightarrow ABD|dD|d|MN \\ B &\rightarrow m|b \\ N &\rightarrow nN|n|dD|d \\ D &\rightarrow dD|d \\ M &\rightarrow m \end{aligned}$$

۴- حذف متغیرهای غیر مفید

این اصل در دو مرحله صورت می گیرد:

۱. شناسایی و حذف متغیرهایی که قادر به تولید عناصر الفبا نمی باشند. برای این منظور ابتدا باید متغیرهایی که مستقیماً به دنباله ای از عناصر الفبا ختم می شوند را جدا نمود و سپس متغیرهایی که بواسطه این متغیرها دنباله ای از عناصر را تولید می کنند، به آنها اضافه کرد و ادامه داد تا امکان اضافه کردن متغیر دیگری نباشد. مجموعه متغیرهای حاصل شامل متغیرهای مفید است و قوانینی که این متغیرها را در خود ندارند از قوانین گرامر حذف می شوند.

۲. شناسایی و حذف متغیرهای غیر قابل دسترس از S

برای تعیین متغیرهایی که از Start Symbol نمی توان به آنها رسید. می توان از گراف ارتباط (Dependency Graph) استفاده نمود. در گرامرهای مستقل از متن یک گراف ارتباط دارای رئوسی به نام متغیر و یالهای میان رئوس مطابق قوانین تولید می باشد.

$$S \rightarrow aS \mid A \mid C$$

$$A \rightarrow a$$

$$B \rightarrow aa$$

$$C \rightarrow aCb$$

مثال ۱: کلیه تولیدات غیر مفید را از گرامر زیر حذف کنید.

$$S \rightarrow AC \mid BS \mid B$$

$$A \rightarrow aF \mid aA$$

$$B \rightarrow CF \mid b$$

$$C \rightarrow CC \mid D$$

$$D \rightarrow aD \mid BD \mid C$$

$$E \rightarrow aA \mid BSA$$

$$F \rightarrow bB \mid b$$

جواب:

$$S \rightarrow B \mid BS$$

$$B \rightarrow b$$

$$A \rightarrow aF \mid aA$$

$$E \rightarrow aA \mid BSA$$

$$F \rightarrow bB \mid b$$

مثال: تمامی تولیدات غیر مفید را از گرامر زیر حذف کنید.

$$\begin{aligned} S &\rightarrow AB \mid CD \\ A &\rightarrow aA \mid a \\ B &\rightarrow bB \mid b \end{aligned}$$

$$useless_var = \{C, F\}$$

$$C \rightarrow CC$$

$$D \rightarrow a$$

$$F \rightarrow a$$

چون D را حذف کردیم
در نتیجه F هم غیر مفید است

useless production (۱)

تولیدات غیر مفید

احتمالاً حذف قواعدی که در آنها متغیر غیر مفید

وجود دارد useless variable

متغیرهای غیر مفید (۱) در تولیدات غیر مفید (۱) که با آنها

متغیرها (۲) که با آنها

متغیر در ارتباط نیستند

مثال ۲: تولیدات غیر مفید را از گرامر زیر حذف کنید.

$$\begin{aligned} S &\rightarrow aA \mid a \mid BC \\ A &\rightarrow aB \mid \lambda \\ B &\rightarrow Aa \\ C &\rightarrow cCd \\ D &\rightarrow ddd \end{aligned}$$

جواب:

$$\begin{aligned} S &\rightarrow aA \mid a \mid B \\ S &\rightarrow aB \mid \lambda \\ B &\rightarrow Aa \end{aligned}$$

مثال ۳: تمامی تولیدات λ را از گرامر زیر حذف کنید.

$$\begin{aligned} S &\rightarrow AaB \mid aaB \\ A &\rightarrow \lambda \\ B &\rightarrow bbA \mid \lambda \end{aligned}$$

جواب:

{A,B}: مستقیم

$$\begin{aligned} S &\rightarrow AaB \mid Aa \mid aB \mid a \mid aaB \mid aa \\ B &\rightarrow bbA \mid bb \end{aligned}$$

مثال ۴: تمامی تولیدات واحد را از گرامر زیر حذف کنید.

$$\begin{aligned} S &\rightarrow aA \mid a \mid BC \\ A &\rightarrow aB \mid \lambda \\ B &\rightarrow Aa \\ C &\rightarrow cCd \\ D &\rightarrow ddd \end{aligned}$$

جواب:

$$\begin{cases} S \rightarrow B \\ S \rightarrow C \end{cases}$$

$$\begin{cases} S \rightarrow aA \mid a \\ A \rightarrow aB \mid \lambda \\ B \rightarrow Aa \\ C \rightarrow cCD \\ D \rightarrow ddd \end{cases}$$

$$S \rightarrow Aa$$

$$S \rightarrow CCD$$

$$S \rightarrow aA \mid a \mid Aa \mid cCD$$

$$A \rightarrow aB \mid \lambda$$

مثال ۵: تمامی تولیدات غیر مفید را از گرامر زیر حذف کنید.

$$S \rightarrow aS \mid AB$$

$$A \rightarrow bA$$

$$B \rightarrow AA$$

مثال ۶: تمامی تولیدات λ را از گرامر زیر حذف کنید.

$$S \rightarrow abAB$$

$$A \rightarrow bAB \mid \lambda$$

$$B \rightarrow BAa \mid A \mid \lambda$$

جواب:

$$\text{Nullable} = \{A, B\}$$

$$S \rightarrow abAB \mid abB \mid abA \mid ab$$

$$A \rightarrow aAB \mid bA \mid bB \mid b$$

$$B \rightarrow BAa \mid Ba \mid Aa \mid a \mid A$$

مشخص نمودن تولیدات واحد:

$$B \rightarrow A$$

مستقیم

$$\begin{cases} S \rightarrow abAB \\ A \rightarrow bAB \mid \lambda \\ B \rightarrow Baa \mid bAB \mid \lambda \end{cases}$$

حذف تولیدات واحد:

$$\begin{cases} S \rightarrow Aa \mid B \\ B \rightarrow A \mid bb \\ A \rightarrow a \mid bC \mid B \end{cases}$$

$$\begin{cases} S \rightarrow B \\ B \rightarrow A \\ A \rightarrow B \end{cases}$$

مستقیم

$$S \rightarrow A$$

غیر مستقیم

II حذف:

$$\begin{aligned} S &\rightarrow Aa \\ B &\rightarrow bb \\ A &\rightarrow a|bC \end{aligned}$$

III جایگزینی:

$$\begin{cases} S \rightarrow bb \\ B \rightarrow a|bC \\ A \rightarrow bb \\ A \rightarrow a|bC \end{cases}$$

ادغام II و III:

$$\begin{cases} S \rightarrow Aa|bb|a|bC \\ B \rightarrow bb|a|bC \\ A \rightarrow a|bC|bb \end{cases}$$

۵- حذف بازگشتی چپ مستقیم

بعد از شناسایی متغیرهای بازگشتی چپ مستقیم با استفاده از روش زیر قوانین زیر جایگزین می شوند.

$$\begin{aligned} A &\rightarrow Aa|\beta \\ A &\rightarrow B\beta \\ B &\rightarrow a\beta|\lambda \end{aligned}$$

⑤ left recursion
بازگشتی چپ مستقیم

$$A \rightarrow A\alpha|B$$

$$\begin{cases} A \rightarrow A\alpha_1|A\alpha_2|\dots|A\alpha_n \\ A \rightarrow \beta_1|\beta_2|\dots|\beta_m \end{cases}$$

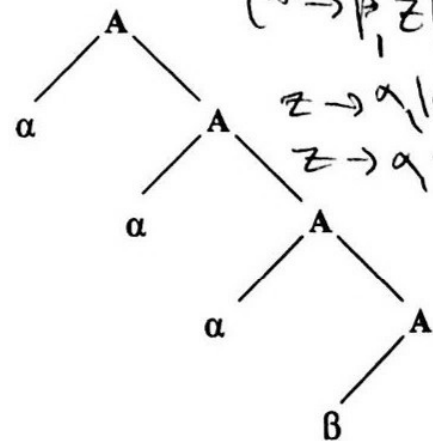
$$A \rightarrow A\alpha_1|A\alpha_2|\dots|A\alpha_n$$

$$A \rightarrow B_1|B_2|\dots|B_m$$

$$\begin{cases} A \rightarrow B_1|B_2|\dots|B_m \\ A \rightarrow B_1Z|B_2Z|\dots|B_mZ \\ Z \rightarrow \alpha_1|\alpha_2|\dots|\alpha_n \\ Z \rightarrow \alpha_1Z|\alpha_2Z|\dots|\alpha_nZ \end{cases}$$

اهتزاز متغیر (Z)

$$\begin{cases} A \rightarrow \beta_1|\beta_2|\dots|\beta_m \\ A \rightarrow \beta_1Z|\beta_2Z|\dots|\beta_mZ \\ Z \rightarrow \alpha_1|\alpha_2|\dots|\alpha_n \\ Z \rightarrow \alpha_1Z|\alpha_2Z|\dots|\alpha_nZ \end{cases}$$



!!? بازگشتی چپ را از این فرمت کنید

$$\begin{cases} A \rightarrow A\alpha_1|A\alpha_2|\dots|A\alpha_n \\ A \rightarrow B_1|B_2|\dots|B_m \end{cases}$$

$$\begin{cases} A \rightarrow bbA|a|bbAZ|aZ \\ Z \rightarrow bb|bA|bbZ|bAZ \end{cases}$$

↑

مثال: حذف بازگشتی چپ مستقیم از گرامر روبرو به صورت زیر می باشد.

$$\begin{aligned} A \rightarrow Abb \mid AbA \\ A \rightarrow aAA \mid a \end{aligned} \rightarrow \begin{aligned} A \rightarrow aAA \mid a \mid aAAZ \mid aZ \\ Z \rightarrow bb \mid BA \mid bbZ \mid bAZ \end{aligned}$$

نکته: دلایل انجام ۵ مرحله تبدیل فوق به صورت زیر می باشد:

۱. حذف نماد آغازگر بازگشتی: عامل حلقه نامتناهی در پارسرهای پائین به بالا.
۲. حذف قوانین λ : عامل حلقه نامتناهی و افزایش مراحل اشتقاق و ابهام در گرامر.
۳. حذف زنجیرهای تولیدات واحد: عامل افزایش مراحل اشتقاق.
۴. حذف متغیرهای غیر مفید: عامل افزایش قوانین گرامر.
۵. حذف بازگشت چپ: عامل حلقه نامتناهی در پارسرهای بالا به پائین.

تبدیل به فرم نرمال گریباخ

در فرم نرمال گریباخ محدودیت روی طول رشته سمت راست نداریم، ولی محل قرار گیری تولیدات و پایانه ها مورد نظر است.

تعریف: یک گرامر مستقل از متن در فرم نرمال گریباخ است، اگر تمامی تولیدات آن به فرم نرمال زیر باشد:

$$\begin{aligned} A \rightarrow ax \\ A \in V \\ a \in \Sigma \\ x \in V^* \end{aligned}$$

مثال: گرامر زیر را به فرم نرمال گریباخ تبدیل کنید.

$$\begin{aligned} S \rightarrow AB &\rightarrow S \rightarrow aAB \mid bBB \mid bB \\ A \rightarrow aA \mid bB \mid b &\rightarrow A \rightarrow aA \mid bB \mid b \\ B \rightarrow b &\rightarrow B \rightarrow b \end{aligned}$$

مثال: گرامر زیر را به فرم نرمال گریباخ تبدیل کنید.

$$S \rightarrow abSb \mid aa \quad \text{فرم نرمال گریباخ}$$

$$\begin{aligned} S \rightarrow aA_b S A_b \mid aA_a \\ A_b \rightarrow b \\ A_a \rightarrow a \end{aligned} \rightarrow \text{یک نماد است، یعنی } A \text{ که جایگزین } b \text{ شده است}$$

مثال: گرامر زیر را به فرم نرمال گریباخ تبدیل کنید.

$$\begin{aligned} S &\rightarrow ABb|a \\ A &\rightarrow aaA|B \\ B &\rightarrow bAb \end{aligned}$$

$$\begin{aligned} \textcircled{3} \quad A &\rightarrow a \cup B \cup b \cup D \\ &\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\ \left\{ \begin{array}{l} A \rightarrow B_a B_b B_D \\ B_a \rightarrow a \\ B_b \rightarrow b \end{array} \right. &\rightarrow \left\{ \begin{array}{l} A \rightarrow B_a T_1 \\ T_1 \rightarrow B_b T_2 \\ T_2 \rightarrow B_D \\ B_a \rightarrow a \\ B_b \rightarrow b \end{array} \right. \end{aligned}$$

گرامر نرمال چامسکی

گرامری مستقل از متن است که قوانین آن به صورت زیر می باشد:

$$\begin{aligned} A &\rightarrow a \\ A &\rightarrow BC \\ S &\rightarrow \lambda \\ a &\in \Sigma \\ A, B, C &\in V \\ \lambda &\in L(G) \quad \text{اگر و فقط اگر} \end{aligned}$$

$$\left\{ \begin{array}{l} A \rightarrow BC \\ A \rightarrow a \\ ABC \in V_N \\ a \in V_T \end{array} \right\} \textcircled{1} \rightarrow \begin{array}{l} a \cup B \\ \downarrow \\ A \rightarrow B_a B_b \\ B_a \rightarrow a \end{array}$$

$$\textcircled{2} \quad \begin{array}{l} A \rightarrow a \cup B \cup b \\ \downarrow \quad \downarrow \quad \downarrow \\ A \rightarrow B_a B_b \\ B_a \rightarrow a \\ B_b \rightarrow b \end{array} \Rightarrow \left\{ \begin{array}{l} A \rightarrow B_a T_1 \\ T_1 \rightarrow B_b B_D \\ B_a \rightarrow a \end{array} \right.$$

نکته: در گرامر چامسکی S بازگشتی نمی باشد. قوانین λ موجود نیستند (به جز $S \rightarrow \lambda$). زنجیرها و تولیدات واحد حذف شده اند و متغیرهای غیر مفید وجود ندارند. تولید گرامر نرمال چامسکی طی همان ۵ مرحله ای که قبلا توضیح داده شد، انجام می شود.

تبدیل نهایی به فرم نرمال چامسکی

تبدیل نهایی گرامر به فرم نرمال چامسکی طی مراحل زیر صورت می گیرد:

- به ازای هر قانون بصورت $A \rightarrow a\alpha$ که در آن $\alpha \in (\Sigma \cup V)^+$ و $length(\alpha) \geq 2$ دو متغیر تعریف شده و قوانین زیر جایگزین می شوند:

$$\begin{aligned} A &\rightarrow T_1 T_2 \\ T_1 &\rightarrow a \\ T_2 &\rightarrow \alpha \end{aligned}$$

- به ازای هر قانون به شکل $A \rightarrow A'\alpha$ که در آن از متغیر جدید T استفاده شده و قوانین زیر جایگزین می شود:

- به ازای هر قانون به شکل $A \rightarrow a_1 a_2$ که در آن $a_1, a_2 \in \Sigma$ با استفاده از متغیرهای جدید T_1, T_2 قوانین زیر جایگزین می شود:

$$\begin{aligned} S' &\rightarrow \delta a B | \delta a | a B | a \\ \left\{ \begin{array}{l} S' \rightarrow \delta T_1 | \delta B_a | B_a B | a \\ T_1 \rightarrow \delta a \\ B_a \rightarrow a \end{array} \right. &\rightarrow \left\{ \begin{array}{l} Z \rightarrow B_a \delta | a | \delta_a T_2 | B_a Z \\ B \rightarrow B_b B | b \\ B_b \rightarrow b \end{array} \right. \end{aligned}$$

$$A \rightarrow T_1 T_2$$

$$T_1 \rightarrow a_1$$

$$T_2 \rightarrow a_2$$

۴. به ازای هر قانون به صورت $A \rightarrow a_1 A'$ با متغیر جدید T ، دو قانون زیر تولید شده و جایگزین می گردد:

$$A \rightarrow T A'$$

$$T \rightarrow a_1$$

۵. به ازای قوانین به صورت $A \rightarrow A' a$ قوانین زیر با استفاده از متغیر جدید T جایگزین می گردد:

$$A \rightarrow A T$$

$$T \rightarrow a$$

پنج مرحله فوق بر گرامر اعمال می شود تا گرامر به فرم نرمال چامسکی تبدیل شود.

مثال: گرامر زیر را به فرم نرمال چامسکی تبدیل کنید.

$$S \rightarrow AaBb$$

$$S \rightarrow ab$$

$$A \rightarrow AaAB$$

$$A \rightarrow aa$$

$$B \rightarrow bB$$

$$B \rightarrow b$$

جواب:

$$S \rightarrow AaBb \Rightarrow \begin{cases} S \rightarrow AT_1 \\ T_1 \rightarrow aBb \end{cases} \Rightarrow \begin{cases} T_1 \rightarrow T_2 T_3 \\ T_2 \rightarrow a \\ T_3 \rightarrow Bb \end{cases} \Rightarrow \begin{cases} T_3 \rightarrow BT_4 \\ T_4 \rightarrow b \end{cases}$$

$$S \rightarrow ab \Rightarrow S \rightarrow T_2 T_4 \quad \text{با استفاده از قوانین موجود:}$$

$$A \rightarrow AaAB \Rightarrow \begin{cases} A \rightarrow AT_5 \\ T_5 \rightarrow aAB \end{cases} \Rightarrow \begin{cases} T_5 \rightarrow T_2 T_6 \\ T_6 \rightarrow AB \end{cases}$$

$$A \rightarrow aa \Rightarrow A \rightarrow T_2 T_2$$

$$B \rightarrow bB \Rightarrow B \rightarrow T_4 B$$

در نتیجه گرامر نهایی به شکل روبرو بدست می آید:

$$S \rightarrow AT_1$$

$$T_1 \rightarrow T_2T_3$$

$$T_2 \rightarrow a$$

$$T_3 \rightarrow BT_4$$

$$T_4 \rightarrow b$$

$$S \rightarrow T_2T_4$$

$$A \rightarrow AT_5$$

$$T_5 \rightarrow T_2T_6$$

$$T_6 \rightarrow AB$$

$$A \rightarrow T_2T_2$$

$$B \rightarrow T_4B$$

$$B \rightarrow b$$

@EngineersRepository

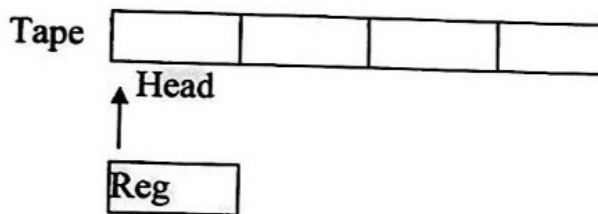
ماشین های متناهی (Finite State Automate)

ماشینها مدلهایی هستند که قادر به پردازش زبان باشند. ماشینهای متناهی (FA) ماشین هایی هستند که زبانهای با قاعده را می پذیرند. ماشین های متناهی به سه دسته تقسیم می شوند:

- | | |
|---------------------------------------|-----------------------|
| DFA: Deterministic Finite Automate | ۱. قطعی |
| N-DFA: Non Deterministic Finite State | ۲. غیر قطعی |
| λ -NFA | ۳. غیر قطعی λ |

ماشین های متناهی قطعی (DFA)

ماشین های متناهی قطعی دارای ساختار نمایش زیر هستند:



خصوصیت ماشین های DFA

این خصوصیات عبارتند از:

- دارای نوار می باشد که به مکانهایی تقسیم شده که هر مکان می تواند یک عنصر الفبا را در خود ذخیره نماید. جمله ورودی ماشین روی نوار قرار می گیرد، بطوریکه سمت چپ ترین عنصر جمله در سمت چپ ترین مکان نوار قرار داده می شود. این نوار از چپ متناهی و از راست نامتناهی است.
- این ماشین ها برای مرور ورودی دارای نوار خوانی هستند که در مقابل یک مکان از نوار قرار دارد و با هر دستور به اندازه یک مکان به راست حرکت می کند.
- این ماشین ها دارای یک ثبات حالت هستند که محتوای آن حالت ماشین در هر لحظه را مشخص می نماید و این محتوا با اجرای دستورهای ماشین تغییر می کند.
- هر ماشین به صورت $M = (q_0, \Sigma, Q, \delta, F)$ تعریف می شود که در آن:

q_0 وضعیت ابتدای ماشین ($q_0 \in Q$)

Σ الفبای زبان

Q مجموعه متناهی حالات ماشین

δ مجموعه دستورالعمل ها

F زیر مجموعه ای از Q که حالت نهایی یا پذیرش است. (Accept شده)

- دستورالعمل های ماشین متناهی قطعی به صورت $\delta: Q \times \Sigma \rightarrow Q$ تعریف می شوند.

$$\delta(q_i, a) = q_j$$

که بدین معناست که اگر a در مقابل نوار خوان است و q_i در ثبات حالت قرار دارد، نوار خوان یک مکان به سمت راست می رود و q_j در ثبات حالت ذخیره می شود.

