ضمیمه‌های ۱ و ۲ "کتاب جامع نظریهٔ زبان‌ها و ماشینها"

ضمیمهٔ ۱ خلاصه‌ای از کتاب پیتر لینز را در حدود ۳۰ صفحه خلاصه کرده است!. این ضمیمه برای جمع-بندی بسیار مناسب است. در ضمیمهٔ ۲، تعدادی از زبان‌های نوع اول، دوم و سوم برای راحتی دانشجو آورده شده است. این کتاب دارای ۸ فصل و ۵ ضمیمه می‌باشد که ضمیمه‌های ۱ و ۲ آن را در این کتاب مشاهده می‌کنید. اشتراک‌گذاری و کپی‌برداری از این فایل PDF به هر نحو مجاز بوده و نیازی به اجازه گرفتن از مولف یا انتشارات را ندارد.

بعضی از ویژگی‌های این کتاب عبارتند از:

- تشریح دروس در ۸ فصل.
- تمرین‌های پایان هر فصل و پاسخ به تمرین‌های انتخابی در یکی از ضمیمه‌های کتاب.
- دو نمونه سوال از امتحانات پایان ترم دانشگاه پیام‌نور همراه با پاسخ تشریحی.
- استفاده از مثال‌های کاربردی در متن کتاب.
- استفاده از ۲۰۰ تصویر برای تفهیم بهتر مطالب.
- ضمیمه‌های ۱ و ۲ که در بالا به آن اشاره شد.
- مثال‌های متنوع برای تفهیم بهتر ماشین‌های تورینگ (فصل ۸).

بسیاری از این ویژگی‌ها برای اولین بار در این کتاب استفاده شده است.

مولف:

حسین ضیائی نافچی

ضمیمه ۱. خلاصه کتاب پیتر لینز

فصل اول: مقدمه‌ای بر تئوری محاسبات

۱) $S_1 \times (S_2 \cup S_3) = (S_1 \times S_2) \cup (S_1 \times S_3)$ **۲)** اگر $S_1 \subseteq S_2$، آنگاه $\overline{S}_2 \subseteq \overline{S}_1$

۳) الحاق = اتصال **۴)** معکوس $w \leftarrow w^R$

۵) اگر $w = vu \leftarrow v$ پیشوند و u پسوند **۶)** $\overline{L} = \Sigma^* - L$

۷) اگر $L = \{a^n b^n : n \geq 0\}$ آنگاه $L^2 = \{a^n b^n a^m b^m : n \geq 0, m \geq 0\}$ (که n و m غیر وابسته‌اند) و $L^R = \{b^n a^n : n \geq 0\}$

مثال : گرامر G با قوانین تولید $S \rightarrow bSa$ ،$S \rightarrow aSb$ ،$S \rightarrow \lambda$ ،$S \rightarrow SS$ زبان $L = \{w : n_a(w) = n_b(w)\}$ را تولید می‌کند.

۸) دو گرامر G_1 و G_2 معادل هستند، اگر آنها زبان یکسانی را تولید نمایند. یعنی $L(G_1) = L(G_2)$.

۹) $a^R = a$ ، $(wa)^R = aw^R$ ، $(uv)^R = v^R u^R$

۱۰) برای همهٔ زبانها : $(L_1 L_2)^R = L_2^R L_1^R$ ، $(L^*)^* = L^*$

مثال : همهٔ رشته‌هایی که حداقل سه a دارند: $A \rightarrow aA|bA|\lambda$, $S \rightarrow AaAaAaA$

مثال : $S \rightarrow \lambda$ ،$S \rightarrow aA$ ،$A \rightarrow bS$ زبان $L(G) = \{(ab)^n : n \geq 0\}$ را تولید می‌کند.

مثال : گرامر $L = \{a^n b^{n-3} : n \geq 3\}$ چیست؟ $L = \{a^{m+3} b^3 : m \geq 0\} \leftarrow$

$A \rightarrow aAb|\lambda$ ، $S \rightarrow aaaA$

مثال : گرامر $L = \{w : |w| \bmod 3 > 0\}$ چیست؟ دو حالت داریم :

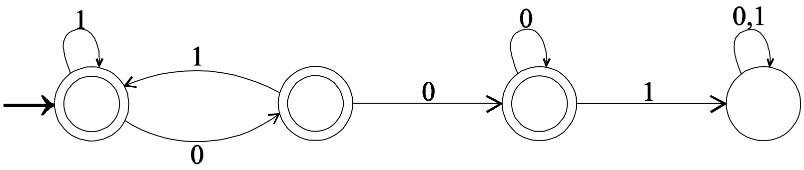
$|w| \bmod 3 = 1$ و $|w| \bmod 3 = 2$

$\leftarrow S \rightarrow S_1 | S_2, S_1 \rightarrow aaaS_1 | a, S_2 \rightarrow aaaS_2 | aa$

فصل دوم: ماشینهای متناهی

۱) پذیرنده متناهی قطعی(dfa)

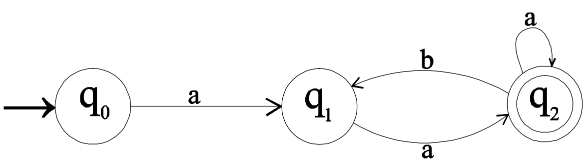
مثال : یک پذیرنده متناهی قطعی بیابید که همهٔ رشته‌های ورودی روی $\{0,1\}$ بـه جـز آنهایی که شامل زیر رشته 001 باشند را بپذیرد :



۲) زبان L را منظم گویند اگر و فقط اگر پذیرنده متناهی قطعیM وجـود داشـته باشـد بطوریکه L=L(M)

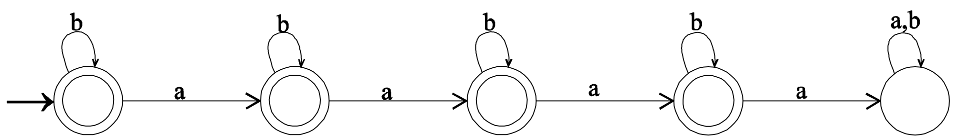
مثال : نشان دهید که زبان $L=\left\{awa : w \in \{a,b\}^*\right\}$ منظم است : یک پذیرنده متناهی قطعی برای آن می‌یابیم :

(بعضی از حالتها حذف شده‌اند)



مثال : زبان $L^2=\left\{aw_1aaw_2a : w_1, w_2 \in \{a,b\}^*\right\}$ نیز منظم است.

مثال : روی $\Sigma=\{a,b\}$ پذیرنده متناهی قطعی بسازید که همه رشته‌هایی کـه بـیش از سه a ندارند را بپذیرد :



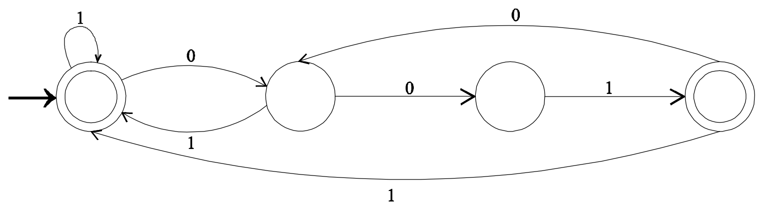
مثـــــال : ایـــن دو زبـــان مـــــنظم‌انـــد : $L=\left\{ab^5wb^4 : w \in \{a,b\}^*\right\}$ و

$L=\left\{w_1abw_2 : w_1 \in \{a,b\}^*, w_2 \in \{a,b\}^*\right\}$

۳) اگر $M=(Q,\Sigma,\delta,q_\circ,F)$ و $M'=(Q,\Sigma,\delta,q_\circ,Q-F)$ دو پذیرنده متناهی قطعی باشند، آنگاه $\overline{L(M)}=L(M')$.

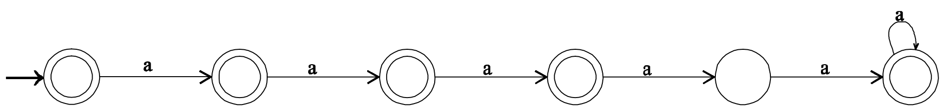
مثال : ماشین متناهی قطعی زبان $L=\{w:n_a(w) \bmod 3 > n_b(w) \bmod 3\}$ دارای ۹ حالت است.