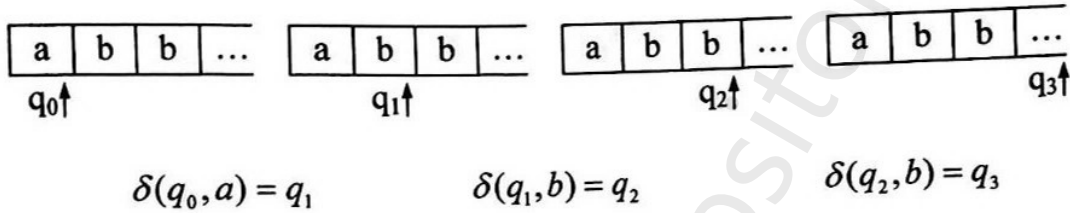
- در ماشین های قطعی متناهی (DFA) به ازای

$$\begin{cases} \forall q_i \in Q \\ \forall a \in \Sigma \end{cases}$$

فقط و فقط باید یک دستور به صورت $(q_i, a) = q_j$ موجود باشد.

نکته ۱: اگر ماشین در یکی از حالات متعلق به F متوقف شود ورودی پذیرفته شده است.
نکته ۲: ماشین قطعی در صورتی متوقف می شود که ورودی تمام شده باشد.

مثال: وضعیت ماشین در مراحل مختلف پذیرش ورودی abb به صورت زیر می باشد:



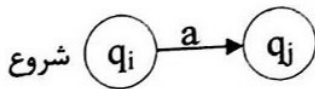
نکته: وضعیت ماشین در انتقال حالت را می توان به صورت زیر نشان داد:

$$\{\text{باقیمانده جمله ورودی}, q_i\} \rightarrow \{\text{باقیمانده جمله ورودی}, q_j\}$$

در توقف ماشین اگر حالت نهایی باشد جمله ورودی معتبر است و اگر حالت غیرنهایی باشد، جمله ورودی معتبر نیست.

دیاگرام حالت ماشین DFA

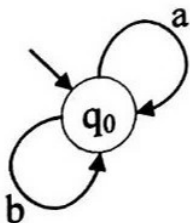
دیاگرام حالت ماشین DFA گرافی جهت دار است که گره های آن حالات ماشین و لبه های آن مبین عناصر ورودی می باشد. به گونه ای که برای دستورالعمل گراف زیر متصور است:



حالت نهایی هم به صورت $\textcircled{q_f}$ نشان داده می شود.

مثال: دیاگرام حالت برای ماشین پذیرنده زبان $L(G)$ زیر به صورت زیر می باشد:

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



$$q_q = \{q_0\}$$

$$Q = \{q_0\}$$

$$\Sigma = \{a, b\}$$

$$F = \{q_0\}$$

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

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

نکته: دیاگرام حالت برای حالات a^+ و a^* به صورت زیر می باشد:



زبان ماشین متناهی قطعی

زبان حاصل از یک ماشین متناهی قطعی M به صورت زیر تعریف می شود:

$$L_{(M)} = \{\omega \mid \omega \in \Sigma^*, [q_0, \omega] \xrightarrow{*} [q_f, -], q_f \in Q\}$$

یک کلمه متشکل از همه حروف الفبا که طی صفر یا یک یا چند مرحله از q_0 تولید شده به حالت **accept** شده q_f برسیم.

ماشین های متناهی غیر قطعی (NFA)

یک ماشین متناهی غیر قطعی به صورت $M = (q_0, \Sigma, Q, \delta, F)$ تعریف می شود که در آن:

Σ مجموعه الفبای ماشین

Q مجموعه متناهی حالات ممکن ماشین

q_0 حالت ابتدایی ماشین ($q_0 \in Q$)

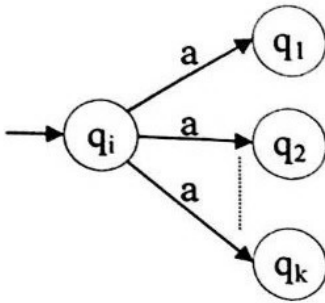
δ مجموعه دستورالعمل ها که به صورت زیر تعریف می شود:

$$\delta = Q \times \Sigma \rightarrow 2^Q$$

F زیر مجموعه ای از حالات ماشین که حالت نهایی یا پذیرش نامیده می شود

نکته: ماشین های متناهی غیر قطعی مانند ماشین های متناهی قطعی دارای دیاگرام حالت با همان تعریف می باشند.

نکته: در ماشین های متناهی غیر قطعی حالت با یک ورودی می تواند به چند حالت منتقل شود.



تفاوت دو نوع ماشین:

فقط و فقط در DFA در ماشین های متناهی قطعی

$$(q_i, a) = q_j$$

در NFA در ماشین های متناهی غیر قطعی

$$(q_i, a) = \{q_1, q_2, \dots, q_k\}$$

نکته: شرط پذیرش یک جمله توسط ماشین NFA توقف در یکی از حالات نهایی و خاتمه ورودی است.

نکته: در ماشین قطعی فقط یک محاسبه داریم که یا موفق است و یا ناموفق است ولی در ماشین غیر قطعی ممکن است چند محاسبه داشته باشیم که موفق یا ناموفق باشند.

نکته: در ماشین متناهی غیر قطعی یک جمله ورودی در صورتی پذیرفته می شود که حداقل یک محاسبه موفق برایش موجود باشد.

به عنوان مثال در ماشین های قطعی و غیر قطعی ارائه شده برای زبان $\Sigma = \{a, b\}, L = a^+b$ محاسبات موفق و ناموفق زیر برای تشخیص تعلق جمله ورودی aab به زبان وجود دارد.

ماشین قطعی موفق

$$[q_0 = aab] \rightarrow [q_1, ab] \rightarrow [q_2, b] \rightarrow [q_3, -]$$

ماشین غیر قطعی ناموفق

$$[q_0, aab] \rightarrow [q_1, ab]$$

$$[q_0, aab] \rightarrow [q_1, ab] \rightarrow [q_2, b]$$

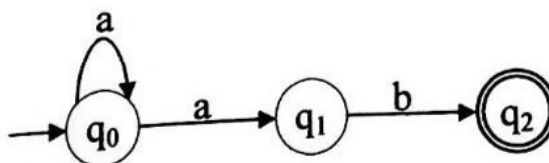
زبان ماشین متناهی غیر قطعی

زبان حاصل از ماشین متناهی غیر قطعی به صورت زیر تعریف می شود:

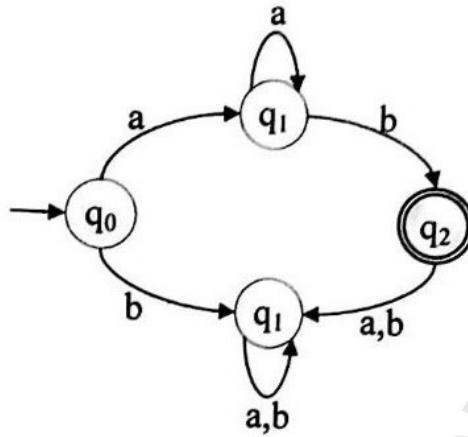
$$L_{(M)} = \{\omega \mid \omega \in \Sigma^*, \exists [q_0, \omega] \xrightarrow{*} [q_f, -], q_f \in f\}$$

مثال: ماشین پذیرنده NFA و DFA برای a^+b به صورت زیر می باشد.

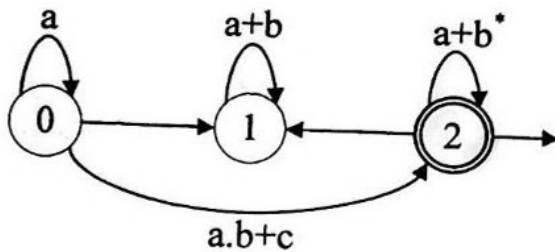
DFA:



NFA:



مثال: زبان پذیرفته شده توسط گراف انتقال زیر کدام است ؟



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

جواب:

ماشین های متناهی غیر قطعی λ (NFA- λ)

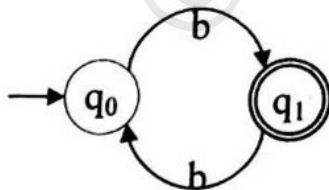
این نوع ماشین ها انتزاعی ترین نوع ماشین های متناهی است که بدون خواندن ورودی می تواند تغییر حالت دهد. هدف از این ماشین انجام طراحی ها به سادگی توسط ماشین است. یک ماشین NFA- λ به صورت $M : (q, \Sigma, Q, \delta, F)$ تعریف می شود که همه موارد همان قبلی است و فقط داریم:

$$\delta : q \times \Sigma \cup \{\lambda\} \rightarrow 2^q$$

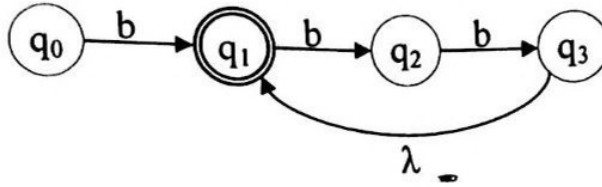
نکته: شرط پذیرش در این ماشین توقف در یکی از حالات نهایی و خاتمه ورودی است.

مثال: در ماشین NFA- λ و NFA زیر را برای پذیرش جملات زبان $L = (bb)^*b$ می توان ارائه نمود.

NFA:



λ -NFA:



تبدیل ماشین متناهی به نوع ماشین غیر قطعی λ

هر ماشین متناهی M را می توان به ماشین متناهی غیر قطعی λ تبدیل کرد بطوریکه زبان هر دو ماشین یکسان باشند. این تبدیل طی دو مرحله انجام می شود:

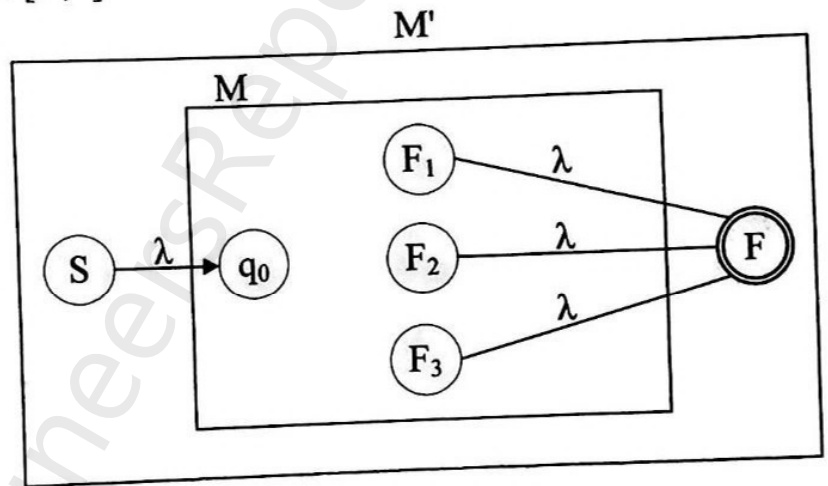
۱. اضافه نمودن حالت ابتدایی S و حالت نهایی F به ماشین.

۲. تعریف دستورالعمل های اضافی زیر:

$$\delta(s, \lambda) = \{q_0, \forall f_i \in F_M\}$$

$$\delta(f_i, \lambda) = \{F\}$$

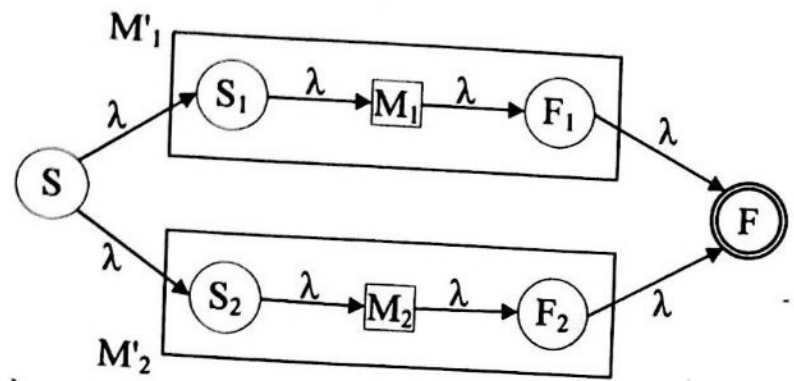
$$[S, \omega] \rightarrow [q_0, \omega] \xrightarrow{\cdot} [F_i, -] \rightarrow [F, -]$$



اتحاد و اتصال ماشین های متناهی

اتحاد: اگر M_1, M_2 ماشین های متناهی باشند، ماشین متناهی غیر قطعی λ (M') که زبان آن برابر $L(M_1) \cup L(M_2)$ می باشد، به روش زیر تولید می شود:

۱. M_1 به غیر قطعی λ (M'_1) تبدیل می شود. (S_1 وضعیت ابتدایی و F_1 وضعیت نهایی است).
۲. M_2 به غیر قطعی λ (M'_2) تبدیل می شود. (S_2 وضعیت ابتدایی و F_2 وضعیت نهایی است).
۳. دستورالعمل های زیر تعریف می شوند و حالت های F_1, F_2 غیر نهایی می شوند.



$$\delta(S, \lambda) = \{S_1, S_2\}$$

$$\delta(F_1, \lambda) = \{F\}$$

$$\delta(F_2, \lambda) = \{F\}$$

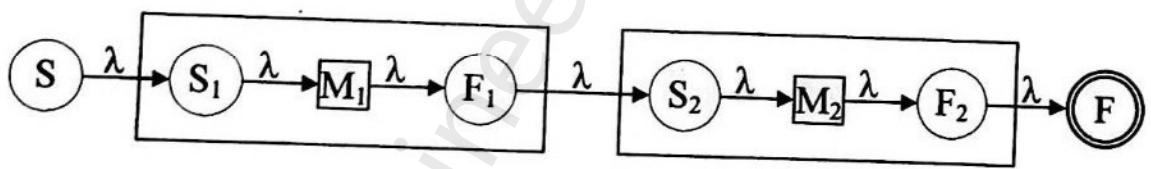
اتصال: اگر M_1, M_2 ماشین های متنهای باشند، ماشین متنهای غیر قطعی $\lambda(M')$ که زبان آن برابر $L(M_1)L(M_2)$ می باشد، به روش زیر تولید می شود:

۱. M_1 به غیر قطعی $\lambda(M'_1)$ تبدیل می شود. (S_1 وضعیت ابتدایی و F_1 وضعیت نهایی است).
۲. M_2 به غیر قطعی $\lambda(M'_2)$ تبدیل می شود. (S_2 وضعیت ابتدایی و F_2 وضعیت نهایی است).
۳. دستورالعمل های زیر تعریف می شوند و حالت های F_1, F_2 غیر نهایی می شوند.

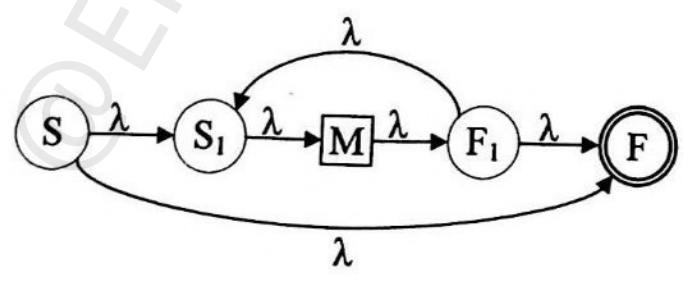
$$\delta(S, \lambda) = \{S_1\}$$

$$\delta(F_1, \lambda) = \{S_2\}$$

$$\delta(F_2, \lambda) = \{F\}$$



در حالت خاص می توان دیاگرام حالت ماشین M' که زبان ماشین آن $L(M') = [L(M)]^*$ می باشد را کلاً به صورت زیر تولید کنید:



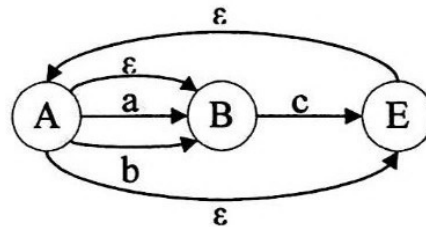
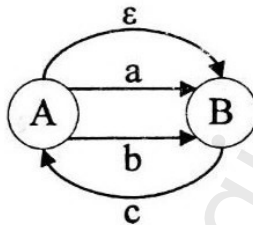
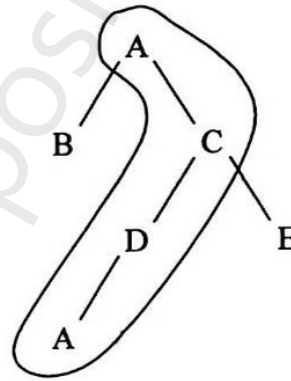
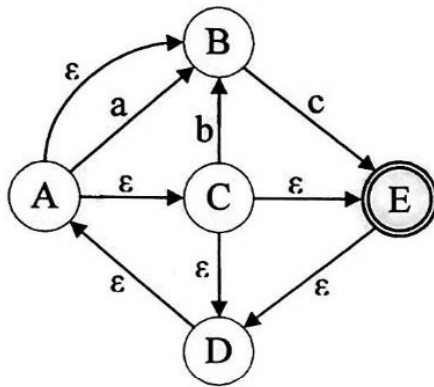
تبدیل DFA به NFA

این تبدیل در چهار مرحله صورت می گیرد:

۱. یافتن و حذف حلقه های متناهی
۲. حذف گذرهای تهی
۳. حذف عدم قطعیت
۴. بهینه سازی

الگوریتم تبدیل NFA به DFA

۱- الگوریتم یافتن و حذف حلقه های تهی:



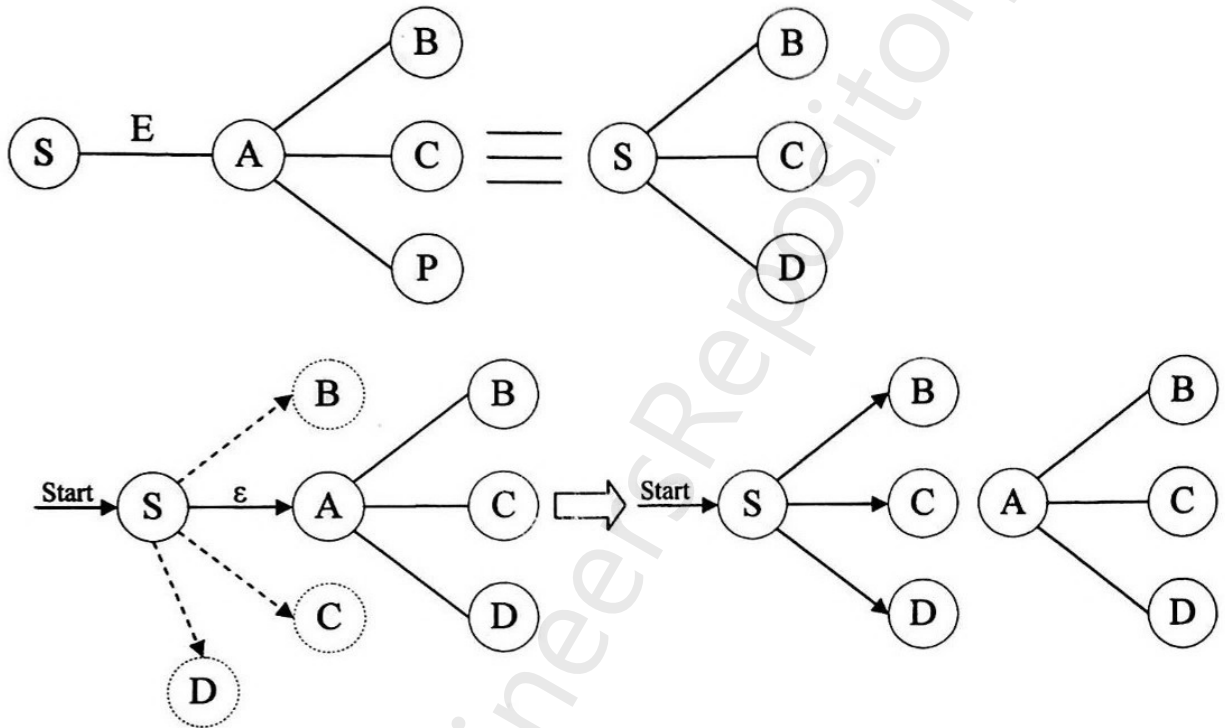
توضیحات: ماشین بالا را در نظر می گیریم ، برای یافتن حلقه های تهی به ترتیب زیر عمل می کنیم:

۱. برای حالت P یک درخت به صورتی که ریشه درخت شامل P باشد تشکیل می دهیم.
۲. کلیه حالاتی که توسط گذر تهی به حالت P متصل است را به عنوان شاخه درخت یا گره های قبل درخت در نظر گرفته و به P متصل می کنیم . همین عمل را برای حالات جدید ادامه می دهیم تا اینکه:
۳. اگر دیگر شاخه جدید نتوان به درخت اضافه کرد و P داخل درخت دوباره تکرار نشود گره P را برچسب زده و می گوئیم هیچ حلقه تهی از P به P وجود ندارد.
۴. در صورتیکه P یکبار به عنوان ریشه و یکبار به عنوان شاخه یا گره میانی درخت تکرار شود مشخص است که حلقه ای از P به خود P در درون گراف موجود است.
۵. کلیه گره های درون مسیر دو تکرار متوالی گره ریشه را مشخص می کنیم. به عنوان نمونه در درخت فوق این مجموعه شامل ACD بود این سه گره تشکیل یک حلقه تهی را می دهد. پس می توان D,C را با A

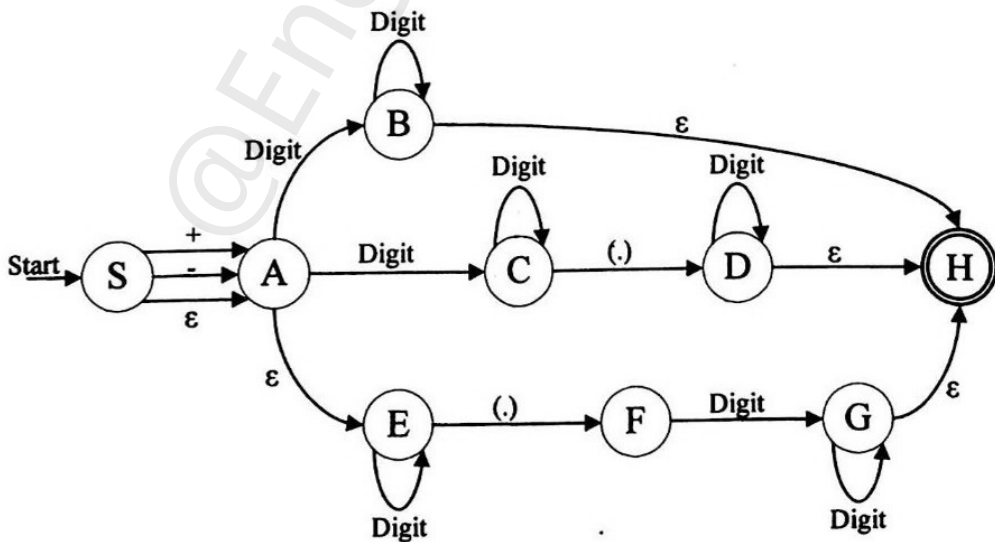
جایگزین نمود. اما با حذف D,C می بایست کلیه گره های متصله به گره جایگزین کننده D,C را در گراف به عهده دارد.

۲- الگوریتم یافتن و حذف گذرهای تهی

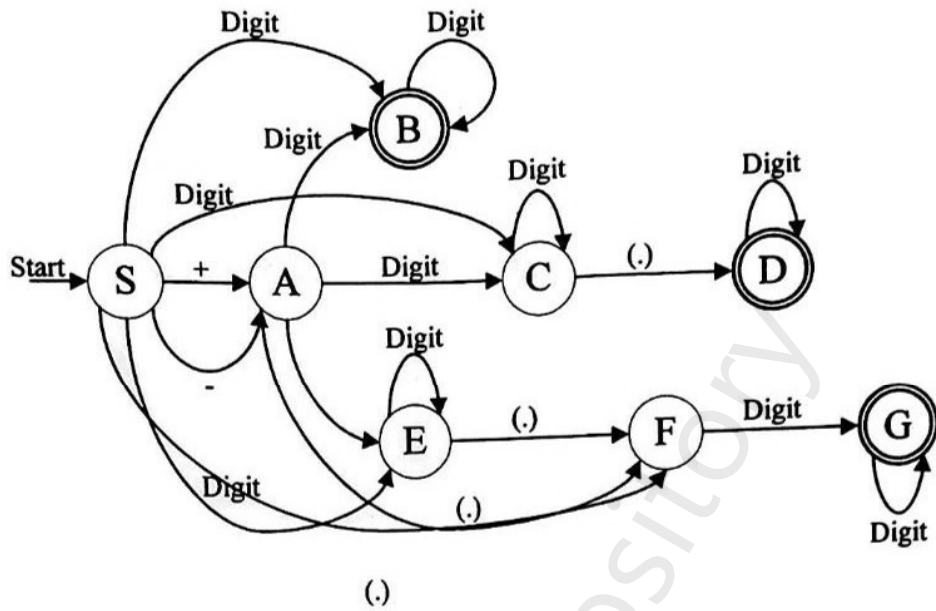
اگر از حالت S گذر تهی به حالت A وجود داشته باشد، این امر نمایانگر یک تساوی یک طرفه است. در این صورت S با A هیچ تفاوتی ندارد. اما A متمایز از S است. می توان هر واکنشی که در A وجود دارد را برای S نیز در نظر گرفت.



۳- حذف عدم قطعیت



بعد از حذف حلقه های تهی و گذرهای تهی



مثال: می خواهیم فرم کلی اعداد صحیح و اعشاری را با ماشین خودکار متناهی تولید نماییم. به دو پیش فرض زیر توجه می کنیم:

۱. عدد می تواند با علامت یا بدون علامت باشد.
۲. عدد می تواند صحیح یا اعشاری با حداقل یک رقم پس از نقطه اعشار و یا اعشاری بدون رقم پیش از نقطه اعشار باشد.

توضیح الگوریتم حذف عدم قطعیت:

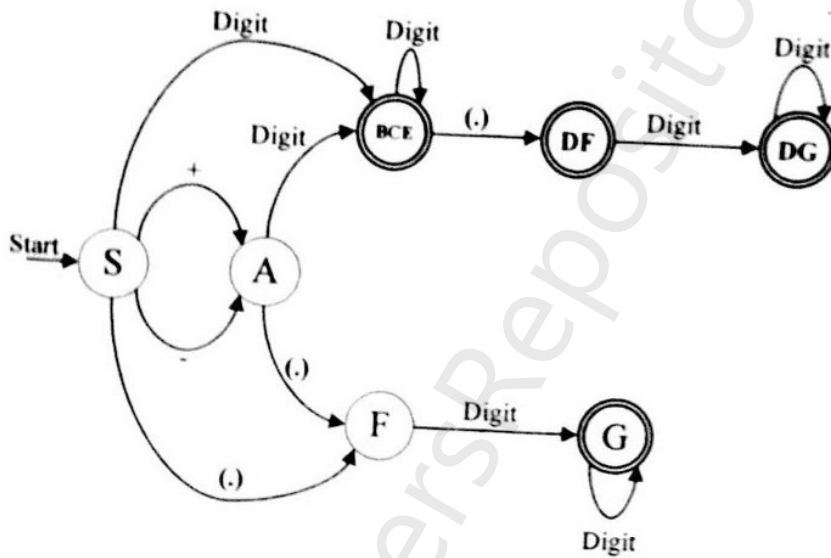
همانگونه که در شکل مشاهده می شود ماشین خودکار غیر قطعی است. به عنوان مثال با دیدن digit نمی توان تشخیص داد که آیا حالت بعدی B یا C یا E می باشد. (از حالت A) اما نکته قابل توجه این است که با دیدن digit حتما به یکی از سه حالت B یا C یا E گذر می شود. جدول گذر حالت فوق به صورت زیر می باشد:

	Digit	(.)	+	-
S	B,C,E	F	A	
A	B,C,E	F	-	
(B)	B	-	-	
C	C	D	-	
(D)	D	-	-	
E	E	F	-	
F	G	-	-	
(G)	G	-	-	

(BCE)	BCE	DF	-	
(DF)	DG			
(DF)	DG			

همانگونه که مشاهده می شود، در حالت A به یکی از سه حالت B یا C یا E به ازای دیدن Digit گذری وجود دارد. لذا می توان گفت که گذری به حالت BCE وجود دارد. در این حالت می توان واکنش های هر سه حالت B یا C یا E را به صورت توأم داشت.

ردیف مربوط به حالت ترکیبی از واکنشهای هر سه حالت تشکیل دهنده آن است. به عبارت دیگر از ترکیب سطرهای B و C و E سطر جدید BCE به جدول اضافه می شود. BCE یک حالت پذیرشی است. زیرا یکی از حالات تشکیل دهنده آن یعنی B پذیرشی است. همانگونه که مشاهده می شود در حالت جدید BCE به ازای ورودی (.) عدم قطعیت وجود دارد. از حالت BCE می توان به هر دو حالت D یا F گذر نمود. لذا برای رفع عدم قطعیت حالت جدید دیگری به نام DF را به جدول حالات باید افزود و برای رفع عدم قطعیت در حالت DF حالت DG ایجاد می شود.



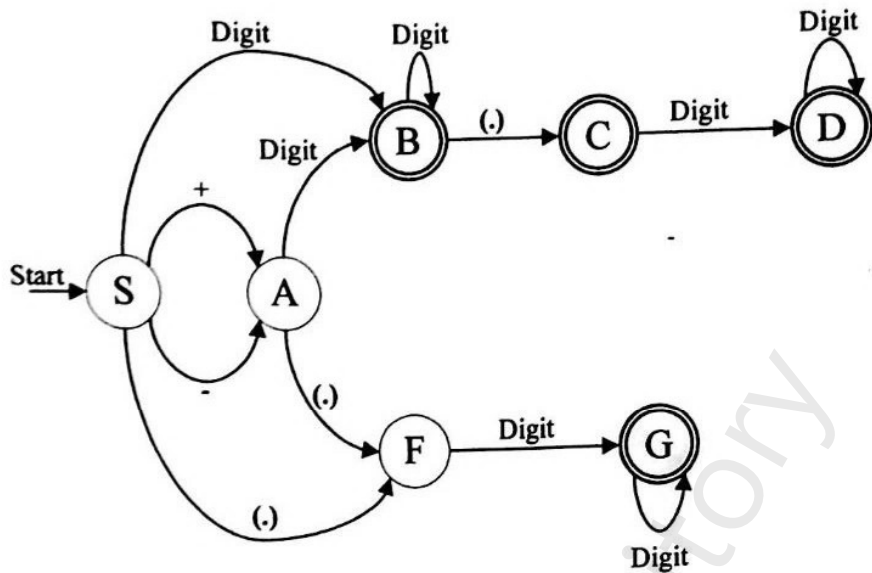
۴- بهینه سازی ماشین خودکار

هدف از بهینه سازی تقلیل تعداد حالات در ماشین های خودکار است. برای این منظور باید حالات معادل در ماشین را مشخص نمود.

تعریف در حالت معادل:

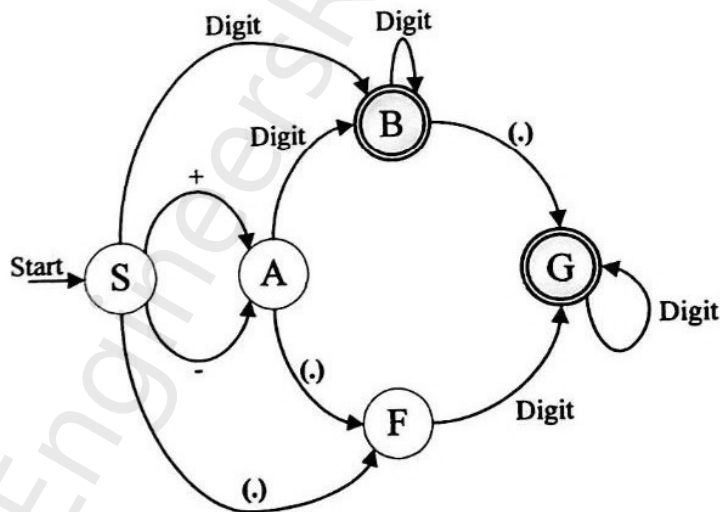
دو حالت را معادل گویند اگر و فقط اگر به ازای هر رشته ای از ورودیها متوالی که موجب رسیدن از آن حالت به یک حالت پذیرفته شده است بتوان از دیگری نیز به یک حالت پذیرشی رسید. حالات معادل را می توان به ترتیب زیر مشخص نمود:

1. ابتدا حالت پذیرش را از حالت غیر پذیرش مجزا می کنیم. بدین ترتیب حالات به دو دسته تقسیم می شوند.
2. در هر دسته باید مشخص نمود که آیا ورودی وجود دارد که به ازای آن حالات متعلق به یک دسته واکنش های جدا از سایرین داشته باشد واکنشها براساس گذر یک حالت به حالت دیگر سنجیده نمی شوند. بلکه اگر از یک حالت به حالت دیگر گذری وجود دارد دسته مربوط به آن حالت را مشخص می کند.
3. مرحله ۲ را تا زمانی که دیگر گذری متفاوت بین دسته ها وجود نداشته باشد و دسته جدید تولید نشود تکرار می کنیم.



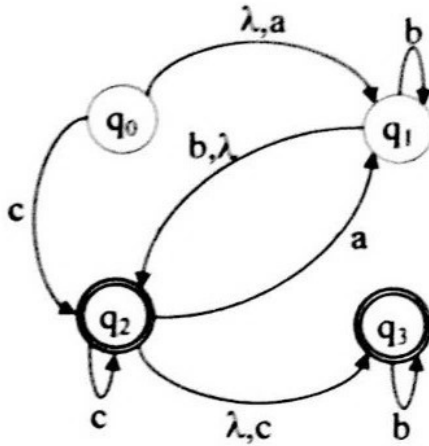
- I) {S,A,F} غیر پذیرشی {B,C,E,G}
- {B} پذیرش {A}
- {C,E,G} {F}
- {S}

شکل بهینه شده:



تابع t

مجموعه حالت هایی که از یک حالت خاص با دریافت یک عنصر حتی به صورت غیر مستقیم (خواندن λ) در دسترس هستند مجموعه t یا تابع t را تشکیل می دهند. به عبارت دیگر مجموعه $T(q_i, a)$ مجموعه کلیه حالت هایی است که از q_i با خواندن a حتی به صورت غیر مستقیم قابل دسترسی هستند. به عنوان نمونه برای ماشین زیر می توان نوشت:



- $T(q_0, b) = \{q_1, q_2, q_3\}$
- $T(q_0, a) = \{q_1\}$
- $T(q_0, c) = \{q_2, q_3\}$
- $T(q_3, b) = \{q_3\}$
- $T(q_3, a) = \{\}$

تولید گرامر با قاعده از ماشین متناهی

زبانهای با قاعده و ماشین های متناهی متناظرند یعنی برای هر زبان با قاعده یک ماشین متناهی وجود دارد و زبان هر دو ماشین متناهی با قاعده است. یک محاسبه در ماشین متناهی یک مرحله اشتقاق در گرامر هاست. در ماشین عناصر یک به یک خوانده می شوند و در اشتقاق عناصر یک به یک تولید می شوند. برای نوشتن گرامر از روی ماشین متناهی مطابق شکل زیر به ازای هر خواندن ورودی دو حالت یک دستور گرامر نوشته می شود:



تولید گرامر از ماشین متناهی

از نگاه دیگر می توان گفت که گرامر $G = \{\Sigma, V, R, S\}$ از ماشین متناهی $M = \{\Sigma, q_0, Q, \delta, F\}$ به صورت زیر حاصل می شود:

$$\begin{aligned} \Sigma_G &= \Sigma_M & \forall \delta(q_0, a) = \{q'_1, q'_2, \dots, q'_k\} \Rightarrow \\ S &= q_0 & \forall q'_j \in \{q'_1, q'_2, \dots, q'_k\} \\ V &= Q & \{q_i \rightarrow aq'_j \text{ if } q'_j \in F \text{ then } q_i \rightarrow q\} \end{aligned}$$

بنابراین ذکر شده برای ماشین λ -NFA داریم: $a \in \Sigma_M \cup \{\lambda\}$
در صورتی که $a = \lambda$ باشد آنگاه:

$$q_i \rightarrow aq'_j \equiv q_i \rightarrow q'_j$$

$$q_i \rightarrow a \equiv q_i \rightarrow \lambda$$

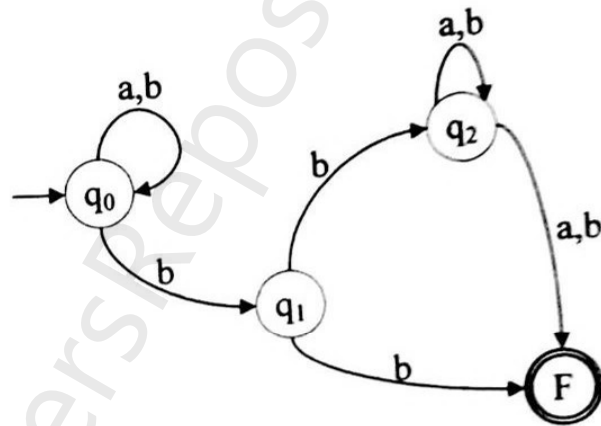
بدست آوردن ماشین متناهی با استفاده از گرامر با قاعده

بدست آوردن ماشین متناهی از گرامر با قاعده عکس حالت تولید گرامر از ماشین است. هر متغیر یک حالت ماشین را می سازد هر دستور گرامر یک انتقال را نشان می دهد و نیز یک حالت اضافی با نام حالت نهایی تعریف می شود و برای هر قانونی که یک متغیر را با یک عنصر الفبا جایگزین می نماید، انتقال به این حالت نهایی پیاده می شود. به عنوان نمونه ماشین حاصل از قوانین گرامر زیر در شکل ذیل آمده است.

$$q_0 \rightarrow bq_0 \mid aq_0 \mid bq_1$$

$$q_1 \rightarrow bq_2 \mid b$$

$$q_2 \rightarrow aq_2 \mid bq_2 \mid a \mid b$$



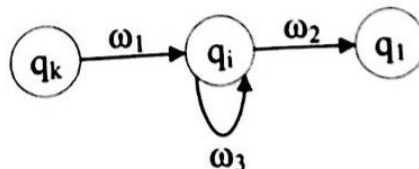
تولید عبارت با قاعده از ماشین متناهی

روش تولید یک عبارت با قاعده از ماشین متناهی بستگی به تعداد حالات نهایی دارد که در زیر بررسی می شود.