مثال : مجموعه رشته‌ها روی $\{0,1\}$ به طوری که پس از هر 00 بلافاصله یک 1 بیاید : (تله حذف شده است)



مثال : چند زبان منظم: $L=\{vwv:v,w\in\{a,b\}^*,|v|=2\}$ ، $L=\{a^n:n\geq 4\}$

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



مثال : مجموعه همه اعداد حقیقی در C یک زبان منظم است.

مثال : زبان $L=\{a^n:n=i+jk,\quad i,k$ ثابت $,j=0,1,2,\ldots\}$ منظم است.

مثال : اگر L منظم باشد آنگاه $L-\{\lambda\}$ نیز منظم است.

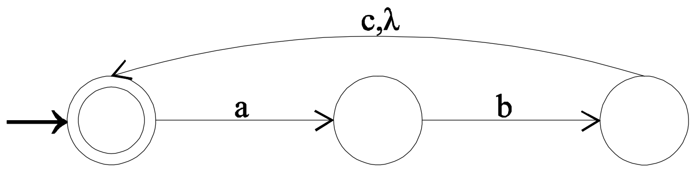
مثال : فرض کنید عمل قطع که سمت راست‌ترین نماد از رشته را حذف می‌کند را تعریف کرده‌ایم (مثلاً $aaaba \to aaab$) در اینصورت اگر L زبان منظمی باشد که شامل رشته λ نباشد آنگاه عمل قطع روی L منظم است.

۴) زبان پذیرفته شده توسط یک پذیرنده متناهی قطعی داده شده، منحصر به فرد است اما به طور معمول پذیرنده‌های متناهی قطعی زیادی وجود دارند که یک زبان را می‌پذیرند.

۵) پذیرنده متناهی غیرقطعی (nfa) : غیر قطعیت به معنای انتخاب حرکات در یک ماشین می‌باشد. سه تفاوت اساسی بین پذیرنده متناهی قطعی و غیرقطعی وجود دارد.

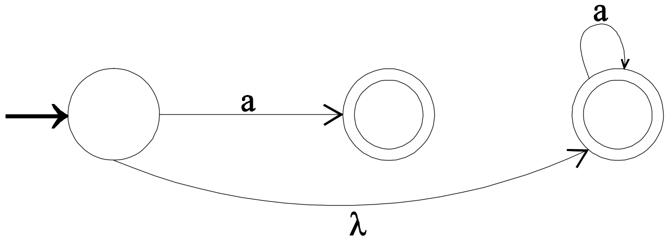
۶) ماشینهای غیرقطعی می‌توانند به عنوان مدلهایی از الگوریتم‌های جستجو و بازگشت به عقب عمل کنند.

مثال : یک پذیرنده متناهی غیرقطعی که زبان $\{ab,abc\}^*$ را بپذیرد بسازید :



مثال : یک زبان منظم : $L=\{a^n : n\geq 1\}\cup\{b^m a^k : m\geq 0, k\geq 0\}$

مثال : یک پذیرنده متناهی غیرقطعی بیابید که $\{a\}^*$ را بپذیرد به طوری که اگر در گراف انتقال آن یک یال تنها حذف شود (بدون هیچ تغییر دیگری)، ماشین بدست آمده $\{a\}$را بپذیرد :



۷) برای هر پذیرنده متناهی غیرقطعی با چندین حالت اولیه یک پذیرنده متناهی غیرقطعی با یک حالت اولیه تنها وجود دارد که همان زبان را می‌پذیرد.

۸) قضیه : فرض کنید L زبان پذیرفته شده توسط یک پذیرنده متناهی غیرقطعی M_N باشد، در اینصورت یک پذیرنده متناهی قطعی M_Dوجود دارد بطوریکه : $L=L(M_D)$

نتیجه مهم : هر زبان پذیرفته شده توسط یک پذیرنده متناهی غیرقطعی، منظم است.

نکته : زبان L پذیرفته شده توسط یک پذیرنده متناهی غیرقطعی $M=(Q,\Sigma,\delta,q_\circ,F)$ به عنوان مجموعه همه رشته‌های پذیرفته شده با این مفهوم تعریف می‌شود : $L(M)=\left\{w\in\Sigma^*:\delta^*(q_\circ,w)\cap F\neq\phi\right\}$ و

$$\overline{L(M)}=\left\{w\in\Sigma^*\middle|\delta^*(q_\circ,w)\cap(Q-F)\neq\phi\right\}$$

۹) برای هر پذیرنده متناهی غیرقطعی با تعداد دلخواه حالات نهایی، یک پذیرنده متناهی غیرقطعی معادل با فقط یک حالت نهایی وجود دارد. این امر عموماً برای پذیرنده‌های متناهی قطعی برقرار نیست.

۱۰) فرض کنید L زبان منظمی باشد که شامل λ نیست، یک پذیرنده متناهی غیرقطعی بدون انتقال λ و با تنها یک حالت نهایی وجود دارد که L را می‌پذیرد.

۱۱) همه زبانهای متناهی منظم هستند.

۱۲) اگر L منظم باشد آنگاه L^R نیز منظم است.

۱۳) در بحث کاهش تعداد حالات در ماشینهای متناهی: قضیه: رویه علامتگذاری، به هر پذیرنده متناهی قطعی M اعمال شود، خاتمه یافته و همه زوجهای حالات متمایز را تعیین می‌کند.

مثال : چند زبان منظم : $L=\left\{a^nb^m:n\geq 2,m\geq 1\right\}$ ، $L=\left\{a^nb:n\geq 0\right\}$
$L=\left\{a^n:n\geq 0,n\neq 3\right\}$ ، $L=\left\{a^n:n\neq 2,n\neq 4\right\}$

۱۴) اگر $M=\left\{Q,\Sigma,\delta,q_0,F\right\}$ یک پذیرنده متناهی قطعی کمینه برای زبان منظم L باشد، آنگاه $\overline{M}=\left\{Q,\Sigma,\delta,q_\circ,Q-F\right\}$ یک پذیرنده قطعی کمینه برای $\bar{L}$ نخواهد بود.

۱۵) اگر حالات q_a و q_b نامتمایز باشند، و اگر q_a و q_c متمایز باشند، آنگاه q_b و q_c باید متمایز باشند.

فصل سوم: زبانهای منظم و گرامرهای منظم

۱) اگر r_1 و r_2 عبارات منظم باشند آنگاه :

$$L(r.r_2)=L(r_1)L(r_2) \quad , \quad L(r_1^*)=(L(r_1))^* \ ,$$
$$L(r_1+r_2)=L(r_1)\cup L(r_2)$$

مثال : عبارت $r=(aa)^*(bb)^*b$ نمایانگر مجموعه‌ای از رشته‌ها با تعداد زوج a است که بوسیله تعداد فردی b دنبال می‌شود : $L(r)=\{a^{2n}b^{2m+1}: n,m\ge 0\}$

مثال*: برای $\Sigma=\{0,1\}$ یک عبارت منظم r بدهید بـه گونـه‌ای کـه $\{$ w دارای حـداقل یک زوج صفر متوالی باشد : $L(r)=\{w\in\Sigma^*$ ، $r=(0+1)^*00(0+1)^*$

مثـــال : یـک عبـارت مـنظم بـرای زبـان $\{w$ دارای هـیچ زوج صـفر متـوالی نباشـد : $L=\{w\in\{0,1\}^*$ بیابید : $r=(1^*011^*)^*(0+\lambda)+1^*(0+\lambda)$ ،

یک جواب کوتاهتر عبارت است از : $r=(1+01)^*(0+\lambda)$

مثال : یک جواب دیگر برای مثال* عبارت است از : $((0+1)(0+1)^*)^*00(0+1)^*$

مثال : چند زبان منظم : $L_1=\{a^nb^m, n\ge 4, m\le 3\}$ ، $L_2=\{a^nb^m : n<4, m\le 3\}$ ، $L_3=\{a^nb^m : n\ge 1, m\ge 1, nm\ge 3\}$ ،

مثال : عبارت منظم زبان $L=\{a^nb^m : n\ge 4, m\le 3\}$: $aaaaa^*(\lambda+b+bb+bbb)$

مثال: عبارت منظم مکمل زبان مثال فوق: {همه رشته‌هایی که حـداقل دارای یـک ba باشند} $\cup$ $\bar{L}_1=\{a^nb^m : n<4 \text{ or } m>3\}$ عبارت است از:

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

مثال : چند زبان منظم :

$$L=\{vwv : v,w\in\{a,b\}^*, |v|=2\} \ ، \ L=\{ab^nw : n\ge 3, w\in\{a,b\}^+\}$$

مثال : روی $\Sigma=\{a,b,c\}$ عبارت منظمی بنویسید که همهٔ رشته‌هایی که از هر نماد در Σ حداقل یک رخداد را دارا باشند :

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

این جواب کامل نیست زیرا a مقدم بر b خواهد بود و مانند آن. برای راه حل کامل باید همهٔ جایگشت‌های سه نماد را تولید کرد که شش عبارت بدست می‌آید.

مثال : بر روی $\Sigma = \{0,1\}$ عبارت منظمی بنویسید که شامل همهٔ رشته‌هایی که شامل تعداد زوج از 0 ها باشند، باشد :

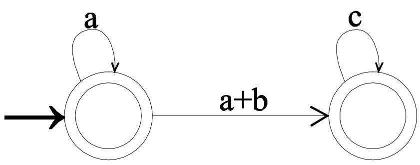
$$(1^*01^*01^*)^* + 1^*$$

مثال : عبارت منظم $L = \{w \mid |w| \bmod 3 = 0\}$ را روی $\{a,b\}$ بیابید :

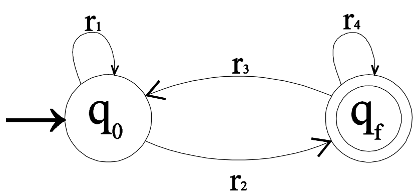
$$\left((a+b+c)(a+b+c)(a+b+c)\right)^*$$

نکته : برای هر زبان منظم یک عبارت منظم وجود دارد و برای هر عبارت منظم یک زبان منظم وجود دارد.

مثال : $L(a^* + a^*(a+b)c^*)$



مثال : $r = r_1^* r_2 (r_4 + r_3 r_1^* r_2)^*$



۲) گرامرهای خطی راست و خطی چپ : گرامر G را خطی راست گویند اگر همه قواعد به شکل $A \to xB$ و $A \to x$ باشد.

یک گرامر را خطی چپ گویند اگر همه قواعد آن به شکل $A \to Bx$ یا $A \to x$ باشند.

یک گرامر منظم گرامری است که یا خطی راست و یا خطی چپ باشد. در گرامر منظم، حداکثر یک متغیر در سمت راست هر قانون ظاهر می‌شود، به علاوه متغیر باید همواره سمت راست‌ترین یا چپ‌ترین نماد در سمت راست هر قانون باشد.

مثـــال : گرامــر $S \to abS|a$ خطــی راســت اســت. گرامــر $S_2 \to a$ **و** $S_1 \to S_1ab|S_2$، $S \to S_1ab$ خطی چپ است و هر دو گرامر منظم هستند.

مثال : گرامر $B \to Ab$ ، $A \to aB|\lambda$ ، $S \to A$ منظم نیسـت چـون نـه خطـی راسـت است و نه خطی چپ است.

۳) یک گرامر خطی گرامری است که حداکثر یک متغیر می‌تواند در سـمت راسـت هـر قانون آن ظاهر شود، بدون اینکه محدودیتی در محل قرار گرفتن این متغیر وجود داشته باشد. واضح است که یک گرامر منظم، همیشه خطی است ولی همـه گرامرهـای خطـی، منظم نیستند. مثال فوق یک گرامر خطی است.

۴) زبان تولید شده توسط یک گرامر خطی راست، همواره منظم است. (قضیه)

۵) قضیه : اگر L یک زبان منظم روی الفبای Σ باشد، آنگاه یک گرامر خطی راسـت G وجود دارد بطوریکه $L = L(G)$.

۶) قضیه : زبان L منظم است اگر و فقط اگر یک گرامر خطی چپ G وجود داشته باشد بطوریکه $L = L(G)$ باشد.

۷) چندین روش برای توصیف زبانهای منظم : پذیرنده‌های متناهی قطعی، پذیرنده‌هـای متناهی غیرقطعی، عبارات منظم و گرامرهای منظم.

۸) یک گرامر خطی چپ می‌توانـد مسـتقیماً از روی یـک پذیرنـده متنـاهی غیرقطعـی بدست آید. (ساختاری وجود دارد که این کار را انجام می‌دهد.)

مثال : این زبان منظم است : $L = \{w : |n_a(w) - n_b(w)|$ فرد باشد $\}$

۹) قواعد گرامر منظم به فرم زیر است :

$$A \to \lambda \quad \text{OR} \quad A \to A'u \quad \text{OR} \quad A \to u \ , \ u \in \Sigma^*$$

فصل۴: ویژگی‌های زبان منظم

۱) قضیه : اگر L_1 و L_2 زبانهای منظم باشند، آنگاه $\overline{L_1}$، L_1L_2، $L_1 \cap L_2$، $L_1 \cup L_2$ و L_1^* نیز منظم هستند. گوییم که خـانواده زبانهـای مـنظم تحـت اجتمـاع، اشـتراک، اتصـال، متمم‌گیری و بستار ستاره‌ای بسته است.

۲) خانواده زبانهای منظم تحت تفاضل بسته است :

منظم $L_1 - L_2 = L_1 \cap \overline{L}_2 \rightarrow$ (ضرورتاً منظم است.)

۳) خانواده زبانهای منظم تحت معکوس کردن بسته است. (در ماشینهای متناهی رأس اولیه را رأس نهایی و رأس نهایی را رأس اولیه می‌سازیم.)

۴) تعریف : فرض کنید Σ و Γ الفبا باشند. در این صورت تابع $h:\Sigma \rightarrow \Gamma^*$ را همریختی گویند. به عبارتی، همریختی یک جایگزینی است که یک حرف تنها با یک رشته جایگزین می‌شود. دامنه تابع h به رشته‌ها با روشی روشن توسعه می‌یابد.

اگر $w = a_1a_2....a_n$ آنگاه $h(w) = h(a_1)h(a_2)....h(a_n)$

اگر L زبانی روی Σ باشد، آنگاه تصویر همریختی آن به صورت زیر تعریف می‌شود :

$$h(L) = \{h(w) : w \in L\}$$

مثال : فرض کنید $\Sigma = \{a,b\}$ و $\Gamma = \{a,b,c\}$و h به صورت زیر تعریف شده باشد :

$h(a) = ab$, $h(b) = bbc$

در اینصورت $h(aba) = abbbcab$. تصویر همریختی $L = \{aa,aba\}$، زبان $h(L) = \{abab, abbbcab\}$می‌باشد.

مثال : فرض کنید $\Sigma = \{a,b\}$ و $\Gamma = \{b,c,d\}$ و h به صورت زیر تعریف شده باشد: $h(a) = dbcc$ و $h(b) = bdc$ اگر L زبان منظمی باشد که با عبارت روبرو نشان داده شود $r = (a+b)^*(aa)^*$ آنگاه $r_1 = (dbcc + (bdc)^*)(dbccdbcc)^*$ نشان دهنده زبان منظم h(L) می‌باشد.

۵) فرض کنید h یک همریختی باشد. اگر L یک زبان منظم باشد، آنگاه تصویر همریختی آن یعنی h(L) نیز منظم است. پس خانواده زبانهای منظم تحت همریختی بسته است.

۶) تعریف : اگر L_1و L_2زبانهایی روی الفبای یکسان باشند در اینصورت خارج قسمت راست L_1 به L_2به صورت زیر تعریف می‌شود :

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

مثـــــــــال : $L_1=\{a^nb^m : n\geq 1, m\geq 0\}\cup\{ba\}$ و $L_2=\{a^m : m\geq 1\}$ آنگـــــــاه $L_1/L_2=\{a^nb^m : n\geq 1, m\geq 0\}$ رشته‌های موجود در L_2 شامل یک یا بیشتر b هستند. بنابراین، با حذف یک یا بیشتر b از رشته‌هایی در L_1 کـه بـه حـداقل یـک b بـه عنـوان پسوند ختم می‌شوند، به جواب می‌رسیم.

۷) قضیه : اگر L_1 و L_2 زبانهایی منظم باشند، آنگاه L_1/L_2 نیـز مـنظم اسـت. زبانهـای منظم تحت خارج قسمت راست نسبت به یک زبان منظم، منظم است.