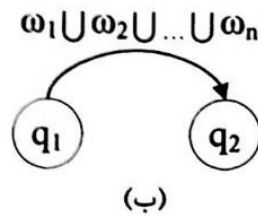
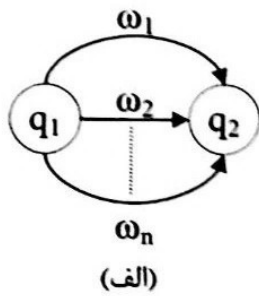
الف: ماشین متناهی با یک حالت نهایی

مراحل زیر برای تولید عبارت با قاعده از این ماشین پیاده می شوند.

۱. به ازای هر q_i به جز حالات ابتدایی و نهایی برای کلیه حالات q_k و q_1 به جز q_i در صورت وجود زیر گراف متقابل لبه ای از q_k به مقصد q_1 با نام $\omega_1 \omega_3 \omega_2$ به دیاگرام اضافه می شود:

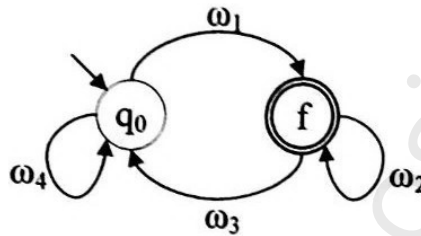


۲. q_i و لبه های مربوطه (دستورالعمل ها) از دیاگرام ماشین حذف می شوند.
۳. در صورت وجود زیر گراف به صورت شکل (الف) با زیر گراف به صورت شکل (ب) جایگزین می شود.

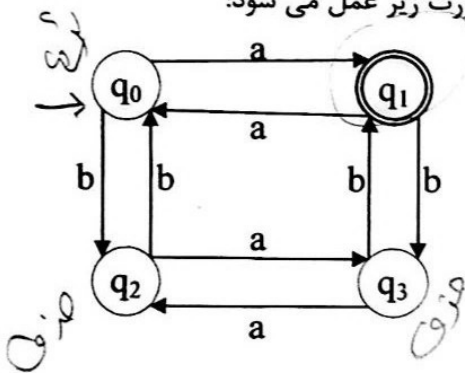


در نهایت با وجود زیر گراف نظیر گراف روبرو عبارت با قاعده:

$$\omega_1^* \omega_2^* (\omega_3 \omega_4^* \omega_1 \omega_2^*)^*$$

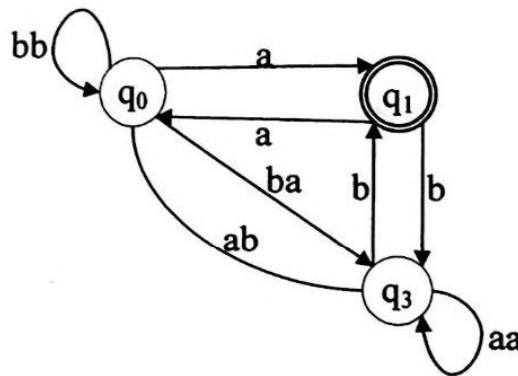


مثال: برای بدست آوردن عبارت با قاعده حاصل از ماشین متناهی روبرو به صورت زیر عمل می شود:

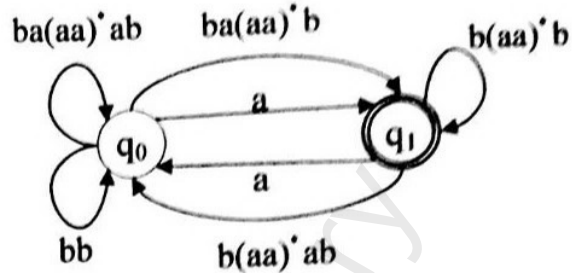
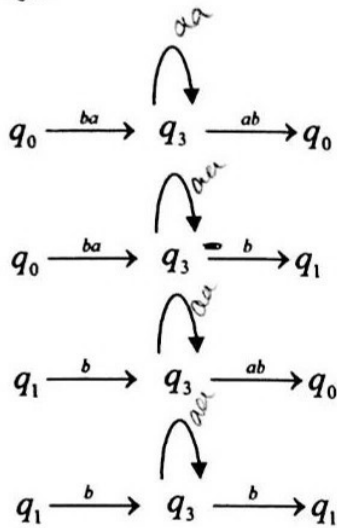


مراحل حذف q_2 :

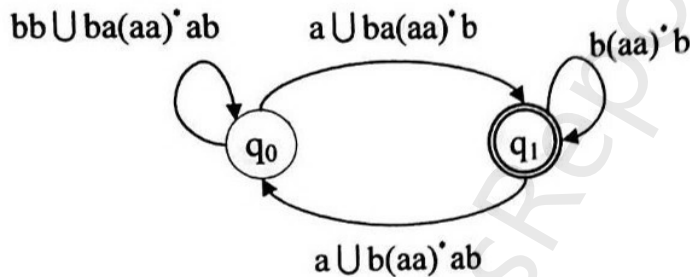
- ✓ $q_0 \xrightarrow{b} q_2 \xrightarrow{b} q_0$
- $q_0 \xrightarrow{b} q_2 \quad q_1$
- ✓ $q_0 \xrightarrow{b} q_2 \xrightarrow{a} q_3$
- $q_1 \quad q_2 \xrightarrow{b} q_0$
- $q_1 \quad q_2 \quad q_1$
- $q_1 \quad q_2 \xrightarrow{a} q_3$
- ✓ $q_3 \xrightarrow{a} q_2 \xrightarrow{b} q_0$
- $q_3 \xrightarrow{a} q_2 \quad q_1$
- ✓ $q_3 \xrightarrow{a} q_2 \xrightarrow{a} q_3$



مراحل حذف q_3 :



و در مرحله آخر نیز از شکل فوق نتیجه می شود که عبارت با قاعده نهایی از روی آن بدست می آید:



عبارت با قاعده نهایی:

$$[((ba(aa)^*ab) \cup bb)^* ((ba(aa)^*b) \cup a)(b(aa)^*b)^*]$$

$$((b(aa)^*ab) \cup a)(ba(aa)^*ab \cup bb)(ba(aa)^*b) \cup a(b(aa)^*b)^*$$

ب: ماشین متناهی با تعدادی حالت نهایی

در این حالت از دیاگرام ماشین به تعداد حالات نهایی کپی برداشته می شود و در هر کپی حالت نهایی یکی از حالات نهایی دیاگرام اصلی است. هر دیاگرام عبارتی را بدست می دهد که در نهایت عبارت با قاعده اصلی اتحاد عبارات با قاعده می باشد.

نکته: اتحاد و اتصال دو زبان با قاعده ، با قاعده است.

نکته: اشتراک دو زبان با قاعده ، مستقل از متن است.

ماشین های پشته ای : (P.D.A) Push Down Automata

ماشینی است متناهی که توسط سازه ای به نام پشته تقویت شده است. ماشین های پشته ای زبان های مستقل از متن را می پذیرد. بدین معنا که برای هر زبان مستقل از متن یک ماشین پشته ای وجود دارد و زبان یک ماشین پشته ای مستقل از متن است. به عبارت دیگر ماشین پشته ای برای مکانیزه نمودن زبان مستقل از متن است.

الفبای پشته

مجموعه عناصری که می تواند در یک پشته قرار گیرد الفبای پشته نامیده می شود.

خصوصیات ماشین های پشته ای

۱. هر ماشین پشته ای علاوه بر امکانات ماشین متناهی نظیر نوار خوان و ثبات حالت دارای یک پشته نیز می باشد.

۲. هر ماشین پشته ای بصورت $M = \{q_0, \Sigma, Q, \delta, F, \Gamma\}$ تعریف می شود که در آن Σ عناصر مجموعه الفبا، q_0 حالت ابتدایی، Q مجموعه حالت ها، δ مجموعه دستور العمل ها، F مجموعه حالت نهایی و Γ الفبای پشته می باشد که مجموعه دستور العمل ها در ماشین پشته ای از تعاریف زیر پیروی می کند:

$$PDA: \delta: Q \times \Gamma' \times \Sigma' \rightarrow T(Q \times \Gamma')$$

$$\Gamma' = \Gamma \cup \{\lambda\}$$

$$\Sigma' = \Sigma \cup \{\lambda\}$$

۳. سه تایی (q_i, A, a) وضعیت فعلی ماشین را نشان می دهد و دستور العمل $\delta(q_i, A, a) = \{(q_j, B)\}$ بیان می کند که اگر ماشین در حالت q_i, A در بالای پشته قرار داشته باشد و a مقابل نوار خوان باشد با اجرای دستور δ ماشین به حالت q_j می رود، نوار خوان یک مکان راست می رود A از بالای پشته POP شده و B به بالای پشته اضافه می شود. در این ماشین با توجه به چگونگی عملکرد حالات خاص زیر باید مد نظر باشد:

I. اگر به جای A و λ استفاده شود. به معنی صرفنظر کردن از عنصر بالای پشته است و عمل POP انجام نمی شود.

II. اگر به جای a و λ استفاده شود، به معنای این است که بدون خواندن ورودی دستور اجرا می شود.

III. اگر به جای B و λ استفاده شود، عدم اضافه کردن عنصر جدید به بالای پشته را نشان می دهد.

نکته: شرط پذیرش یک جمله در ماشین پشته ای خاتمه ورودی، خالی بودن پشته و توقف در یک حالت نهایی است.

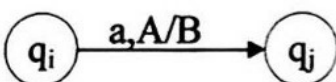
نکته: یک ماشین پشته ای در صورتی متوقف می شود که هیچ دستور العملی شرایط اجرا نداشته باشد.

نکته: در ماشین های پشته ای هر حالت با یک ورودی و یک عنصر بالای پشته می تواند به چند حالت منتقل شود.

$$\delta(q_i, A, a) = \{(q_1, B_1), (q_2, B_2), \dots, (q_n, B_n)\}$$

دیاگرام حالت ماشین

در ماشین پشته ای دیاگرام حالت ماشین گرافیکی است جهت دار که گره های آن حالت ماشین، لبه های آن بیانگر عنصر مقابل نوار خوان، عنصر فعلی بالای پشته و عنصر جدید بالای پشته می باشد.



انواع ماشین پشته ای

ماشین های پشته ای به سه صورت زیر موجود می باشد:

۱. ماشین های پشته ای عادی
۲. ماشین های پشته ای ساده (اتمی)
۳. ماشین های پشته ای پیشرفته

که هر سه نوع ماشین های فوق معادل و قابل تبدیل به یکدیگرند.

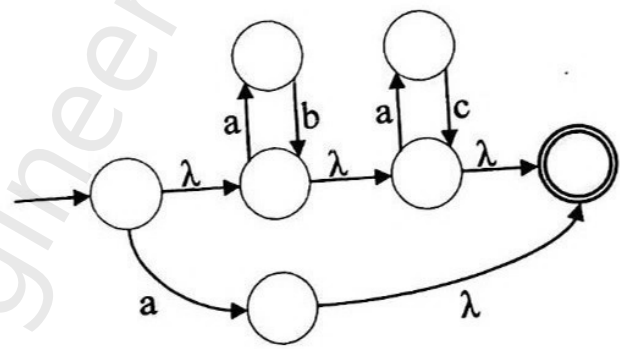
ماشین های پشته ای ساده (اتمی)

ماشینی است پشته ای که در هر مرحله فقط یکی از عملیات های زیر را می تواند انجام دهد:

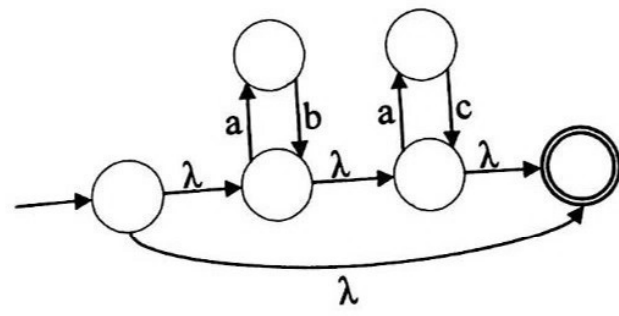
- ا. بخواند و تغییر حالت دهد.
- ب. از پشته بردارد و تغییر وضعیت دهد.
- ج. عنصری به بالای پشته اضافه کند و تغییر حالت دهد.
- د. بدون خواندن ورودی یا کار با پشته تغییر وضعیت دهد.

نکته: ماشین پشته ای ساده حالت خاص ماشین پشته ای ساده است.

مثال ۱: برای زبان $L = (ab)^* (ac)^* \cup a$ کدام ماشین پذیرنده زیر مناسب تر است؟

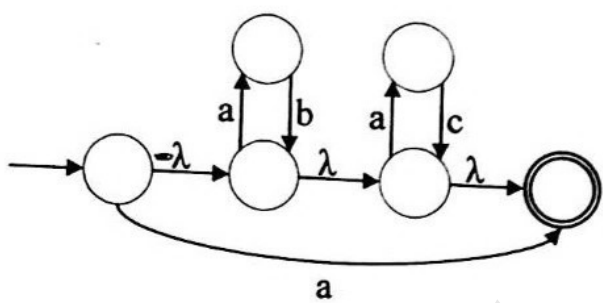


۱.

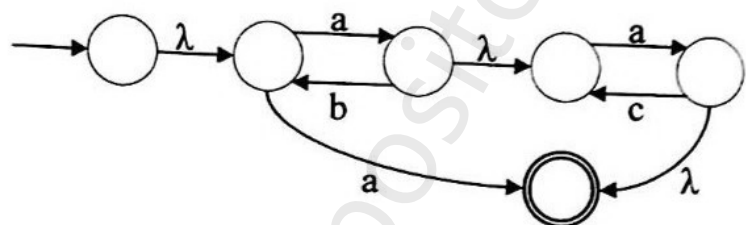


۲.

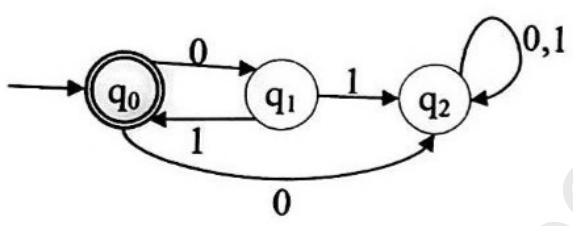
۳.



۴.

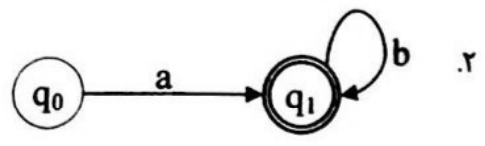
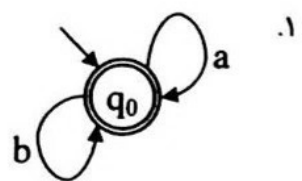


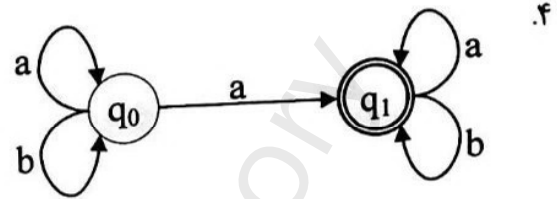
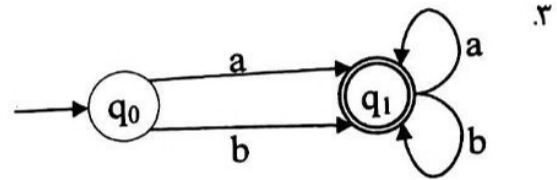
مثال ۲: زبان ماشین ارائه شده کدام است؟



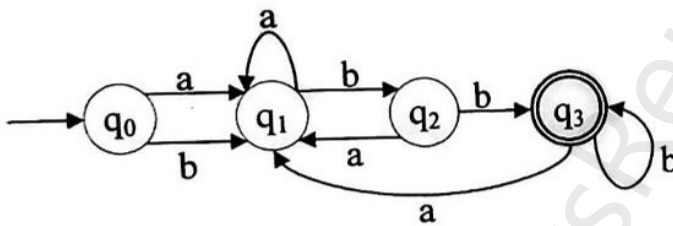
- ۱. $\{(10)^n \cup (011) \mid n \geq 0\}$
- ۲. $\{1(01)^n \mid n \geq 0\}$
- ۳. $\{(10)^n \mid n \geq 0\}$
- ۴. $\{1(01)^n \cup (011)^n \mid n \geq 0\}$

مثال ۳: دیاگرام حالت ماشین DFA پذیرنده زبان $L(G) = (a \cup b)^+$ کدام است؟



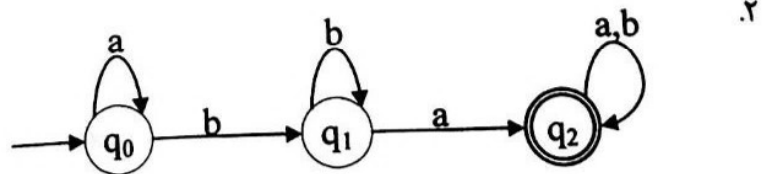
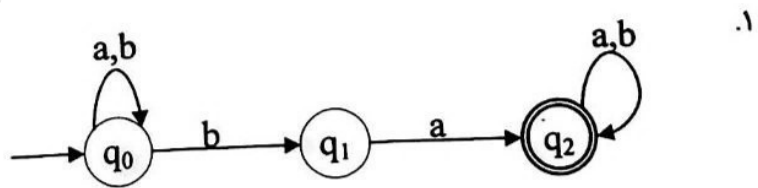


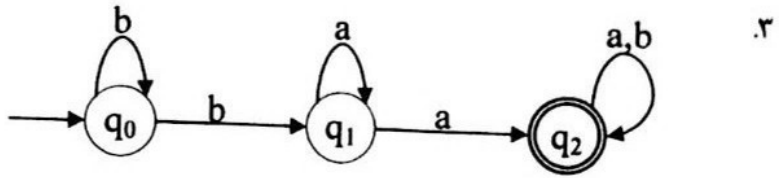
مثال ۴: در دیاگرام حالت ماشین زیر عبارت با قاعده قابل پذیرش کدام است؟



۱. $(a \cup b)^* a^* b a a$
۲. $(a \cup b)^+ b b$
۳. $(a \cup b)^+ b a b b$
۴. ۳ و ۲

مثال ۵: کدام یک از ماشین های NFA یا DFA زیر زبان $(a \cup b)^* b a (a \cup b)^*$ را در $\Sigma = \{a, b\}$ می پذیرد؟





۲ و ۱ .۴

تولید ماشین پشته ای ساده از ماشین عادی

یک ماشین پشته ای ساده به شکل $M = (q_0, \Sigma, Q', \delta', F, \Gamma')$ از ماشین پشته ای عادی با تبدیلات زیر قابل تولید است:

$$\delta(q, A, a) = \{(q_j, B)\} \begin{cases} \delta'(q_i, A, a) = \{(p_1, \lambda)\} \\ \delta'(p_1, A, \lambda) = \{(p_2, \lambda)\} \\ \delta'(p_2, \lambda, \lambda) = \{(p_j, B)\} \\ p_1, p_2 \notin Q \quad p_1, p_2 \in Q' \end{cases}$$

ماشین های پشته ای پیشرفته

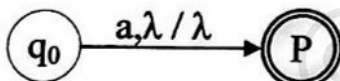
این ماشین قادر است به جای دستیابی به یک عنصر از پشته در یک دستور، دنباله ای از عناصر بالای پشته را دستیابی کند:

$$\delta(q_i, A_1, A_2, A_3, \dots, A_n, a) = \{(q_j, B_1, B_2, \dots, B_m)\}$$

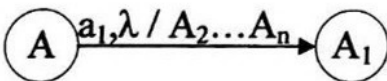
یک ماشین پشته ای عادی حالت خاص ماشین پشته ای پیشرفته است.

تبدیل گرامر مستقل از متن نوع چامسکی و گریباخ به ماشین پشته ای

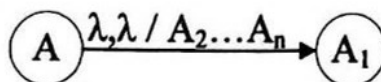
۱. نماد آغازگر S به عنوان حالت ابتدای ماشین منظور می شود. $q_0 = S$
۲. حالت فرضی P به عنوان تنها حالت نهایی ماشین در نظر گرفته می شود. $F = \{P\}$
۳. به ازای هر قانون نظیر $A \rightarrow a$ زیر گراف شکل زیر و یا دستور العمل $(P, \lambda) \in \delta(A, \lambda, \lambda)$ اضافه می شود.



۴. به ازای هر قانون $A \rightarrow a, A_1, A_2, \dots, A_n$ زیر گراف شکل زیر و یا دستور العمل $(A_1, A_2, \dots, A_n) \in \delta(A, \lambda, a)$ اضافه می شود

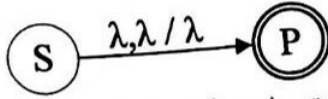


۵. برای هر قانونی نظیر $A \rightarrow A_1 A_2 \dots A_n$ زیر گرافی به شکل زیر و دستور العمل $(A_1, A_2, A_3, \dots, A_n) \in \delta(A, \lambda, \lambda)$

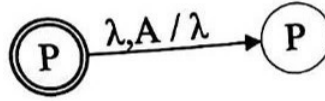


۶. برای قانون $\lambda \rightarrow S$ زیر گراف شکل زیر و دستور العمل:

$$(P, \lambda) \in \delta(S, \lambda, \lambda)$$



۷. در نهایت برای هر عنصر الفبای پشته مانند A زیر گراف و دستور العمل $\delta(P, A, \lambda) = \{(A, \lambda)\}$ اضافه می گردد.



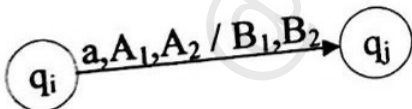
نکته: در مراحل مختلف تبدیل الفبای پشته نیز باید مشخص شود.
نکته: در مرحله ۷ دقیقا می توان نوشت $\delta(P, A, \lambda) = \{(A, \lambda)\}$. ولی در مراحل قبل این امکان وجود نداشت. زیرا در این مرحله فقط امکان انتقال به حالت A وجود دارد با وجود اینکه حالت فعلی P بوده و بالای پشته عنصر A قرار دارد ولی در مراحل قبل امکان وجود قوانین نظیر دیگر موجب می شود که به طور مطلق نتوان انتقال را به یک حالت در نظر گرفت.

ماشین پشته ای تقویت شده

ماشینی است پشته ای که توسط یک پشته اضافی تقویت شده است. به عنوان مثال توسط ماشین های پشته ای بررسی شده نمی توان زبان هایی بصورت $L = a^n b^n c^n$ را محاسبه نمود. ولی در نوع تقویت شده این امکان بوجود آمده است.
در دستور العمل در ماشین پشته ای تقویت شده به صورت زیر می باشد:

$$\delta(q_i, A_1, A_2, a) = \{(q_j, B_1, B_2), \dots\}$$

در این دستور العمل: q_i حالت فعلی، A_1 عنصر فعلی بالای پشته یک، A_2 عنصر فعلی بالای پشته دو، a عنصر ورودی، q_j حالت بعدی، B_1 عنصر جدید بالای پشته یک و B_2 عنصر جدید بالای پشته دو می باشد.
دستور العمل بالا به این معناست که با خواندن a از ورودی در صورت وجود A_1 در بالای پشته یک و A_2 در بالای پشته دو انتقال صورت می گیرد و عناصر یاد شده از بالای پشته برداشته شده و به جای آنها B_1 در بالای پشته یک و B_2 در بالای پشته دو واقع می شوند.
دیگرام حالت در این ماشین به ازای دستور فوق به صورت زیر است.



نکته: عدم قطعیت که در ماشین های پشته ای قبلی وجود داشت در این نوع ماشین نیز وجود دارد.
نکته: شرایط پذیرش در این نوع ماشین خاتمه ورودی، توقف در حالت نهایی و خالی بودن پشته یک و دو می باشند.

نکته: در بعضی حالات بهتر است یا لازم است که خالی بودن پشته در مراحل چک شود، برای رسیدن به این محصول عنصری دلخواه مثلا # (Number Signed) در پشته قرار داده شود و با استفاده از آن خالی بودن پشته تست شود.

تولید گرامر مستقل از متن از ماشین پشته ای
تولید گرامر مستقل از متن از ماشین پشته ای به صورتی انجام می شود که اشتقاق یک جمله از نماد آغازگر محاسبه همان جمله در ماشین شبیه سازی می نماید.

گرامر مستقل از متن طی دو مرحله از ماشین پشته ای $M = \{\Sigma, q_0, Q, \delta, F, \Gamma\}$ با در نظر گرفتن این که $A, B, C \in \Gamma$, $q_i, q_j, q_k, q_l \in Q$, $a \in \Sigma$

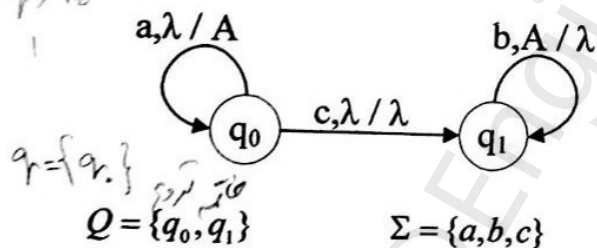
۱. به ازای هر $(q_j, B) \in \delta(q_i, \lambda, a)$ مجموعه دستورات زیر تولید می شود:
 $1 - \delta(q, \lambda, a) = (q_j, B)$
 $2 - \forall A \in \Gamma \rightarrow (q_i, A, a) = \{(q_j, B, A)\}$
۲. هر متغیر از گرامر به صورت $\langle q_i, A, q_k \rangle$ می باشد و این متغیر متناظر است با محاسبه بخشی از جمله ورودی به وسیله عنصر A به طوری که انتقال از q_i به q_k صورت می پذیرد.

برای تولید قوانین گرامر چهار مرحله عملیات انجام می شود:

- I. برای همه $\forall q_f \in F$ قانون $S \rightarrow \langle q_0, A, q_f \rangle$ تولید می شود (q_0 حالت ابتدایی ماشین است).
- II. برای هر دستور $\delta(q_i, A, a) = \{(q_j, B)\}$ قانون $\langle q_i, A, q_k \rangle \rightarrow a \langle q_j, B, q_k \rangle$: $\forall q_k \in Q$ تولید می شود.
- III. برای هر دستور $\delta(q_i, A, a) = \{(q_j, BC)\}$ مجموعه ای از قوانین به صورت زیر تولید می شوند:
 $\langle q_i, A, q_k \rangle \rightarrow a \langle q_j, B, q_l \rangle \langle q_l, C, q_k \rangle$: $\forall q_l, q_k \in Q$
- IV. برای همه $\forall q \in Q$ قانون $\langle q, \lambda, q \rangle \rightarrow \lambda$ تولید می شود.

مثال: گرامر مستقل از متن ماشین پشته ای زیر را بنویسید.

$q_0, q_1 \in Q$
 $\{a, b, c\} \in \Sigma$



$Q = \{q_0, q_1\}$

$\Sigma = \{a, b, c\}$

$\Gamma = \{A\}$

$F = \{q_1\}$

دستورات:

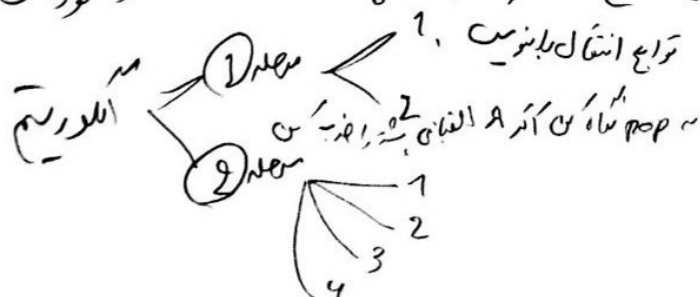
$\delta(q_0, \lambda, a) = \{(q_0, a)\}$

$\delta(q_0, \lambda, c) = \{(q_1, \lambda)\}$

$\delta(q_1, A, b) = \{(q_1, \lambda)\}$

$S \rightarrow \langle q_0, \lambda, q_1 \rangle$

دستورات: $S(q_0, \lambda, a) = \{(q_0, a)\}$ / $S(q_0, \lambda, c) = \{(q_1, \lambda)\}$



$$\delta(q_0, \lambda, a) = \{(q_0, A)\} : \begin{aligned} &\langle q_0, \lambda, q_0 \rangle \rightarrow a \langle q_0, A, q_0 \rangle \\ &\langle q_0, \lambda, q_1 \rangle \rightarrow a \langle q_0, A, q_1 \rangle \end{aligned}$$

$$\delta(q_0, A, a) = \{(q_0, AA)\} : \begin{aligned} &\langle q_0, A, q_0 \rangle \rightarrow a \langle q_0, A, q_0 \rangle \langle q_0, A, q_0 \rangle \\ &\langle q_0, A, q_1 \rangle \rightarrow a \langle q_0, A, q_0 \rangle \langle q_0, A, q_1 \rangle \\ &\langle q_0, A, q_0 \rangle \rightarrow a \langle q_0, A, q_1 \rangle \langle q_1, A, q_0 \rangle \\ &\langle q_0, A, q_1 \rangle \rightarrow a \langle q_0, A, q_1 \rangle \langle q_1, A, q_1 \rangle \end{aligned}$$

میدان دو الفبای است
با دو بار تکثیر شدن

$$\delta(q_0, \lambda, C) = \{(q_1, \lambda)\} : \begin{aligned} &\langle q_0, \lambda, q_0 \rangle \rightarrow C \langle q_1, \lambda, q_0 \rangle \\ &\langle q_0, \lambda, q_1 \rangle \rightarrow C \langle q_1, \lambda, q_1 \rangle \end{aligned}$$

$$\delta(q_0, A, C) = \{(q_1, A)\} : \begin{aligned} &\langle q_0, A, q_0 \rangle \rightarrow C \langle q_1, A, q_0 \rangle \\ &\langle q_0, A, q_1 \rangle \rightarrow C \langle q_1, A, q_1 \rangle \end{aligned}$$

$$\delta(q_1, A, b) = \{(q_1, \lambda)\} : \begin{aligned} &\langle q_1, A, q_0 \rangle \rightarrow b \langle q_1, \lambda, q_0 \rangle \\ &\langle q_1, A, q_1 \rangle \rightarrow b \langle q_1, \lambda, q_1 \rangle \\ &\langle q_0, \lambda, q_0 \rangle \rightarrow \lambda \\ &\langle q_1, \lambda, q_1 \rangle \rightarrow \lambda \end{aligned}$$

مراحل اشتقاق

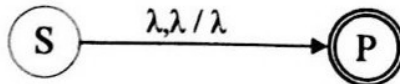
مراحل اشتقاق برای محاسبه جمله aacbb توسط گرامر فوق به صورت زیر است:

$$\begin{aligned} S &\rightarrow \langle q_0, \lambda, q_1 \rangle \xrightarrow{a} \langle q_0, A, q_1 \rangle \rightarrow aa \langle q_0, A, q_1 \rangle \langle q_1, A, q_1 \rangle \\ &\rightarrow aac \langle q_1, A, q_1 \rangle \langle q_1, A, q_1 \rangle \rightarrow aacb \langle q_1, \lambda, q_1 \rangle \langle q_1, A, q_1 \rangle \\ &\rightarrow aacb \langle q_1, A, q_1 \rangle \rightarrow aacbb \langle q_1, \lambda, q_1 \rangle \rightarrow aacbb \end{aligned}$$

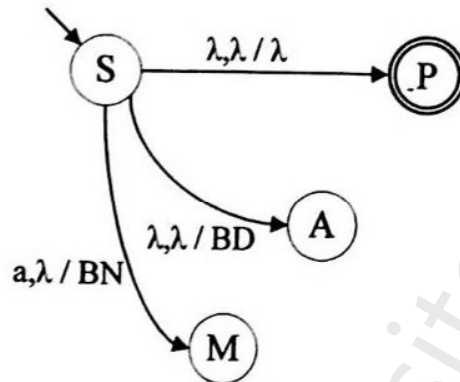
مثال: گرامر مستقل از متن زیر را به ماشین پشته ای تبدیل کنید.

$$\begin{aligned} S &\rightarrow \lambda \\ S &\rightarrow ABD \mid aMBN \\ A &\rightarrow aA \mid aMD \mid a \\ B &\rightarrow bBA \mid b \\ M &\rightarrow DB \mid mM \mid m \\ D &\rightarrow aD \mid d \\ N &\rightarrow nNM \mid n \end{aligned}$$

قانون ۱: $S \rightarrow \lambda$

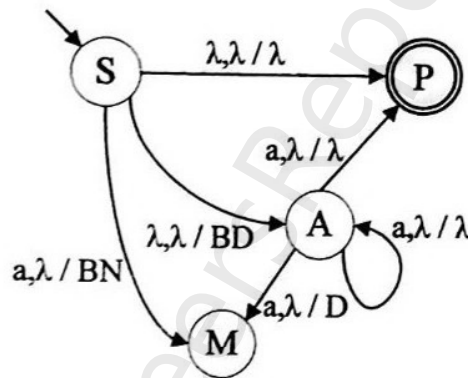


قانون ۲: $S \rightarrow ABD \mid aMBN$

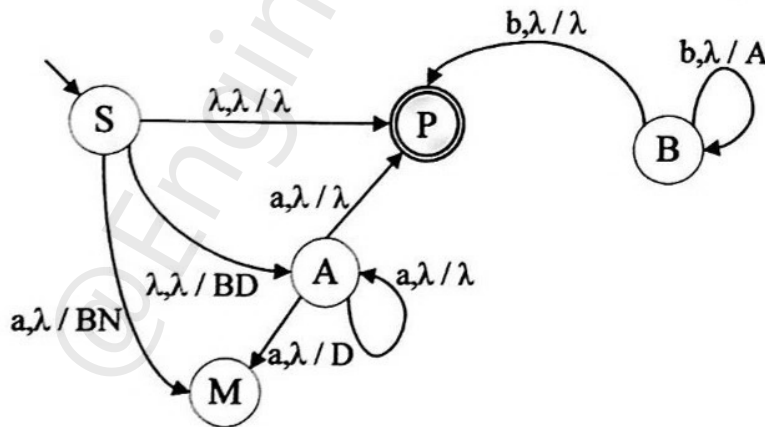


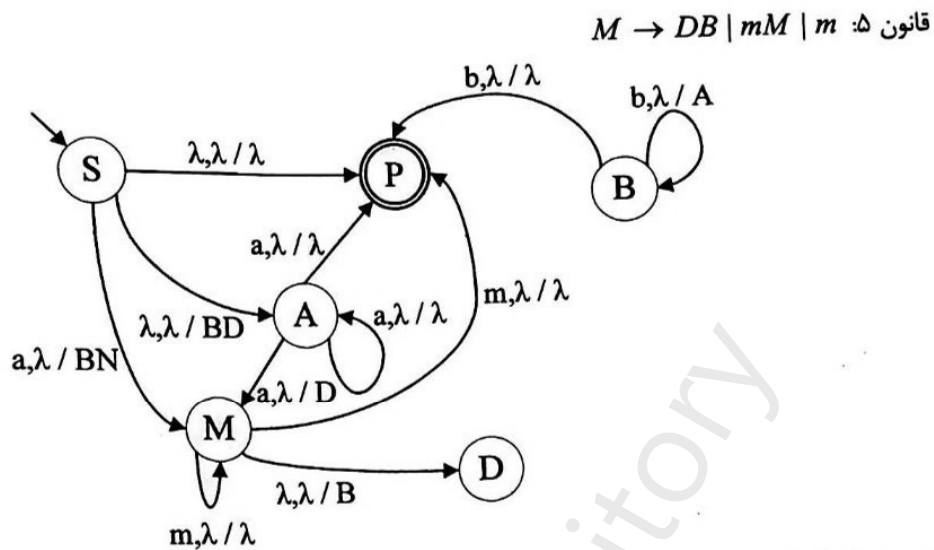
تا این مرحله الفبای پشته به صورت $\Gamma = \{B, N, D\}$ می باشد.