مثال : $a^*b+a^*baa^*=a^*ba^*$

۸) خانواده زبانهای منظم تحت اجتماع و اشتراک متناهی بسته است :

$$L_\cup=\bigcup L_i \quad i=\{1,2,....,n\}$$

$$L_I=\bigcap L_i \quad i=\{1,2,...,n\}$$

۹) خانواده زبانهای منظم تحت تفاضل متقارن بسته است :

$S_1\ominus S_2=\{x : x\in S_1$ یـا $x\in S_2,$ ولی x متعلق به هر دو مجموعـه S_1 و S_2 نیســت $\}$

۱۰) خانواده زبانهای منظم تحت عملیات nor بسته است.

$$nor(L_1,L_2)=\{w : w\notin L_1 \quad \text{and} \quad w\notin L_2\}$$

۱۱) رابطه $L_1=L_1L_2/L_2$ برای همهٔ زبانهای L_1 و L_2 درست نیست.

۱۲) فرض کنید می‌دانیم که $L_1\cup L_2$ منظم است و این کـه L_1 متنـاهی اسـت، در ایـن صورت می‌توان نتیجه گرفت که L_2 منظم است.

مثال : اگر L یک زبان منظم باشد آنگاه $L_1=\{uv : u\in L\ ,\ |v|=2\}$ نیز منظم است.

مثال : اگر L یک زبان منظم باشد آنگاه $\{uv : u\in L, v\in L^R\}$ نیز منظم است.

۱۳) خانواده زبانهای منظم تحت خارج قسمت چپ نسبت بـه یـک زبـان مـنظم بسـته است. $L_2\setminus L_1=\{y : x\in L_2, xy\in L_1\}$

۱۴) اگر جملهٔ " اگر L_1 منظم باشد و $L_1 \cup L_2$ نیز منظم باشد، آنگاه L_2 باید منظم باشد " برای همهٔ زبانهای L_1 و L_2 درست باشد، آنگاه همهٔ زبانها باید منظم باشند.

۱۵) خانواده زبانهای منظم تحت اکثر عملیات بسته می‌باشد.

۱۶) قضیه : یک نمایش استاندارد از هر زبان منظم L روی Σ و هر $w \in \Sigma^*$ داده شده است، الگوریتمی برای تعیین این که آیا w در L هست یا خیر، وجود دارد.

۱۷) قضیه : برای تعیین این که آیا زبان منظم، که به صورت استاندارد نمایش داده شده است تهی، متناهی یا نامتناهی است، الگوریتمی وجود دارد.

۱۸) قضیه : نمایشهای استاندارد دو زبان منظم L_1 و L_2 داده شده است، الگوریتمی وجود دارد که تعیین می‌کند آیا $L_1=L_2$ می‌باشد یا خیر.

۱۹) برای هر w و هر زبان منظم L_1 و L_2 داده شده، الگوریتمی برای تعیین این که آیا $w \in L_1 - L_2$ هست یا خیر، وجود دارد.

۲۰) برای هر زبان منظم L_1 و L_2، الگوریتمی برای تعیین این که آیا $L_1 \subseteq L_2$ هست وجود دارد.

۲۱) برای هر زبان L الگوریتمی برای تعیین این که آیا $\lambda \in L$ هست وجود دارد.

۲۲) برای دو زبان منظم L_1 و L_2 ، الگوریتمی برای تعیین این که آیا $L_1 = L_1/L_2$ هست یا خیر، وجود دارد.

۲۳) یک زبان را زبان مقلوب گویند اگر $L = L^R$. الگوریتمی وجود دارد که تعیین می‌کند آیا زبان منظم داده شده، یک زبان مقلوب است یا خیر.

۲۴) در اکثر موارد در مورد زبانها و گرامرهای منظم الگوریتم‌هایی وجود دارد.

مثال : زبان $L = \{a^n b^n : n \geq 0\}$ نامنظم است.

لم تزریق، از اصل لانه کبوتر به شکل دیگری استفاده می‌کند. اثبات بر این پایه استوار است که در یک گراف انتقال با n رأس، هر راهی با طول n یا بیشتر باید یک رأس را تکرار کند. یعنی دارای چرخه باشد.

۲۵) فرض کنید L یک زبان منظم نامتناهی باشد. در اینصورت یک عدد صحیح مثبت m وجود دارد بطوریکه هر $w \in L$ با $|w| \ge m$ می‌تواند به صورت زیر تجزیه شود :

w=xyz، با $|xy| \le m$, $|y| \ge 1$

بطوریکه $w_i = xy^iz$ به ازای همهٔ i= 0,1,2... نیز در L باشد.

مثال : $L = \{ww^R : w \in \Sigma^*\}$ نامنظم است.

مثال : چند زبان نامنظم : $L = \{w \in \Sigma^* : n_a(w) < n_b(w)\}$ ،

$L = \{(ab)^n a^k : n > k, k \ge 0\}$, $L = \{a^{n!} : n \ge 0\}$

مثال : نشان دهید $L = \{a^n b^k c^{n+k} : n \ge 0, k \ge 0\}$ منظم نیست: از بستار تحت همریختی استفاده می‌کنیم.

فرض کنید $h(a) = a$, $h(b) = a$, $h(c) = c$، در این صورت $h(L) = \{a^{n+k} c^{n+k} : n + k \ge 0\}$ که برابر است با $\{a^i c^i : i \ge 0\}$، می‌دانیم که این زبان منظم نیست بنابراین L نیز نمی‌تواند منظم باشد.

مثال : زبان $L = \{a^n b^l : n \ne l\}$ منظم نیست.

مثال : زبان $L = \{w : n_a(w) = n_b(w)\}$ منظم نیست. $\leftarrow \bar{L}$ نیز منظم نیست.

مثال : زبانهای زیر منظم نیستند :

$L = \{a^n b^l a^k : k \ne n + l\}$ ، $\{L = a^n b^l a^k : k \ge n + l\}$

$L = \{w : n_a(w) \ne n_b(w)\}$ ، $L = \{a^n b^l : n \le l\}$ ، $L = \{a^n b^l a^k : n = l \quad \text{or} \quad l \ne k\}$

$L = \{wwww^R : w \in \{a,b\}^*\}$ ، $L = \{ww : w \in \{a,b\}^*\}$

$L = \{a^n b^k : n > k\} \cup \{a^n b^k : n \ne k - 1\}$ ، $L = \{a^n : n \ge 2,$ یک عدد اول است $n\}$

۲۶) (جمله غلط) : اگر L_1 و L_2 زبانهای نامنظم باشند، آنگاه $L_1 \cup L_2$ نیز نامنظم است.

مثال : $L_1 = \{a^n b^m : n \le m\}, L_2 = \{a^n b^m : n > m\}$ ، داریم $L_1 \cup L_2 = L(a^* b^*)$

مثال : $L = \{a^n b^l a^k : n + 1 + k > 5\}$ منظم است.

مثال : $L=\{a^n b^l a^k : n>5, l>3, k\le l\}$ نامنظم است.

۲۷) فرض کنید L_1 و L_2 زبانهایی منظم باشند. آیا زبان $L=\{w : w\in L_1$ و $w^R\in L_2\}$ لزوماً منظم است؟ بلی

مثال : $L=\{uww^R v : u,v,w\in\{a,b\}^+\}$ منظم است.

مثال : $L=\{uww^R v : u,v,w\in\{a,b\}^+, |u|\ge|v|\}$ نامنظم است.

۲۸) زبانهای منظم تحت اجتماع نامتناهی و اشتراک نامتناهی بسته نیست.

فصل۵: زبانهای مستقل از متن

قوانین یک گرامر مستقل از متن به دو روش محدود می‌شوند: سمت چپ باید یک متغیر تنها باشد ولی در سمت راست هر چیزی مجاز دانسته می‌شود.

۱) هر گرامر منظم مستقل از متن است، بنابراین یک زبان منظم، مستقل از متن نیز می‌باشد.

مثال : گرامر $S\to aSa,\ S\to bSb,\ S\to\lambda$ مستقل از متن است و زبان $L=\{ww^R : w\in\{a,b\}^*\}$ را تولید می‌کند. (منظم نیست) ، $\bar{L}$ نیز مستقل از متن است.

مثال : گرامر G با قوانین $S\to abB$ و $A\to aaBb$ و $B\to bbAa$ و $A\to\lambda$ مستقل از متن است. $L(G)=\{ab(bbaa)^n bba(ba)^n : n\ge 0\}$

۲) هر دو مثال بالا گرامرهایی هستند که نه تنها مستقل از متن، بلکه خطی نیز هستند. واضح است که گرامرهای منظم و خطی، مستقل از متن هستند ولی یک گرامر مستقل از متن لزوماً خطی نیست.