قانون ۳: $A \rightarrow aA \mid aMD \mid a$



قانون ۴: $B \rightarrow bBA \mid b$





@EngineersRepository

a^*ba^*
 $(ba^*b)^*ba^*$ → $\{a, ba^*b, b a^* b b a^* b, b a^* b b a^* b b a^* b, \dots\}$

نمونه سوالات امتحانی آزمون ۷۹/۰۳/۳۰ - واحد جنوب

۱- L زبانی است شامل تجان رشته های a و b که دارای تعدادی فرد از کاراکتر b می باشند. (Regular Expression) مربوط به این زبان کدام است؟

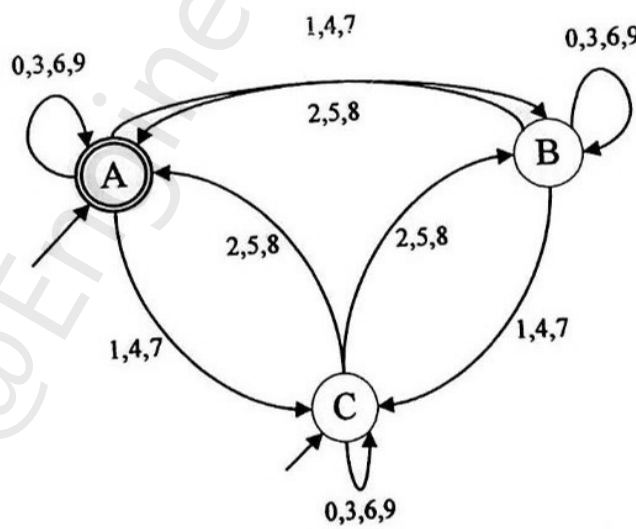
- (۱) $ba^*(a|ba^*b)^*$ فقط با ط شروع می شود (محدودیت)
- (۲) $ab^*(ab^*ab^*)^*$ فقط با a
- (۳) $a^*b(a^*ba^*b)^*$ فقط با b ختم می شود
- (۴) $(a|ba^*b)^*ba^*$

ادیس با A یا b و یا a شروع می شود

۲- زبان گرامر زیر چیست؟
 $S \rightarrow a$
 $A \rightarrow bbA | baV | as | b$
 $B \rightarrow abA | aaB | a | bs$

- (۱) فقط رشته هایی را تولید می کند که در آنها تعداد a ها و b ها برابر می باشند، مانند: aababb
- (۲) فقط رشته هایی را تولید می کند که در آنها تعداد زوجی a و تعداد زوجی b وجود دارند، مانند: aabbaa
- (۳) فقط رشته هایی را تولید می کند که از تعداد فردی a و تعداد فردی b تشکیل شده اند، مانند: baabab
- (۴) فقط رشته هایی را تولید می کند که در آنها تعداد a ها از تعداد b ها بیشتر می باشد، مانند: babaa

۳- ماشین پذیرنده متناهی (Finite State Automate) زیر را در نظر بگیرید. کدامیک از مجموعه رشته های زیر توسط این ماشین پذیرفته می شود؟



- (۱) رشته هایی از اعدادی که بر ۳ بخشپذیرند.
- (۲) رشته هایی از اعدادی که طولشان بر ۳ بخشپذیر است.
- (۳) رشته هایی از اعداد که طولشان بر ۳ بخشپذیر بوده و با یکی از ارقام ۳، ۶، ۹ و ۰ ختم شود.
- (۴) هیچکدام.

نمونه سوالات امتحانی آزمون ۷۹/۱۰/۲۶ - واحد جنوب

« از یازده سوال زیر، به ده سوال پاسخ دهید »

۵- عبارت منظم زیر را در نظر بگیرید:

کلمه a
کلمه a
کلمه a

$$R_1 = b^* a (a+b)^*$$

$$R_2 = (a+b)^* a (a+b)^*$$

$$R_3 = (a^* b^*)^* ab^*$$

$$R_4 = (a+b)^* ab^*$$

هر چهار عبارت تمامی ترکیبات a و b را تولید می کنند
ولیکن اول یک a تسلسل شده اند پس هر چهار عبارت
با هم معادل اند یعنی $R_1 = R_2 = R_3 = R_4$

کدامیک از گزینه های زیر صحیح است؟

- (۱) $R_1 = R_4, R_1 \neq R_2$
- (۲) $R_1 \neq R_4, R_1 = R_2$
- (۳) $R_2 = R_4, R_1 = R_4$ ✓
- (۴) $R_3 \neq R_4, R_2 \neq R_3$

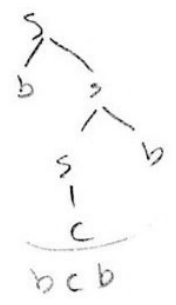
۶- کدامیک از گرامرهای زیر مبهم است؟ (S علامت اولیه می باشد و λ نشانه رشته تهی است.)

(۱) $S \rightarrow AcA$
 $A \rightarrow bA | \lambda$

(۲) $S \rightarrow bS | A$
 $A \rightarrow Ab | c$

(۳) $S \rightarrow bS | c | cA$
 $A \rightarrow bA | b$

(۴) $S \rightarrow bS | Sb | c$ ✓



۷- فرض کنید L_1 زبان عبارت منظم a^* و L_2 زبان عبارت منظم b^* و $L_3 = \{a^n b^n \mid n > 0\}$ باشد و زبان L از الحاق سه زبان فوق بدست آید $(L = L_1 \cdot L_2 \cdot L_3)$. کدامیک از گزینه های زیر درست است؟

- (۱) L یک زبان منظم است.
- (۲) L یک زبان مستقل از متن است و منظم نیست.
- (۳) L یک زبان حساس به متن است و مستقل از متن نیست.
- (۴) هر سه مورد درست است.

۸- با فرض اینکه علامت شروع گرامر زیر نمایانگر رشته تهی باشد، در زبان این گرامر چند رشته وجود دارد که دقیقاً دارای سه پرانتز باز و سه پرانتز بسته است؟

x $S \rightarrow F \rightarrow a$

x $S \rightarrow F \rightarrow (p) \rightarrow ((F))$

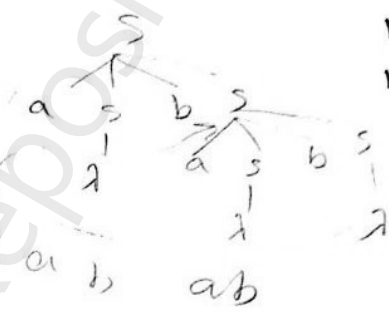
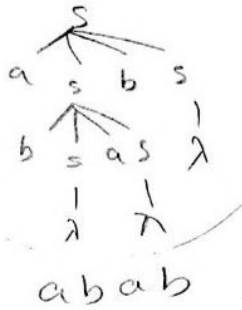
x $S \rightarrow F \rightarrow (p) \rightarrow ((F)) \rightarrow ((:(p))) \rightarrow (((:(:F))))$

$S \rightarrow (E) \rightarrow (S, SH) \rightarrow (S, S) \rightarrow (F, F) \rightarrow (a, (p)) \rightarrow (a, ((:F)))$ ✓

$(p), a \rightarrow ((:F), a)$ ✓

- $S \rightarrow F$
 - $S \rightarrow (E)$
 - $E \rightarrow S, SH$
 - $F \rightarrow a$
 - $F \rightarrow (P)$
 - $H \rightarrow, SH$
 - $H \rightarrow \lambda$
 - $P \rightarrow (: F)$
- ↓
Column

- $S \rightarrow aSbS$
- $S \rightarrow bSaS$
- $S \rightarrow \lambda$



- ۴ (۱)
- ۳ (۲)
- ۲ (۳) ✓
- ۱ (۴)

۹- گرامر

- (۱) زبانی منظم (Regular) را تولید می کند.
- (۲) نا مبهم (Unambiguous) است.
- (۳) مبهم (Ambiguous) است. ✓
- (۴) زبانی را تولید می کند که توسط هیچ ماشین پشته ای قطعی (Deterministic Pushdown Automata) پذیرفته نمی شود.

۱۰- اگر G یک گرامر مستقل از متن و رشته W با طول K متعلق به $L(G)$ باشد، کدامیک از گزینه های زیر نادرست است؟

- (۱) اگر گرامر G به شکل طبیعی چامسکی (Chomsky Normal Form) باشد برای اشتقاق W (Derivation) به $2K-1$ مرحله نیاز است.
- (۲) با حذف قواعدی که به فرم $A \rightarrow B$ هستند اگر از گرامر G ، ممکن است تعداد مراحل اشتقاق رشته W کمتر شود.
- (۳) اگر گرامر G به شکل طبیعی گریباخ (Greibach Normal Form) باشد برای اشتقاق W ، حداقل به K مرحله نیاز است.
- (۴) تعداد مراحل اشتقاق بستگی به فرم گرامر ندارد و قابل پیشبینی نمی باشد.

۱۱- زبانهای مستقل از متن L_1 و L_2 به شرح زیر مفروزند:

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

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

کدام گزینه در مورد زبان $L = \{x \mid xy \in L_1 \text{ and } y \in L_2\}$ درست است؟

(۱) $L = \{a^n b \mid n \geq 0\}$

(۲) $L = \{a^n ba \mid n \geq 0\}$

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

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

۱۲- فرض کنید عبارت منظم $(0+01)^*$ زبان L را نشان می دهد و W^R نمایانگر معکوس رشته W است. زبان $L' = \{W \in L \mid W = W^R\}$

(۱) غیر حساس به متن.

(۲) غیر منظم ولی مستقل از متن است.

(۳) غیر مستقل از متن ولی حساس به متن است.

(۴) منظم است.

برین خصوصیت لیکنون سوال را با سوال من

$$S \rightarrow aSC \mid aS'C$$

$$S' \rightarrow bS'D \mid bS''D$$

$$S''C \rightarrow c$$

$$DC \rightarrow CD$$

$$cC \rightarrow cc$$

$$D \rightarrow d$$

$$s \rightarrow asc \rightarrow a^m s c^m \rightarrow a^n s' c^n \rightarrow a^n s' c^n$$

$$\rightarrow a^n b^m s' c^n d^m \rightarrow a^n b^m s' c^n d^m \rightarrow a^n b^m s' c^n d^m \rightarrow a^n b^m s' c^n d^m$$

$$L = \{w \mid w = a^n b^m c^n d^m, n, m > 0\} \quad (1) \checkmark$$

$$L = \{w \mid w = a^n b^m c^m d^n, n > 0, m > 0\} \quad (2)$$

$$L = \{w \mid w = a^n b^n c^n d^n, n > 0\} \quad (3)$$

(۴) هیچکدام

- I
- $D^3 C^3 \rightarrow C^3 C^3$
 - $DDDC$
 - $DDCC$
 - $DDCC$
 - $CCDD$
 - $CCDD$
 - $CCDD$
 - $CCDD$

- II
- $C^3 C^4 \rightarrow C^4$
 - $CCCC$
 - $CCCC$
 - $CCCC$

۱۴- پذیرفته حالت متناهی مطابق شکل مفروض است. این پذیرفته هم ارز کدام گرامر است؟ (λ نشانه رشته تهی است.)

LL زبان ترمینال شروع می شود
LR زبان ترمینال تمام می شود

MANSOOR

نمونه سوالات امتحانی آزمون ۸۰/۰۳/۲۸ - واحد جنوب

« به ده تست زیر پاسخ دهید (۷،۵ نمره) »

- ۱- مجموعه تمام عبارات منظم (regular expressions) روی الفبای $\{a, b\}$:
 - (۱) یک مجموعه ناشمارا (uncountable) را تشکیل می دهد.
 - (۲) یک زبان منظم (regular) را تشکیل می دهد.
 - (۳) یک زبان مستقل از متن (context-free) تشکیل می دهد.
 - (۴) یک گرامر ذاتاً مبهم (inherently ambiguous) است.

۲- گرامر G را در نظر بگیرید.

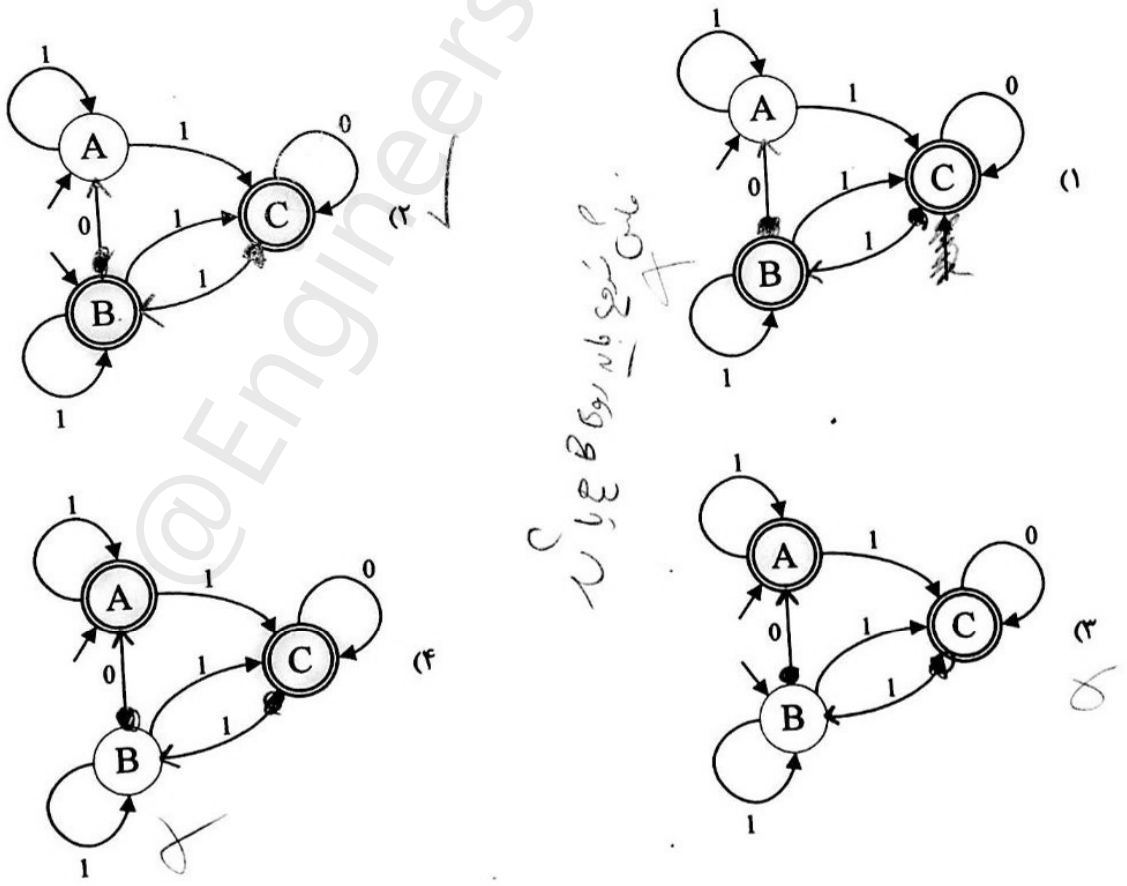
$$G = (\{0,1\}, \{S, A, B, C\}, S, P)$$

P شامل قوانین روبرو می باشد: (منظور از λ رشته به طول صفر می باشد)

- $S \rightarrow \lambda | A | B$
- $A \rightarrow 1A | 1$
- $B \rightarrow 1C | 0A | 1B | 1$
- $C \rightarrow 1B | 0C | 0 | 1$

تست (در جواب)

L(G) توسط کدامیک از ماشین های حالت منتهای (Finite-State-Automata) زیر پذیرفته می شود؟



۳- فرض کنید:

$$L_1 = \{a^p b^q a^p b^s \mid p, q, s \geq 0\}$$

$$L_2 = \{a^p b^p a^r b^s \mid p, r, s \geq 0\}$$

$$L_3 = \{a^p b^q a^r b^p \mid p, q, r \geq 0\}$$

کدامیک از گزینه های زیر صحیح است؟

- (۱) L_1 یک زبان مستقل از متن (context-free) می باشد.
- (۲) L_3 یک زبان مستقل از متن نمی باشد.
- (۳) L_2 یک زبان مستقل از متن نمی باشد.
- (۴) $L_1 \cap L_2 \cap L_3$ یک زبان مستقل از متن می باشد.

S و مساوی که با اول یا آخر فریب شد

- I. $S \rightarrow aSb \mid ab$
- II. $S \rightarrow \overset{RL}{aA} \mid ab$ $A \xrightarrow{LL} Sb$
- III. $S \rightarrow \overset{RL}{aA} \mid ab$ $A \xrightarrow{RL} bS$
- IV. $S \rightarrow \underset{LL}{Ab} \mid ab$ $A \xrightarrow{LL} Sa$

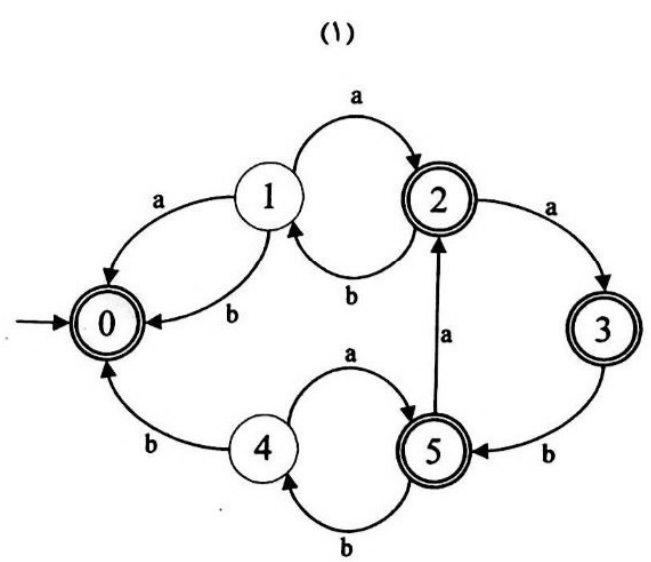
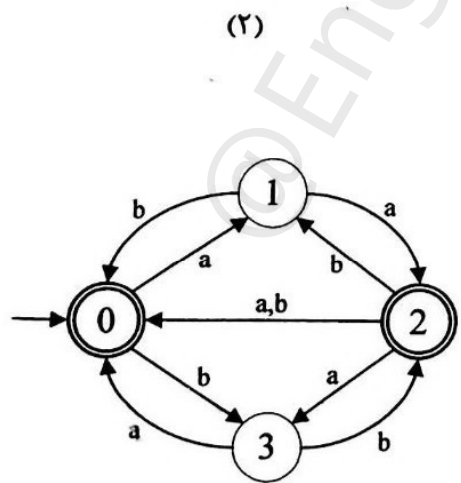
مانع الجمع
مس و راست
نیاید درین گرامر بیاید

۴- در میان گرامرهای زیر، گرامرهای منظم کدامند؟

- (۱) III و IV ✓
- (۲) فقط III
- (۳) همه بجز I
- (۴) هر چهار گرامر

۵- کدامیک از اتومات های متناهی (Finite Automata) زیر زبان L را شناسایی می کند؟ (منظور از $N_a(W)$ تعداد a های محدود در رشته W است).

$$L = W : (N_a(W) - N_b(W) \bmod 5 \neq 1)$$



نمونه سوالات امتحانی آزمون ۸۰/۱۰/۲۴ - واحد جنوب

۱- گرامر زیر را در نظر بگیرید: (۵، ۱ نمره)

$$\begin{aligned} G: S &\rightarrow A|B \\ A &\rightarrow aaB|Aab|Aba \\ B &\rightarrow bB|Bb|aba \end{aligned}$$

الف) یک عبارت با قاعده برای $L(G)$ بنویسید.

ب) یک گرامر معادل G' که شامل هیچ قانون بازگشتی چپ مستقیمی نباشد ایجاد کنید.

۲- گرامر زیر را در نظر بگیرید: (۵، ۲ نمره)

$$\begin{aligned} G: S &\rightarrow AsB|B \\ A &\rightarrow aA|a \\ B &\rightarrow bB|\lambda \end{aligned}$$

الف) یک عبارت با قاعده برای $L(G)$ بنویسید.

ب) نشان دهید که گرامر G مبهم است.

ج) یک گرامر غیر مبهم معادل برای G ایجاد کنید.

۳- به سوالات زیر باختصار پاسخ دهید: (۵، ۲ نمره)

۱-۳) L زبانی است شامل کلیه رشته های a و b که دارای تعداد فردی از کاراکتر b می باشد. عبارت با قاعده مربوط به این زبان را بنویسید.