مثال : زبان $L=\{a^n b^m : n\ne m\}$ مستقل از متن است: حالت $n\succ m \leftarrow$ $S\to aS_1\ ,\ S_1\to aS_1b|\lambda\ ,\ A\to aA|a$ و پس از نوشتن گرامر برای حالت $n\prec m$ در نهایت به جواب نهایی می‌رسیم :

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

گرامر بدست آمده مستقل از متن است، بنابراین L یک زبان مستقل از متن است. با وجود این، گرامر خطی نیست.

مثال : گرامر $S \to aSb|SS|\lambda$ گرامری است که مستقل از متن است ولی خطی نیست، بعضی از رشته‌هایی که تولید می‌کند: ababab,aababb ، پس نتیجه می‌گیریم که

$L = \{w \in \{a,b\}^* : n_a(w), n_a(v) \geq n_b(v)$ و که v هر پیشوندی از w می‌باشد $\}$

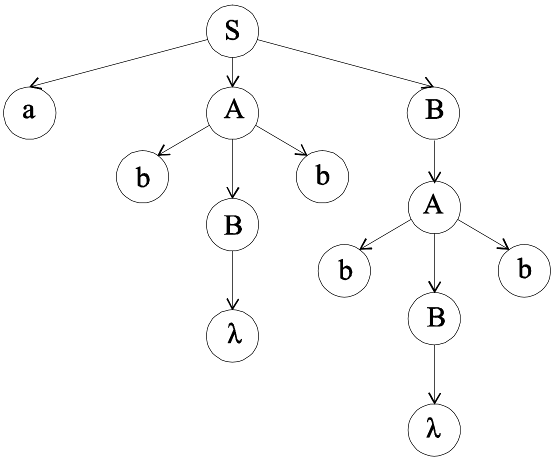
۳) قضیه : فرض کنید G یک گرامر مستقل از متن باشد، آنگاه برای هر $w \in L(G)$ یک درخت اشتقاق برای G وجود دارد که حاصل آن w است.

مثال : گرامر G را در نظر بگیرید:

$S \to AB\ ,\ A \to aaA\ ,\ A \to \lambda\ ,\ B \to Bb\ ,\ B \to \lambda$ این گرامر زبان $L(G) = \{a^{2n}b^m : n \geq 0, m \geq 0\}$ را تولید می‌کند. در این زبان رشته aab از هر دو اشتقاق چپ و راست بدست می‌آید.

مثال : گرامر $S \to aAB\ , A \to bBb\ ,\ B \to A|\lambda$ را در نظر بگیرید. در اینصورت رشته abbbb هم از اشتقاق چپ‌ترین و هم راست‌ترین بدست می‌آید.

مثال : درخت اشتقاق $S \to aAB\ ,\ A \to bBb\ ,\ B \to A|\lambda$



مثال : زبانهای زیر مستقل از متن هستند(همگی $n \ge 0, m \ge 0$)

الف) $L=\{a^n b^m : n \le m+3\}$ ب) $L=\{a^n b^m : n \ne m-1\}$

ج) $L=\{a^n b^m : n \ne 2m\}$ د) $L=\{a^n b^m : 2n \le m \le 3m\}$

ه) $L=\{w \in \{a,b\}^* : n_a(w) \ne n_b(w)\}$

و) $L=\{w \in \{a,b\}^* : n_a|v| \ge n_b(v)$ و v هر پیشوندی از w است $\}$

ز) $L=\{w \in \{a,b\}^* : n_a(w) = 2n_b(w)+1\}$

مثال : گرامر الف) : ابتدا مورد $n = m+3$ را حل می‌کنیم. سپس b های بیشتری اضافه می‌کنیم. موارد بدون a، یک a و دو a را فراموش نکنید:

$$S \to aaaA|aaA|aA|\lambda,\ A \to aAb|B,\ B \to Bb|\lambda$$

مثال : گرامر د) یک راه حل ساده : $S \to aSbb|aSbbb|\lambda$ که برای هر a ، bb یا bbb را بصورت غیرقطعی تولید می‌کند.

مثال : زبانهای زیر مستقل از متن هستند : $(n \ge 0, m \ge 0, k \ge 0)$

الف) $L=\{a^n b^m c^k : n=m \quad \text{or} \quad m \le k\}$

ب) $L=\{a^n b^m c^k : n=m \quad \text{or} \quad m \ne k\}$ (و نیز $\bar{L}$)

ج) $L=\{a^nb^mc^k : k=n+m\}$ د) $L=\{a^nb^mc^k : n+2m=k\}$

ه) $L=\{a^nb^mc^k : k=|n-m|\}$

و) $L=\{w\in(a,b,c)^* n_a(w)+n_b(w)\neq n_c(w)\}$

ز) $L=\{a^nb^mc^k : k\neq n+m\}$ ح) $L=\{a^nb^nc^k : k\geq 3\}$

ر) اجتماع قسمتهای (الف) و (د)

مثال : گرامر الف) : مورد اول $n=m$ و k دلخواه

$\leftarrow S_1\to AC, A\to aAb|\lambda, C\to Cc|\lambda$ و در مورد دوم n دلخواه بوده و $m\leq k$ می‌باشد. $\leftarrow S_2\to BD, B\to aB|\lambda, D\to bDc|E, E\to Ec|\lambda$ و نهایتاً $S\to S_1|S_2$

مثال : گرامر ه): مسئله را به دو قسمت تقسیم می‌کنیم : n=k+m و m=k+n که در اولین مورد داریم $S_1\to aS_1b|\lambda$, $S\to aSc|S_1|\lambda$ و الی آخر.

مثال : این زبان مستقل از متن است : $L=\{a^nww^Rb^n : w\in\Sigma^*, n\geq 1\}$

مثال : فرض کنید $L=\{a^nb^n : n\geq 0\}$ در این صورت L^R مستقل از متن است. همچنین L^k $(k\geq 0)$ و $\overline{L}$ و L^* مستقل از متن هستند.

مثال : زبان $L=\{uvwv^R : u,v,w\in\{a,b\}^+, |u|=|w|=2\}$ مستقل از متن است.

مثال : $L=\{w_1cw_2 : w_1,w_2\in\{a,b\}^+, w_1\neq w_2^R\}$ با $\Sigma=\{a,b,c\}$ مستقل از متن است.

مثال : یک گرامر مستقل از متن برای مجموعه همهٔ عبارات منظم روی الفبای $\{a,b\}$:

$E\to E+E|E.E|E^*|(E)|\lambda|\phi$

۴) اگر G یک گرامر مستقل از متن باشد، آنگاه هر $w\in L(G)$ دارای یک اشتقاق چپ‌ترین و یک اشتقاق راست‌ترین می‌باشد.

۵) فرض کنید $G=(V,T,S,P)$ یک گرامر مستقل از متن باشد بطوریکه هر یک از قوانین آن به شکل $A\to v$ با $|v|=k>1$ باشد. نشان دهید که درخت اشتقاق برای هر $w\in L(G)$ دارای ارتفاع h خواهد بود بطوریکه

$$\log_k^{|w|}\le h\le\frac{(|w|-1)}{k-1}$$

۶) الگوریتمی که می‌تواند پاسخگو باشد که w در $L(G)$ هست یا خیر الگوریتم عضویت می‌باشد. اصطلاح تجزیه به یافتن دنباله‌ای از قوانین که به وسیله آنها $w\in L(G)$ مشتق می‌شود اطلاق می‌گردد.

۷) قضیه : برای هر گرامر مستقل از متن الگوریتمی وجود دارد که هر $w\in L(G)$ را در تعداد مراحلی متناسب با $|w|^3$ تجزیه می‌کند.

۸) اگر G یک گرامر ساده باشد، آنگاه هر رشته w در $L(G)$ می‌تواند با تلاشی متناسب با $|w|$ تجزیه شود.

۹) تعریف : یک گرامر مستقل از متن G را مبهم گویند اگر یک $w\in L(G)$ وجود داشته باشد که حداقل دارای دو درخت اشتقاق متفاوت باشد. به عبارت دیگر، ابهام به وجود دو یا بیشتر اشتقاق چپ‌ترین یا راست‌ترین اشاره دارد.

۱۰) تعریف : اگر L یک زبان مستقل از متن باشد که برای آن گرامر غیرمبهم وجود داشته باشد، آنگاه L را غیرمبهم گویند. اگر هر گرامری که L را تولید می‌کند مبهم باشد، آنگاه زبان را ذاتاً مبهم گویند.

مثال : زبان $L=\{a^nb^nc^m\}\cup\{a^nb^mc^m\}$ که $n,m\ge 0$، یک زبان مستقل از متن ذاتاً مبهم است.

مثال : یک گرامر ساده برای $L=\{a^nb^n:n\ge 1\}$ بیابید: $S\to aA, A\to aAB, B\to b$

توجه کنید که گرامر واضحتر به صورت $S\to aS_1B \quad , \quad S_1\to aS_1B|\lambda\ ,B\to\lambda$ می‌باشد که این گرامر ساده نیست.

۱۱) یک زبان منظم نمی‌تواند ذاتاً مبهم باشد.

مثال : زبان $L=\left\{ww^R : w\in\{a,b\}^*\right\}$ ذاتاً مبهم نیست.

۱۲) ممکن است یک گرامر برای یک زبان مبهم باشد ولی زبانی که به آن دلالت می‌کند مبهم نباشد.

۱۳) فرض کنید $G=(V,T,S,P)$ یک گرامر مستقل از متن باشد به طوری که هر $A\in V$ در سمت چپ حداکثر یک قانون داشته باشد. در اینصورت G غیرمبهم است.

فصل ۶: ساده‌سازی گرامرهای مستقل از متن و شکلهای نرمال

الگوریتم عضویت CYK برای زبانها (گرامرهایی) که در فرم نرمال چامسکی باشند جواب می‌دهد و از مرتبه $O(n^3)$ می‌باشد. به فصل ۵ کتاب مراجعه کنید.

فصل ۷: ماشینهای پشته‌ای(pda)

۱) تعریف : یک پذیرنده پشته‌ای غیرقطعی (npda) با ۷ تایی روبرو تعریف می‌شود :

$M=(Q,\Sigma,\Gamma,\delta,q_\circ,z,F)$

$z\in\Gamma$ نماد شروع پشته است، $F\subseteq Q$ مجموعه حالات نهایی است و ....

مثال : عملکرد این پذیرنده پشته‌ای غیرقطعی چیست؟

$Q=\{q_\circ,q_1,q_2,q_3\}$، $\Sigma=\{a,b\}$، $\Gamma=\{0,1\}$, $z=0$, $F=\{q_3\}$

$\delta(q_\circ,a,0)=\{(q_1,10),(q_3,\lambda\}$, $\delta\{q_\circ,\lambda,0\}=\{(q_3,\lambda)\}$,

$\delta(q_1,a,1)=\{(q_1,11)\}$, $\delta(q_1,b,1)=\{(q_2,\lambda)\},\delta(q_2,b,1)=\{(q_2,\lambda)\}$

و $\delta(q_2,\lambda,0)=\{(q_3,\lambda)\}$

توجه کنید که نحوهٔ قرارگیری 10 در پشته به صورت $\begin{array}{|c|}1\\\hline 0\end{array}$ می‌باشد. جواب :

$L=\left\{a^nb^n : n\geq 0\right\}\cup\{a\}$

مثالهای دیگر در کتاب عبارتند از :

$L=\left\{w\in\{a,b\}^* : n_a(w)=n_b(w)\right\}$, $L=\left\{ww^R : w\in\{a,b\}^+\right\}$

مثال : زبانهای زیر مستقل از متن هستند : الف: $L=\{a^n b^{2n} : n \geq 0\}$

ب) $L=\{wcw^R : w \in \{a,b\}^*\}$ ج) $L=\{a^n b^m c^{n+m} : n \geq 0, m \geq 0\}$

د) $L=\{a^n b^{n+m} c^m : n \geq 0, m \geq 1\}$ ه) $L=\{a^3 b^n c^n : n \geq 0\}$

و) $L=\{a^n b^m : n \leq m \leq 3n\}$ ز) $L=\{w : n_a(w) = n_b(w)+1\}$

ح) $L=\{w : n_a(w) = 2n_b(w)\}$

ط) $L=\{w : n_a(w)+n_b(w) = n_c(w)\}$

ی) $L=\{w : 2n_a(w) \leq n_b(w) \leq 3n_a(w)\}$

ک) $L=\{w : n_a(w) < n_b(w)\}$

مثال : حل الف) :

$$\delta(q_\circ,\lambda,z)=\{(q_f,z)\},\delta(q_\circ,a,z)=\{(q_1,11z)\}$$
$$\delta(q_\circ,a,1)=\{(q_1,111)\},\delta(q_1,b,1)=\{(q_1,\lambda)\},\delta(q_1,\lambda,z)=\{(q_f,z)\}$$

مثال : زبان مستقل از متن : $L=\{a^n b^m : n \geq 0, n \neq m\}$ و

$$L=\{w_1 c w_2 : w_1,w_2 \in \{a,b\}^*, w_1 \neq w_2^R\} \quad \left(\sum=\{a,b,c\}\right)$$

مثال : اگر ماشین پشته‌ای در هر مرحله z را هم pop و هم push کند در نتیجه اصلاً از پشته استفاده نمی‌کند و پذیرنده متناهی است، در این حالت می‌توان از یک ماشین پشته‌ای قطعی استفاده کرد :

$$\delta(q_\circ,a,z)=\{(q_1,z)\}\xrightarrow{\text{تبدیل به}}\delta(q_\circ,a)=q_1$$

مثال : یک پذیرنده پشته‌ای غیرقطعی با حداکثر ۲ حالت داخلی بیاید که زبان $L(aa^*ba^*)$ را بپذیرد :

$$\delta(q_\circ,a,1)=\{(q_\circ,1)\},\delta(q_\circ,b,1)=\{(q_\circ,z)\},\delta(q_\circ,a,z)=\{(q_\circ,z)\},\delta(q_\circ,\lambda,z)=\{(q_f,z)\}$$

۲) قضیه : برای هر زبان مستقل از متن L ، یک پذیرنده پشته‌ای غیرقطعی M وجود دارد بطوریکه L=L(M) .

۳) قضیه : اگر در یک پذیرنده پشته‌ای غیر قطعی M داشته باشیم L=L(M) آنگاه L یک زبان مستقل از متن است.

زبانهای مستقل از متن : $L=\{a^{n+1}b^{2n}:n\geq 0\}$ و $L=\{a^nb^{n+1}:n\geq 0\}$

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

۴) برای هرپذیرنده پشته‌ای غیر قطعی یک پذیرنده پشته‌ای غیرقطعی معادل وجود دارد که شرایط قضیه (۳-قضیه) را بر آورده می‌سازد.

۵) یک پذیرنده پشته‌ای غیرقطعی بر (dpda) یک ماشین پشته‌ای است که هرگز حرکتی در انتخابش ندارد.

۶) تعریف : زبان L را یک زبان مستقل از متن قطعی می‌گویند اگر و فقط اگر یک پذیرنده پشته‌ای قطعی M وجود داشته باشد بطوریکه L=L(M) باشد.

مثال : زبان $L=\{a^nb^n:n\geq 0\}$ مستقل از متن قطعی است.

مثال : چند زبان مستقل از متن قطعی :

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

$L=\{wcw^R:w\in\{a,b\}^*\}$, $L=\{a^nb^n:n\geq 1\}\cup\{a\}$

مثال : این زبان غیرقطعی است :

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

مثال : زبان $L=\{w\in\{a,b\}^*:n_a(w)\neq n_b(w)\}$ یک زبان مستقل از متن قطعی است.

۷) هر زبان منظم یک زبان مستقل از متن قطعی است.

۸) اگر L_1 یک زبان مستقل از متن قطعی و L_2 یک زبان منظم باشد، آنگاه زبان $L_1\cup L_2$ یک زبان مستقل از متن قطعی است. ($L_1\cap L_2$ نیز چنین است)

۹) معکوس یک زبان مستقل از متن قطعی ممکن است قطعی نباشد.

۱۰) یک زبان مستقل از متن قطعی هرگز ذاتاً مبهم نیست.