۲-۳) اگر عبارات منظم $(ab)^*a$ زبان L را تحت الفبای ورودی $\{a, b\}$ تعریف کند، عبارت منظم دیگری بنویسید که تحت همان الفبای ورودی زبان L را تعریف کند.

۳-۳) گرامر $S \rightarrow a$ ، $B \rightarrow b$ ، $S \rightarrow abb$ ، $S \rightarrow ASBB$ جزء کدام طبقه از گرامرهاست؟

۴-۳) زبان $\{a^n b^{2^n} \mid n \leq 2^{100}\}$ جزء کدام نوع از زبانهاست؟

۵-۳) یک الگوریتم تجزیه هنگام بررسی یک جمله به زبان در حلقه نامتناهی قرار گرفته است. در صورتیکه بدانیم جمله متعلق به زبان است، این الگوریتم کدام یک از ۴ الگوریتم پارسر می تواند باشد.

۴- فرض کنید که G گرامر زیر باشد. (۵، ۲ نمره)

$$\begin{aligned} S &\rightarrow aS \\ S &\rightarrow AB \\ A &\rightarrow bAa \\ A &\rightarrow a \\ B &\rightarrow bB \\ B &\rightarrow b \end{aligned}$$

الف) درخت تجزیه ایجاد شده توسط تجزیه بالا به پایین سطحی $baab$ را رسم کنید.

ب) پشته تجزیه بالا به پایین عمقی $baab$ را بررسی کنید.

۵- گرامر فرم نرمال چامسکی زیر را بفرم نرمال گریباخ تبدیل کنید. (۱ نمره)

$$S \rightarrow AB \mid BC$$

$$A \rightarrow AB \mid a$$

$$B \rightarrow AA \mid CB \mid b$$

$$C \rightarrow a \mid b$$

۶- گرامر های مستقل از متنی ایجاد کنید که زبانهای زیر را بپذیرند. (۲ نمره)

$$L(G) = \{a^n b^m c^m d^{2n} \mid n \geq m > 0\}$$

$$L(G) = \{w \mid w \in \{a,b\}^*, |w| \bmod 3 \neq |w| \bmod 2\}$$

۷- به سوالات زیر پاسخ دهید: (۳ نمره)

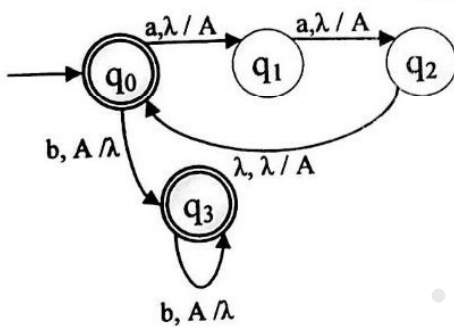
۷-۱) یک FSA (Finite State Automata) زبان $L(M)$ را می پذیرد. چه تغییری در این FSA ایجاد نمائیم تا مکمل $L(M)$ را قبول نماید.

۷-۲) اگر L زبانی باشد که توسط یک آتاماتای متناهی شناخته می شود، آنگاه زبان

$$\text{Reverse}(L) = \{w \text{ such is the reverse of } v \text{ where } v \in L\}$$

معرف چه زبانی است؟

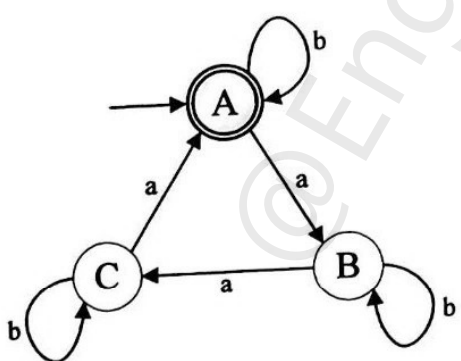
۷-۳) زبان پذیرفته شده توسط یک ماشین پشته ای با دیاگرام حالت زیر کدام است؟



Handwritten notes and diagrams:

- A diagram showing a stack with 'A's and a string 'aaabbb' below it.
- Another diagram showing a stack with 'A's and a string 'aaaaabbb' below it.
- Handwritten expression: $L(M) = \{a^n b^{2n} \mid n \geq 0\}$

۷-۴) گرامری که هم ارز پذیرنده زیر باشد کدام است؟



۸- با استفاده از الگوریتم تبدیل NFA به DFA, DFA به NFA بهینه معادل NFA زیر را رسم کنید. (۲ نمره)

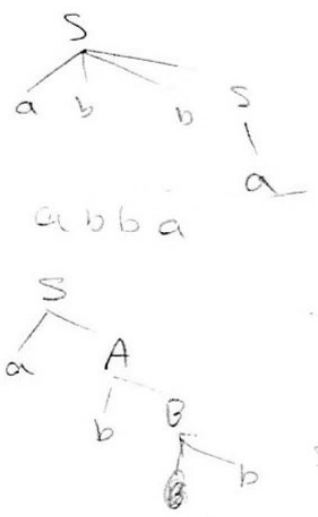
نمونه سوالات امتحانی آزمون ۸۱/۰۴/۰۳ - واحد جنوب

« تست: ۴، ۵ - نمره »

۱- عبارتهای منظم زیر روی القای $\Sigma = \{a, b\}$ تعریف شده اند:

- I. $((a^*b^*)(b^*a^*))^*$
 - × II. $(a^*ba^*)^*$ ← عبارت را فزاینده دند
 - ✓ III. $(a^*b^*) \cup (b^*a^*)$
 - × IV. $((ab)^* \cup (ba)^*)^*$ ← رشته هایی با طول فرد نمی تواند تولید کند
- رشته هایی با طول فرد نمیتواند تولید کند
 مرتبه کبی از
 زوج زوج زوج زوج زوج زوج

- کدامیک از عبارتهای فوق معادل می باشند؟
- IV I (۱)
 - III I (۲) ✓
 - III II (۳)
 - IV III II I (۴) ✓



۲- کدامیک از گرامرهای زیر مبهم است؟

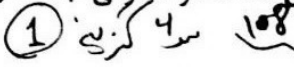
- $S \rightarrow aA \mid a$
- $A \rightarrow bB$ (۱) ✓
- $B \rightarrow bS$
- $S \rightarrow abbS \mid aA \mid a$
- $A \rightarrow bB$ (۲) ✓
- $B \rightarrow bS$
- $S \rightarrow abbSa$ (۳)
- $S \rightarrow Sab \mid b$ (۴)

۳- گرامر $G = (\{S, B, D\}, \{a, b, c\}, S, P)$ را که مجموعه قوانین P آن به صورت زیر است، را در نظر بگیرید:

- $S \rightarrow aSBD$
- $S \rightarrow abD$
- $bB \rightarrow bb$
- $bD \rightarrow bc$
- $DB \rightarrow BD$
- $cD \rightarrow cc$

- (۱) $L(G) = \{a^n b^n c^k \mid n, m, k \geq 1\}$
- (۲) $L(G) = \{a^n b^m c^{n+m} \mid n, m \geq 1\}$
- (۳) $L(G) = \{a^n b^n c^n \mid n \geq 1\}$
- (۴) $L(G) = \{a^n b^n c^m \mid n, m \geq 1\}$

نمونه سوالات امتحانی آزمون ۸۱/۱۰/۲۳ - واحد جنوب

- ۱- ✓ زبان L مربوط به رشته های غیر تهی تشکیل شده از ارقام صفر و یک که دارای سه صفر متوالی نباشد را در نظر بگیرید. عبارت منظمی بنویسید که زبان L را توصیف نماید. (۵، ۱ نمره) 
- ۲- گرامر زیر را در نظر بگیرید: (۲ نمره)

$$\text{bexpr} \rightarrow \text{bexpr or bterm} \mid \text{bterm}$$

$$\text{bterm} \rightarrow \text{bterm and bfactor} \mid \text{bfactor}$$

$$\text{bfactor} \rightarrow \text{not bfactor} \mid (\text{bexpr}) \mid \text{true} \mid \text{false}$$

الف) درخت تجزیه برای عبارت $\text{not}(\text{true or false})$ ایجاد کنید.

ب) آیا این گرامر مبهم است؟

- ۳- گرامر زیر مفروض است: (۵، ۱ نمره)

$$S \rightarrow aSC \mid aS'C$$

$$S' \rightarrow bS'D \mid bS''D$$

$$S''C \rightarrow c$$

$$DC \rightarrow CD$$

$$cC \rightarrow cc$$

$$D \rightarrow d$$

زبان این گرامر چیست؟

- ۴- فرض کنید که G گرامر زیر باشد. (۲ نمره)

$$S \rightarrow aS$$

$$S \rightarrow AB$$

$$A \rightarrow bAa$$

$$A \rightarrow a$$

$$B \rightarrow bB$$

$$B \rightarrow b$$

الف) درخت تجزیه ایجاد شده توسط تجزیه بالا به پایین سطحی $baab$ را رسم کنید.

ب) پشته تجزیه بالا به پایین عمقی $baab$ را بررسی کنید.

- ۵- فرم نرمال چامسکی و گریباخ گرامر زیر را بدست آورید: (۳ نمره)

$$G: S \rightarrow SaB \mid ab$$

$$B \rightarrow bB \mid \lambda$$

و سپس تعلق پذیری $w=abaaba$ را در هر سه فرم معمولی، چامسکی، گریباخ بررسی کنید.

نمونه سوالات امتحانی آزمون ۸۲/۰۴/۰۲ - واحد جنوب

۱- نوع زبانهای زیر را مشخص کنید. (منظم، نام منظم، وابسته به متن، مستقل از متن، بدون محدودیت). (۲،۵ نمره)

1. $L = \{WW : W \in \{a, b\}^*\}$

2. $L = \{W \in \{a, b, c\}^* \mid |W|_a = |W|_b\} \cap \{a^i b^j c^k \mid i, j, k \geq 0\}$

3. $L = \{(ab)^n (cd)^n \mid n \geq 0\}$

4. $L = \{a^n b^m \mid n \geq 1, m \geq 1, nm \geq 1\}$

5. $L = \{a^n b^m \mid m = n \text{ or } m = 3n\}$

۲- برای سوالات زیر یک عبارت با قاعده ارائه دهید که نشان دهنده مجموعه مورد نظر باشد. (۱،۵ نمره)

الف) مجموعه رشته هایی روی $\{a, b, c\}$ که شامل aaa نمی باشد.

ب) مجموعه رشته هایی روی $\{a, b, c\}$ که دقیقاً پس از هر b حداقل c یک بیاید.

۳- یک گرامر مستقل از متن بنویسید که زبان $L = \{x \in \{a, b\}^* \mid x \neq y, x \in \{a, b\}^*\}$ را بپذیرد. (۱ نمره)

۴- با استفاده از الگوریتم پارس یک تعلق جمله $aabbcc$ را به گرامر زیر بررسی کنید. (۱،۵ نمره)

$S \rightarrow aSc \mid A$

$A \rightarrow bAc \mid B$

$B \rightarrow cB \mid \lambda$

$S \rightarrow \lambda \mid AB$

$A \rightarrow S \mid 1A$

$B \rightarrow S \mid 0B$

۵- گرامر مستقل از متن G به شرح زیر مفروض است. (۱ نمره)

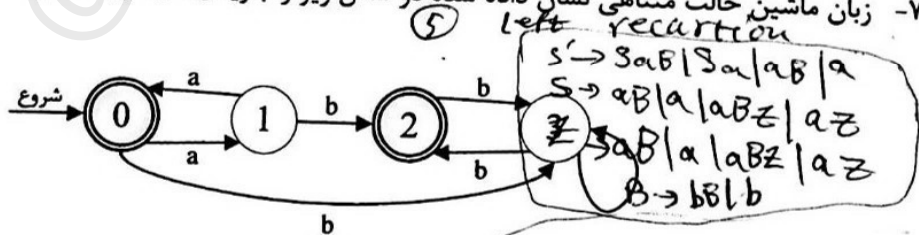
یک عبارت با قاعده برای این زبان بنویسید.

$S \rightarrow SaB \mid aB$

$B \rightarrow bB \mid \lambda$

۶- گرامر زیر را بفرم نرمال چامسکی گریباخ تبدیل کنید. (۲،۵ نمره)

۷- زبان ماشین حالت منتهای نشان داده شده در شکل زیر را بنویسید. (بصورت عبارت منظم). (۱،۵ نمره)



بازرسی روی تمام سوراخها

$S' \rightarrow S$
 $S \rightarrow SaB \mid aB$
 $B \rightarrow bB \mid \lambda$

② λ -production

Null varz $\{B\}$
 $S \rightarrow S$
 $S \rightarrow SaB \mid Sa \mid aB \mid a$
 $B \rightarrow bB \mid b$

③ unit-production

$S' \rightarrow SaB \mid Sa \mid aB \mid a$
 $S \rightarrow S \mid aB \mid Sa \mid bB \mid b$
 $S \rightarrow bB \mid b$

④ useless production

enter

نمونه سوالات امتحانی آزمون ۸۲/۱۰/۲۳ - واحد رودهن

« تست: ۸ نمره »

۱- گرامرهای G_1 و G_2 را در نظر بگیرید:

$$G_1: S \rightarrow SS | (S) | ()$$

$$G_2: S \rightarrow ES | E \\ E \rightarrow (S) | ()$$

در مورد این دو گرامر کدام جمله صحیح است؟

- (۱) گرامر G_1 و G_2 هر دو مبهم و با هم معادل نیستند.
- (۲) گرامرهای G_1 و G_2 هر دو مبهم و با هم معادلند.
- (۳) گرامر G_1 مبهم و گرامر G_2 غیر مبهم است و با هم معادل نیستند.
- (۴) گرامر مبهم G_1 و G_2 گرامر غیر مبهم است و هر دو معادلند.

۲- گرامرهای G_1 و G_2 را بصورت زیر در نظر بگیرید:

$$G_1: S \rightarrow AB \\ A \rightarrow xA \\ A \rightarrow \lambda \\ B \rightarrow yBz \\ B \rightarrow \lambda$$

$$G_2: S \rightarrow AB \\ A \rightarrow xAy \\ A \rightarrow \lambda \\ B \rightarrow zB \\ B \rightarrow \lambda$$

$L = L(G_1) \cap L(G_2)$ برابر است با:

$$S \rightarrow AB \rightarrow x^k A y^m B z^n \rightarrow x^k A y^m B z^n \rightarrow x^k y^m z^n$$

$$A \rightarrow xA \rightarrow x^k A \rightarrow x^k A$$

$$S \rightarrow AB \rightarrow x^k A y^m B z^n \rightarrow x^k A y^m z^n B$$

تعداد n تعداد y و تعداد z = تعداد y

- (۱) $\{x^n y^n z^m \mid n, m = 0, 1, \dots\}$
- (۲) $\{x^n y^n z^n \mid n = 0, 1, \dots\}$ ✓
- (۳) $\{x^m y^n z^n \mid n, m = 0, 1, \dots\}$
- (۴) $\{x^m y^n z^m \mid n, m = 0, 1, \dots\}$

۳- زبان $L = \{a^n b^n c^k \mid n, k \geq 1\}$ را در نظر بگیرید. فرض کنید G گرامری است که $L = L(G)$. کدام

گزینه صحیح است؟

- (۱) $S \rightarrow ABC \mid \lambda \\ A \rightarrow aA \mid \lambda \\ B \rightarrow bB \mid \lambda \\ C \rightarrow cC \mid \lambda \\ S \rightarrow ABC \mid Sc \mid \lambda \\ A \rightarrow aAb \mid \lambda \\ B \rightarrow bBa \mid \lambda$
- (۲) $S \rightarrow ABC \mid \lambda \\ A \rightarrow aA \mid \lambda \\ B \rightarrow bB \mid \lambda \\ C \rightarrow cC \mid \lambda \\ S \rightarrow ABC \mid Sc \mid \lambda \\ A \rightarrow aAb \mid \lambda \\ B \rightarrow bBa \mid \lambda$

$(aa)^*$: جمع a
 $(aa)^* a$: فرد a

a^* : جمع و فرد a

$$S \rightarrow ABS | Sc | \lambda$$

$$A \rightarrow aBb | \lambda \quad (۳)$$

$$B \rightarrow aBbc | \lambda$$

$$S \rightarrow Sc | Cc \quad (۴)$$

$$C \rightarrow aCb | ab$$

۴- زبان L تولید شده توسط عبارت منظم r را در نظر بگیرید که در آن L مجموعه رشته های غیر تهی از ارقام 0 یا 1 که دارای سه 0 متوالی نمی باشند، باشد. کدام گزینه درست است؟
* پاسخ سوال 1 صحیح 103 *

$$(a+b=aUb) \quad 00100100$$

$$((\lambda + (\lambda + 0)0)1)^*(\lambda + 0)(\lambda + 0)10(\lambda + 0) \quad (۱) \checkmark$$

$$\times (00)^*(10+11)^*(\lambda + 1)^*(10)^* \quad (۲)$$

$$\times (00)^*(10+\lambda)^*(\lambda + 1)^*(10+1)^* \quad (۳)$$

$$\times ((\lambda + 0)0)^*(1+10)^*(\lambda + 0)10(\lambda + 1)^* \quad (۴)$$

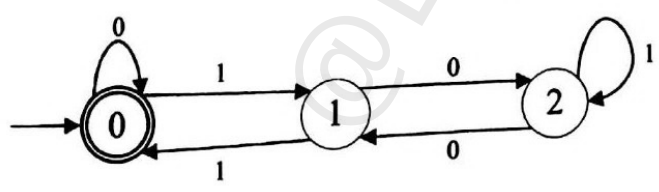
۵- عبارت منظم $(0+1)^*01(0+1)^*+1^*0^*$ با عبارت منظم ... معادل است.

- (۱) 01^* با منزه شروع
- (۲) 0^*1 با یک 0 تمام
- (۳) 0^*1^* با شروع 1 تمام
- (۴) $(0+1)^*$

۶- گزینه نادرست را انتخاب کنید.

- (۱) هر NFA دارای یک NFA معادل با فقط یک حالت پذیرش است.
- (۲) هر PDA دارای یک PDA معادل با فقط یک حالت پذیرش است.
- (۳) هر DFA دارای یک DFA معادل با فقط یک حالت پذیرش است.
- (۴) هیچکدام.