فصل ۸: خواص زبانهای مستقل از متن

۱) قضیه : فرض کنید L یک زبان مستقل از متن متناهی باشد. در این صورت یک عـدد مثبت m وجود دارد بطوریکه هر $w \in L$ با $|w| \ge m$ می‌تواند به صورت زیر تجزیه شود: $w = uvxyz$ با $|vxy| \le m$ و $|vy| \ge 1$ بطوریکه $uv^ixy^iz \in L$ بـرای همـهٔ $i=0,1,2,\ldots$.
این قضیه به عنوان لم تزریق برای زبانهای مستقل از متن شناخته می‌شود.

مثال : این زبانها مستقل از متن نیستند : $L = \{a^nb^nc^n : n \ge 0\}$

$$L = \{ww : w \in \{a,b\}^*\} \quad , \quad L = \{a^nb^n\} \cup \{a^nb^{2n}\} \cup \{a^nb^nc^n\}$$

$$L = \{a^nb^j : n = j^2\} \quad , \quad L = \{a^{n!} : n \ge 0\}$$

۲) تعریف : یک زبان مستقل از متن L را خطی گویند اگر یـک گرامـر مسـتقل از مـتن خطی G وجود داشته باشد بطوریکه $L=L(G)$.

۳) یک لم تزریق برای تعیین خطی بودن یا نبودن زبان نیز وجود دارد. (خطی نبودن)

۴) خانواده زبانهای خطی یک زیرمجموعه محـض از خـانواده زبانهـای مسـتقل از مـتن است. مستقل از متن $\subset$ خطی

مثال : چند زبان که مستقل از متن نیستند : $L = \{ww^Rw : w \in \{a,b\}^*\}$

$L = \{a^n :$ یک عدد اول است $n\}$ و $L = \{w \in \{a,b,c\}^* : n_a^2(w) + n_b^2(w) = n_c^2(w)\}$

$L = \{a^{n^2} : n \ge 0\}$ و همچنین $\overline{L}$

مثال : چند زبان غیر مستقل از متن روی $\Sigma = \{a,b,c\}$:

الف) $L = \{a^nb^j : n \le j^2\}$ ب) $L = \{a^nb^j : n \ge (j-1)^3\}$

ج) $L = \{a^nb^jc^k : k > n, k > j\}$ ه) $L = \{a^nb^jc^k : k > n, k > j\}$

و) $L = \{w : n_a(w) < n_b(w) < n_c(w)\}$ ز) $L = \{w : n_a(w)/n_b(w) = n_c(w)\}$

ح) $L=\left\{w\in\{a,b,c\}^*: n_a(w)+n_b(w)=2n_c(w)\right\}$

خ) $L=\left\{a^n b^j a^n b^j : n\geq 0, j\geq 0\right\}$

مثال : این زبان $L=\left\{w_1cw_2 : w_1w_2\in\{a,b\}^*, w_1\neq w_2\right\}$ مستقل از متن است.

مثال : $L=\left\{a^n b^n a^m b^m : n\geq 0, m\geq 0\right\}$ مستقل از متن است ولی خطی نیست.

مثال : $L=\left\{w\in\{a,b,c\}^*: n_a(w)+n_b(w)=n_c(w)\right\}$ مستقل از متن است ولی خطی نیست.

مثال : زبان $L=\left\{w: n_a(w)\geq n_b(w)\right\}$ خطی نیست.

مثال : $L=\left\{a^{nm}:\right.$ n و m اعداد اول هستند $\left.\right\}$ مستقل از متن نیست.

۵) قضیه : خانواده زبانهای مستقل از متن تحت اجتماع، اتصال و بستار ستاره‌ای بسته است.

۶) قضیه : خانواده زبانهای مستقل از متن تحت اشتراک و متمم‌گیری بسته نیست.

۷) قضیه : اگر L_1 یک زبان مستقل از متن و L_2 یک زبان منظم باشد در اینصورت $L_1\cap L_2$ مستقل از متن است. (خانواده زبانهای مستقل از متن تحت اشتراک منظم بسته است.)

مثال : نشان دهید زبان $L=\left\{a^n b^n : n\geq 0, n\neq 100\right\}$ مستقل از متن است: داریم $L_1=\left\{a^{100}b^{100}\right\}$ که منظم است (چون متناهی است) $\leftarrow$ $L=\left\{a^n b^n : n\geq 0\right\}\cap\bar{L}_1$ $\leftarrow$ مستقل از متن است.

مثال : زبان $L=\left\{w\in\{a,b,c\}^*: n_a(w)=n_b(w)=n_c(w)\right\}$ مستقل از متن نیست.

۸) قضیه : با داشتن یک گرامر مستقل از متن G، الگوریتمی برای تصمیم‌گیری در مورد این که آیا L(G) تهی است یا خیر وجود دارد.

۹) قضیه : الگوریتمی برای تعیین این که آیا L(G) متناهی است یا خیر وجود دارد. (با داشتن گرامر مستقل از متن G)

۱۰) الگوریتمی وجود ندارد که مشخص کند دو گرامر مستقل از متن یک زبـان را تولیـد می‌کنند.

مثال : زبان $L=\{w\in\{a,b,c\}^*:n_a(w)\neq n_b(w)\neq n_c(w)\}$

مثال : زبان $L=\{a^nb^nc^m:n\geq 0,m\geq 0\}$ مستقل از متن قطعی است.

۱۱) خانواده زبانهای مستقل از متن و خانواده زبانهای خطی تحت همریختی بسته است.

۱۲) خانواده زبانهای مستقل از متن تحت معکوس کردن بسته است.

۱۳) خانواده زبانهای مستقل از متن تحت تفاضل بسته نیست ولی تحت تفاضـل مـنظم بسته است، یعنی اگر L_1 مستقل از متن و L_2 منظم باشـد، آنگـاه $L_1 - L_2$ مسـتقل از متن است.

۱۴) خانواده زبانهای مستقل از متن قطعی تحت تفاضل منظم بسته است.

۱۵) خانواده زبانهای خطی تحت اجتماع بسته است، ولی تحت اتصال بسته نیست.

۱۶) خانواده زبانهای خطی تحت اشتراک بسته نیست.

۱۷) خانواده زبانهای مستقل از متن قطعی تحت اجتماع و اشتراک بسته نیست.

۱۸) اگر L_1 خطی و L_2 منظم باشد، آنگاه L_1L_2 یک زبان خطی است.

۱۹) خانواده زبانهای مستقل از متن غیرمبهم تحت اجتماع بسته نیست.

۲۰) خانواده زبانهای مستقل از متن غیرمبهم تحت اشتراک بسته نیست.

مثال : زبانهای زیر مستقل از متن هستند: $L=\{a^nb^n:n\geq 0$ و n مضربی از 5 نیست$\}$، $L=\{w\in\{a,b\}^*:n_a(w)=n_b(w)$ و w شامل زیر رشته aab نیست$\}$

۲۱) الگوریتمی وجود دارد که برای هر گرامر مستقل از متن داده شده G، بتواند تعیـین کند آیا $\lambda\in L(G)$ هست یا خیر.

۲۲) الگوریتمی وجود دارد که تعیین می‌کند آیا زبـان تولیـد شـده توسـط یـک گرامـر مستقل از متن شامل کلماتی با طول کمتر از عدد داده شده n هست.

۲۳) فرض کنید L_1 یک زبان مستقل از متن و L_2 منظم باشد، الگوریتمی وجود دارد که تعیین می‌کند آیا L_1 و L_2 عضو مشترکی دارند یا خیر.

فصل ۹. ماشینهای تورینگ

۱) چندین تعریف مختلف از یک ماشین تورینگ وجود دارد. ویژگی‌های اصلی یک ماشین تورینگ استاندارد را خلاصه می‌کنیم:

۱ . ماشین تورینگ دارای یک نوار است که از دو طرف نامحدود است و هر تعداد حرکت به چپ و راست امکان‌پذیر است.

۲ . ماشین تورینگ قطعی است بدین معنی که δ برای هر پیکربندی حداکثر یک حرکت تعریف می‌کند.

۳. هیچ فایل ورودی خاصی وجود ندارد. ما فرض می‌کنیم که نوار در ابتدا دارای محتوای مشخصی است و بخشی از این را می‌توان به عنوان ورودی در نظر گرفت. به طور مشابه هیچ وسیله خروجی خاصی وجود ندارد. هر گاه ماشین توقف کند برخی یا همه محتویات نوار را می‌توان به عنوان خروجی در نظر گرفت.