۷- کدامیک از زبانهای داده شده زیر مجموعه ای از زبان اتومات زیر هستند؟



- (۱) $0^*1(10^*1)^*1$
- (۲) $0^*1(10^*1)0(100)^*$
- (۳) $(0^*1(010)^*11)^*$
- (۴) $0^*1(010)11)^*$

نمونه سوالات امتحانی آزمون ۸۲/۱۰/۲۹ - واحد جنوب

« تست: ۹ نمره »

۱- گرامر زیر را در نظر بگیرید:

- $S \rightarrow L | LP | LR$
- $P \rightarrow LL | PLL$
- $R \rightarrow DD | RDD$
- $L \rightarrow a | b | \dots | z$
- $D \rightarrow 0 | 1 | 2 | \dots | 9$

این گرامر را با طول زوج
را در نظر بگیرید

کدامیک از کلمات زیر می‌تواند از تولید شود؟
Word. ۱
Words. ۲
c22.۳

- (۱) فقط ۱ و ۲
- (۲) فقط ۱ و ۳
- (۳) ۱ و ۲ و ۳
- (۴) فقط ۲ و ۳ ✓

$S \rightarrow LR \rightarrow DD \rightarrow c22$

$S \rightarrow LP \rightarrow LPLL \rightarrow LLL LL \rightarrow words$

۲- زبان توصیف شده بوسیله گرامر G کدامیک می‌باشد؟

- $G: S \rightarrow aAb | aBbb$
 $A \rightarrow aAb | ab$
 $B \rightarrow aBbb | abb$

$\rightarrow a^n abbb^{2n}$
 $\rightarrow a^n b^{2n}$

$S \rightarrow aAb \rightarrow a^n A b^n \rightarrow a^n a b b^n \rightarrow a^n b^n$
 $S \rightarrow aBbb \rightarrow a^n B (bb)^n \rightarrow a^n B (b^n)^n$

- (۱) $L(G) = \{a^m b^n | m, n \geq 1 \& n \neq m \& m < n\}$
- (۲) $L(G) = \{a^m b^n | m, n \geq 0 \& n = 2m\}$
- (۳) $L(G) = \{a^k b^k | k \geq 1\} \cup \{a^k b^{2k} | k \geq 1\}$ ✓
- (۴) $L(G) = \{b^m a^n | m, n \geq 0 \& m = 2n\}$

۳- فرض کنید G گرامری به فرم نرمال چامسکی، L زبان تولید شده توسط آن و ω رشته‌ای در L به طول n باشند. اگر طول طولانی‌ترین مسیر جهت دار در درخت اشتقاق متناظر با ω (ارتفاع درخت اشتقاق)، l باشد. آنگاه مجموعه تمامی مقادیر ممکن n کدامیک از مجموعه‌های زیر است؟

- (۱) $\{n \in \mathbb{N} | l - 1 \leq n \leq 2^l\}$
- (۲) $\{n \in \mathbb{N} | l \leq n \leq 2^{l-1}\}$
- (۳) $\{n \in \mathbb{N} | l \leq n \leq 2(l-1)\}$
- (۴) $\{n \in \mathbb{N} | l \leq n \leq 2l\}$

$S \rightarrow aAb \rightarrow a aAb \rightarrow a a a b b b$

« سوالات تشریحی »

۱- یک عبارت با قاعده بنویسید که نشاندهنده مجموعه رشته هایی روی $\{a, b, c\}$ با تعداد فردی از زیر رشته ab باشد. (۱ نمره)

۲- گرامرهای مستقل از متنی بنویسید که زبانهای زیر را تولید کنند. (۱،۵ نمره)

الف) $L(G) = \{a^n b^m c^{2n+m} \mid n, m > 0\}$

ب) $L(G) = \{a^n b^m \mid 0 \leq n \leq m \leq 3n\}$

۳- فرض کنید که G گرامر زیر باشد. (۱،۵ نمره)

1. $S \rightarrow A$

2. $S \rightarrow AB$

3. $A \rightarrow aAb$

4. $A \rightarrow b$

5. $B \rightarrow baB$

6. $B \rightarrow a$

امکان
همان عمل کرد

با استفاده از الگوریتم پارس ۲ تعلق جمله $abbbaa$ را به گرامر مقابل بررسی کنید. (۱،۵ نمره)

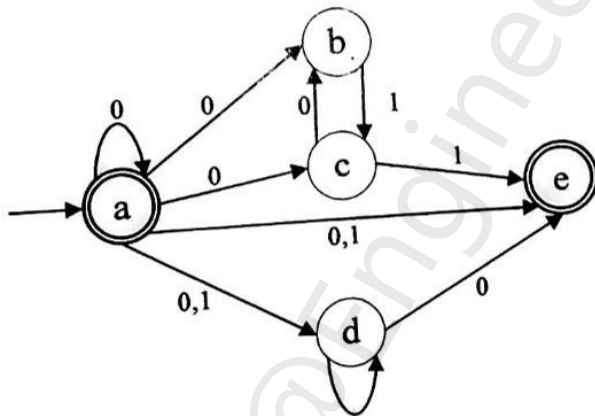
۴- گرامر زیر را بفرم نرمال چامسکی تبدیل کنید. (۱،۵ نمره)

$S \rightarrow A + T \mid b \mid (A)$

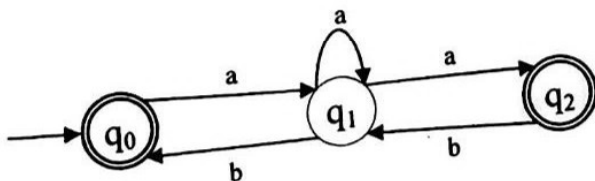
$A \rightarrow A + T \mid b \mid (A)$

$T \rightarrow b \mid (A)$

۵- DFA معادل با NFA زیر را رسم کنید. (۱،۵ نمره)



۶- NFA زیر را در نظر بگیرید. (۱ نمره)



یک گرامر با قاعده ایجاد کنید که زبان ماشین فوق را بپذیرد.

۷- ماشین PDA زیر را در نظر بگیرید. (۲ نمره)

$$Q = \{q_0, q_1, q_2\} \quad \delta(q_0, a, \lambda) = [q_0, A]$$

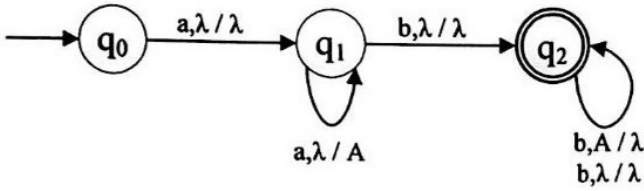
$$\Sigma = \{a, b\} \quad \delta(q_0, b, A) = [q_1, \lambda]$$

$$\Gamma = A \quad \delta(q_1, b, \lambda) = [q_2, \lambda]$$

$$F = \{q_2\} \quad \delta(q_2, b, A) = [q_1, \lambda]$$

گرامر مستقل از متنی بنویسید که زبان ماشین فوق را تولید نماید، سپس تعلق جمله $aabbbb$ را به گرامر بررسی کنید.

۸- زبان پذیرفته شده توسط ماشین پشته ای با دیاگرام حالت زیر را بدست آورید. (۱ نمره)



نمونه سوالات امتحانی آزمون ۸۳/۰۴/۰۲ - واحد جنوب

« تست: ۸ نمره »

۱- عبارت منظم معادل مجموعه ذیل کدام است؟

$$\{a^n b^m \mid n + m \text{ is even}\}$$

(ϵ رشته ای به طول صفر است)

(۱) $a^*(ab)b^*$

(۲) $(aa)^*(\epsilon + ab)(bb)^*$

(۳) $(aa + bb)^*(\epsilon + ab)$

(۴) $(aa)^*(ab)(bb)^*$

سنت $ab = 2$

۲- کدامیک از زبانهای زیر مستقل از متن است؟

(۱) $L = \{a^n b^m c^k \mid k = |n - m|\}$

(۲) $L = \{a^n b^j \mid n \leq j^2\}$

(۳) $L = \{a^n b^j c^k \mid n < j, n \leq k \leq j\}$

(۴) $L = \{a^n b^j c^k \mid k > n, k > j\}$

۳- کدامیک از جملات زیر صحیح است؟

(الف) زبانهای مستقل از متن تحت اشتراک منظم بسته هستند.

(ب) زبانهای مستقل از متن تحت مکمل گری بسته هستند.

(ج) زبانهای مستقل از متن تحت تفاضل بسته هستند.

(د) اگر گرامری مستقل از متن باشد، الگوریتمی برای تصمیم گیری در مورد تهی بودن یا تهی نبودن وجود دارد

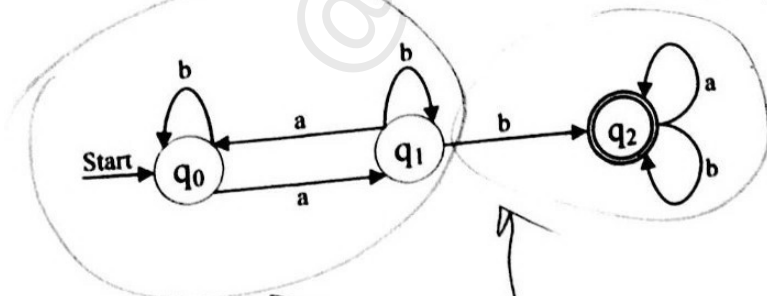
(۱) الف و ب

(۲) الف و د

(۳) ج و د

(۴) الف و ج و د

۴- زبان پذیرفته شده توسط NFA زیر کدام است؟



(۱) $(b^* + ab^*a)ab^*(a+b)^*$

(۲) $((b + aa)ab)^*ab^*(a+b)^*$

(۳) $(b + aa)^*b^*b(a+b)^*$

(۴) $(b + ab^*a)^*ab^*b(a+b)^*$

یا هر دو زوج یا هر دو فرد
زوج زوج
فرد فرد
زوج فرد
فرد زوج

رشته ای که از انتهای کلمه از طرفین به سمت وسط می آید
در هر کلمه دو راه داریم یا به جلو یا به عقب
در هر کلمه دو راه داریم یا به جلو یا به عقب
در هر کلمه دو راه داریم یا به جلو یا به عقب

$$S \rightarrow AS_1 \rightarrow AAS_1b \rightarrow a^n S_1 b^m$$

- $S \rightarrow AS_1 | S_1B$
- $S_1 \rightarrow aS_1b | \lambda$
- $A \rightarrow aA | a$
- $B \rightarrow bB | b$

۵- زبان گرامر زیر چیست؟ (رشته ای به طول صفر است)
 $S \rightarrow AS_1 \rightarrow aAAS_1b \rightarrow a^n A a^m S_1 b^m \rightarrow a^n a^m b^m$
 $S \rightarrow S_1B \rightarrow aS_1bbB \rightarrow a^n S_1 b^n b^m B \rightarrow a^n b^n b^m$
 (Handwritten notes: $a^n b^m$ for $n > m$, $a^n b^m$ for $n < m$, $a^n b^n b^m$ for $n = m$)

- (۱) $\{a^n b^m | n > m\}$
- (۲) $\{a^n b^m | n < m\}$
- (۳) $\{w | w \in (a+b)^*, n \neq m\}$
- (۴) $\{a^n b^m | n \neq m\}$ ✓

۶- کدامیک از زبانهای زیر وابسته به متن است؟

- الف: $\{a^n | n \geq 0\}$
- ب: $\{a^n | n = m^2, m \geq 1\}$
- ج: $\{a^n b^m | n, m \geq 0\}$
- د: $\{a^n | n = 2^k, k \geq 0\}$

- (۱) فقط الف
- (۲) فقط ب
- (۳) الف و ج
- (۴) تمامی زبانها وابسته به متن نیز هستند.

۷- فرض کنید $L_1 = \{a^n b^n a^n b^n : n \geq 0\}$ و $L_2 = \{(ab)^n (cd)^n : n \geq 0\}$ زبانهایی در $\{a, b, c, d\}^*$ باشند. آنگاه ... زبانی مستقل از متن است.

- (۱) L_1
- (۲) L_2
- (۳) $L_1 \cup L_2$
- (۴) $L_1 \cdot L_2$

۸- اگر زبان L بوسیله عبارت منظم $ba(b^*(ab)^* + a^*(bb)^*)ba$ توصیف شده باشد، وارون L (یعنی L^R) بوسیله کدام یک از عبارات منظم زیر بیان می گردد؟

- (۱) $ab((ab)^* b^* + (bb)^* a^*)ab$
- (۲) $ab(a^*(ba)^* + b^*(aa)^*)ab$
- (۳) $ab((ba)^* b^* + (bb)^* a^*)ab$
- (۴) $ab(b^*(ab)^* + a^*(bb)^*)ab$

نمونه سوالات امتحانی آزمون ۸۳/۱۰/۲۸ - واحد جنوب

« تست: ۷.۵ نمره »

- ۱- کدامیک از زبانهای زیر روی الفبای $\{0,1\}$ منظم نیست؟
- (۱) تمامی رشته هایی که پنجمین سمبل آنها از راست صفر است.
- (۲) مجموعه تمامی رشته هایی که به تعداد مساوی صفر و یک دارند. *آر منظم*
- (۳) مجموعه همه رشته هایی که به عنوان یک عدد باینری بر ۱۲ بخش پذیرند.
- (۴) مجموعه تمامی رشته هایی که طول آنها ۱۲ است.

$$S \rightarrow AB$$

$$A \rightarrow a$$

$$A \rightarrow BaB$$

$$B \rightarrow bba$$

۲- گرامر زیر را در نظر بگیرید:

- گزینه نادرست را انتخاب کنید.
- (۱) طول هر رشته ای که این گرامر تولید کند زوج است.
- (۲) این گرامر هیچ رشته ای که در آن ۴ ترمینال پشت سر هم باشند تولید نمی کند.
- (۳) این گرامر هیچ رشته ای با تعداد فرد b تولید نمی کند.
- (۴) در هر رشته ای که توسط این گرامر تولید می شود تعداد b ها حداقل باندازه تعداد a هاست.

۳- گرامرهای G_1 و G_2 با قوانین تولید P_1 و P_2 بصورت زیر را در نظر بگیرید:

$$P_1: S \rightarrow AB$$

$$A \rightarrow X|Y$$

$$X \rightarrow a|aX$$

$$Y \rightarrow c|cY$$

$$B \rightarrow b|bB$$

$$P_2: S \rightarrow aV|cT$$

$$V \rightarrow aV|bB|b$$

$$T \rightarrow cT|bB|b$$

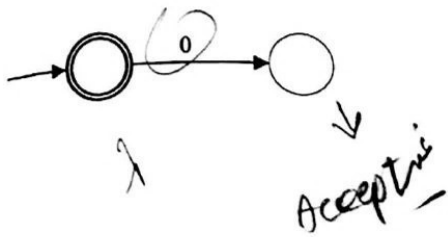
$$B \rightarrow bB|b$$

- گزینه صحیح را در خصوص نوع و رابطه این دو گرامر با هم انتخاب کنید.
- (۱) گرامر G_1 منظم نیست و گرامر G_2 منظم است و با یکدیگر معادل هستند.
- (۲) گرامر G_1 و G_2 منظم هستند و با یکدیگر معادل نیستند.
- (۳) گرامر G_1 منظم نیست و گرامر G_2 منظم است و با یکدیگر معادل نیستند.
- (۴) گرامر G_1 منظم است و گرامر G_2 منظم نیست و با یکدیگر معادل هستند.

۴- فرض کنید L_1 و L_2 زبانهای نوع سوم و L_3 و L_4 زبانهای نوع دوم باشند. کدامیک از عبارات زیر غلط است؟

(۱) زبان $L_2 - L_3 \cup L_4$ یک زبان نوع دوم است.

(۲) زبان $L_2 - L_3$ یک زبان نوع دوم است.



۹- زبان اتوماتون زیر با الفبای $\{0,1\}$ عبارتست از:

- (۱) $\{0\}$
- (۲) $\{0,1\}^*$
- (۳) $\{\lambda, 0\}$
- (۴) $\{\lambda\}$ ✓

۱۰- زبانهای $L_1 = \{a^n \mid n = m^2, m \geq 1\}$ و $L_2 = \{a^n \mid n \text{ اول است}\}$:

- (۱) L_1 مستقل از متن و L_2 وابسته به متن است.
- (۲) هر دو وابسته به متن هستند.
- (۳) L_1 وابسته به متن و L_2 مستقل از متن است.
- (۴) L_1 منظم و L_2 وابسته به متن است.

@EngineersRepublic



۱- یک عبارت با قاعده بنویسید که نشاندهنده مجموعه رشته هایی روی $\{a, b, c\}$ باشد که مجموع تعداد b ها و c ها برابر ۳ است. (۵، ۱ نمره)

۲- یک گرامر روی $\{a, b\}$ ایجاد کنید که زبان آن شامل رشته هایی با طول فرد بوده و عنصر ابتدا و وسط هر رشته مشابه باشد. (۵، ۱ نمره)

۳- فرض کنید که G گرامر زیر باشد. (۲ نمره)

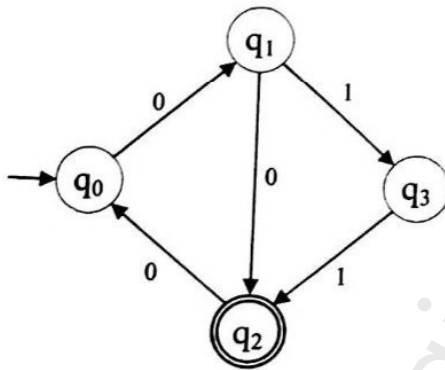
- $S \rightarrow A$
- $A \rightarrow T$
- $A \rightarrow T + T$
- $T \rightarrow b$
- $T \rightarrow (A)$

با استفاده از الگوریتم پارس ۲ تعلق جمله $(a + b)$ به گرامر فوق را بررسی کنید.

۴- گرامر زیر را به فرم نرمال گریباخ تبدیل کنید. (۲ نمره)

- $S \rightarrow BB$
- $A \rightarrow AA | a$
- $B \rightarrow AA | BA | b$

۵- DFA معادل با NFA زیر را رسم کنید. (۲ نمره)

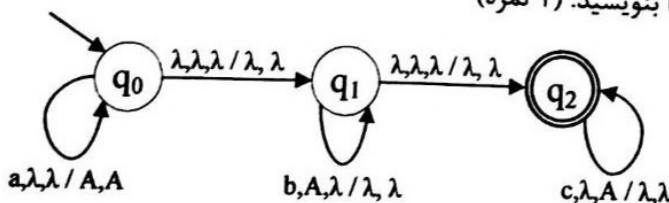


۶- ماشین زیر را در نظر بگیرید:

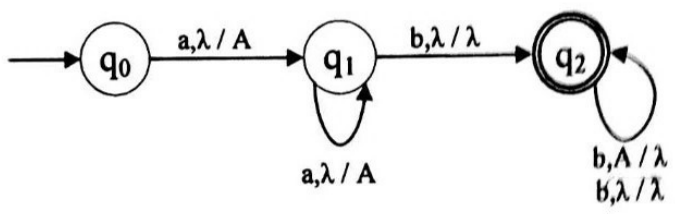
- $Q = \{q_0, q_1, q_2\}$ $\delta(q_0, a, \lambda) = [q_0, A]$
- $\Sigma = \{a, b\}$ $\delta(q_0, b, A) = [q_1, \lambda]$
- $\Gamma = A$ $\delta(q_1, b, \lambda) = [q_2, \lambda]$
- $F = \{q_2\}$ $\delta(q_2, b, A) = [q_1, \lambda]$

یک گرامر مستقل از متن G بسازید که $L(M)$ را تولید کند.

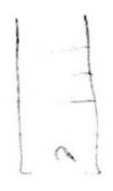
۷- زبان پذیرفته شده توسط ماشین پشته ای زیر را بنویسید. (۱ نمره)



۷- ماشین پشته ای با دیاگرام حالت زیر چه زبانی را می پذیرد؟ (۱.۵ نمره)



$b = A^n$
 ...
 ...



$a a^n b b^n$

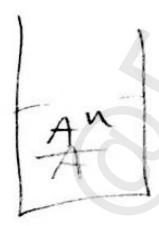
$$L(m) = \{ a^n b^m \mid m \geq n \geq 1 \}$$

برای $n=1$

MANSOOR

حلی

DFA و NFA



برای $n=1$

$$a, a^n b \quad b^n b = a^n b^{n+1} b$$

$$L(m) = \{ a^n b^m \mid m \geq n \geq 1 \}$$

ف $a b b$