۲) تعریف : فرض کنید $M=(Q,\Sigma,\Gamma,\delta,q_\circ,\square,f)$ یک ماشین تورینگ باشد. در این صورت زبان پذیرفته شده بوسیله M عبارت است از :

$$L(M)=\left\{w\in\Sigma^+ : q_\circ w \xrightarrow{*} x_1 q_f x_2\right\} \quad , \quad q_f\in F, x_1, x_2\in\Gamma^*$$ برای بعضی }

مثال : برای $\Sigma=\{0,1\}$، یک ماشین تورینگ طراحی کنید که زبان نمایش داده شده توسط عبارت منظم 00^* را بپذیرد.

$$\delta(q_\circ,0)=(q_\circ,0,R) \quad , \quad \delta(q_\circ,\square)=(q_1,\square,R)$$

توجه کنید که اگر ماشین تورینگ در حالت $q_\circ$ شروع کند در حالیکه روی یک نماد خالی باشد، در این صورت در یک حالت نهایی متوقف می‌شود. این را به عنوان پذیرش λ تفسیر می‌کنیم.

۳) برخی از اعمال قابل محاسبه توسط ماشین تورینگ (ماشینهای تورینگ به عنوان تراگذرها) :

۱. با داشتن دو عدد صحیح مثبت x وy ، x+y را محاسبه می‌کند.

۲. رشته‌هایی از ۱ها را کپی کند. به طور دقیق‌تر محاسبه زیر را انجام دهد : برای هر $\left\{q_{\circ}w \xrightarrow{*} q_f ww, w \in \{1\}^+\right\}$

۳. با داشتن دو عدد صحیح مثبت x وy، اگر $x \geq y$ باشد در یک حالت نهایی متوقف شده و اگر x<y باشد در یک حالت غیر نهایی متوقف می‌شود.

۴) از ترکیب ماشینهای تورینگ می‌توان برای انجام وظایف پیچیده استفاده کرد:

۱) ماشین تورینگی که محاسبه زیر را انجام دهد :

$$\begin{cases} f(x,y) = x + y, & x \geq y \\ f(x+y) = 0 \quad , & x < y \end{cases}$$

۲) $\text{if } a \text{ then } q_j \text{ else } q_k$

تز تورینگ :

۱. هر چیزی که بتواند بر روی هر کامپیوتر رقمی موجود انجام شود، با یک ماشین تورینگ نیز قابل انجام است.

۲. هیچکس تا کنون قادر نبوده است که مسئله‌ای پیشنهاد نماید که آنچه ما به اصطلاح یک الگوریتم در نظر می‌گیریم قابل حل باشد ولی نتوان یک برنامه ماشین تورینگ برای آن نوشت.

۳. مدل های دیگری برای محاسبه مکانیکی پیشنهاد شده‌اند، ولی هیچ یک از آن‌ها قوی‌تر از مدل ماشین تورینگ نیستند.

۵) تز تورینگ بر خلاف اعتبارش، هنوز یک فرضیه است.

۶) ماشین‌های تورینگ قوی‌تر از ماشین های پشته ای به نظر می‌رسند. از آنجایی که نوار یک ماشین تورینگ همواره می‌تواند شبیه یک پشته رفتار کند به نظر می‌رسد واقعا نمی‌توانیم ادعا کنیم که یک ماشین تورینگ قوی‌تر است. اما از این حقیقت که یک

ماشین تورینگ هم چنان که تعریف شد قطعی است در حالیکه یک ماشین پشته‌ای می‌تواند غیر قطعی باشد چشم‌پوشی کرده ایم . بنابراین هنوز نمی‌توانیم ادعا کنیم که که ماشین های تورینگ قوی تر از یک ماشین پشته ای هستند.

فصل۱۰. مدل‌های دیگر ماشین‌های تورینگ

۱) قضیه : رده ماشین‌های تورینگ با انتخاب توقف معادل با رده ماشین‌های تورینگ استاندارد می‌باشد.

۲) رده ماشین‌های تورینگ با نوار نیمه محدود و ماشین‌های تورینگ برون خط با ماشین‌های تورینگ استاندارد یکسان می‌باشد.

۳) ماشین تورینگی را در نظر بگیرید که در هر حرکت بخصوص می‌تواند یا نماد نوار را تغییر دهد و یا نوک خواندن نوشتن را حرکت دهد ولی نه هر دوی آن‌ها را. رده چنین ماشین تورینگی با رده ماشین‌های تورینگ معادل یکسان است.

۴) رده ماشینهای تورینگ چند نواره و چند بعدی معادل با ماشینهای تورینگ استاندارد.

۵) رده ماشین‌های تورینگ یک ماشین تورینگ استاندارد معادل با حداکثر6 حالت وجود دارد.

۶) هر محاسبه‌ای که بتواند با یک ماشین تورینگ استاندارد انجام شود می‌تواند با یک ماشین چند نواره دارای یک انتخاب توقف و حداکثر دو حالت انجام شود.

۷) در ماشین تورینگ غیر قطعی برد δ مجموعه‌ای از انتقال‌های ممکن است که هر یک می‌تواند توسط ماشین انتخاب شود.

۸) از آن جایی که واضح نیست غیر قطعیت چه نقشی در توابع محاسباتی ایفا می‌کند، ماشین‌های غیر قطعی معمولا به عنوان پذیرنده نگریسته می‌شوند.

۹) قضیه : رده ماشین‌های تورینگ قطعی و رده ماشین‌های تورینگ غیر قطعی برابر است.

۱۰) ماشین‌های تورینگ نمی‌توانند معادل با کامپیوترهای رقمی همه منظوره در نظر گرفته شوند، اما با طراحی یک ماشین تورینگ قابل برنامه ریزی مجدد که ماشین تورینگ عمومی نامیده می‌شود، می‌توان این ایراد را بر طرف کرد. ماشین تورینگ عمومی می‌تواند محاسبات را شبیه سازی کند.

۱۱) مجموعه‌ها متناهی و یا نامتناهی هستند. برای مجموعه های نامتناهی، بین مجموعه های قابل شمارش و مجموعه های غیر قابل شمارش تمایز قائل می‌شویم.

۱۲) قضیه : مجموعه همه ماشین‌های تورینگ، هرچند متناهی، قابل شمارش هستند.

۱۳) مجموعه همه سه‌تایی‌های (i , j, k) که i , j, k اعداد صحیح مثبت هستند، قابل شمارش است.

۱۴) فرض کنید که S_1 و S_2 مجموعه های قابل شمارش هستند. در این صورت $S_1 \cup S_2$ و $S_1 \times S_2$ نیز قابل شمارش هستند.

۱۵) حاصلضرب کارتزین تعداد متناهی از مجموعه های قابل شمارش، قابل شمارش است.

۱۶) یک ماشین کراندار خطی (LBA) یک ماشین تورینگ غیر قطعی می‌باشد که دارای یک نوار نامحدود است، ولی این که چه مقدار از نوار را می‌توان به کاربرد تابعی از ورودی است.

۱۷) برای زبان های زیر می‌توان یک ماشین کراندار خطی طراحی کرد : (در نتیجه زبان‌هایی وابسته به متن هستند.)

a) $L = \{a^n b^n c^n : n \geq 1\}$ b) $L = \{a^{n!} : n \geq 0\}$

c) $L = \{a^n : n = m^2, m \geq 1\}$ d) $L = \{a^n :$ یک عدد اول است $n\}$

e) $L = \{a^n :$ یک عدد اول نیست $n\}$ f) $L = \{ww : w \in \{a,b\}^+\}$

g) $L = \{w^n : w \in \{a,b\}^+, n \geq 1\}$ h) $L = \{www^R : w \in \{a,b\}^+\}$

۱۸) برای هر زبان مستقل از متن یک ماشین پشته‌ای پذیرنده وجود دارد بطوریکه تعداد نمادهای پشته هرگز از طول رشته ورودی بیش از یکی تجاوز نمی‌کند.

۱۹) با توجه به نظریه فوق می‌توان نشان داد که هر زبان مستقل از متن که شامل λ نباشد، توسط یک ماشین کراندار خطی پذیرفته می‌شود.

فصل۱۱. سلسله مراتبی از زبان‌های صوری و ماشین‌ها

تعریف : یک زبان L را شمارش‌پذیر بازگشتی گوییم اگر ماشین تورینگی وجود داشته باشد که آن را بپذیرد.

تعریف : یک زبان L بر روی Σ را بازگشتی گوییم اگر یک ماشین تورینگ M وجود داشته باشد که L را بپذیرد و روی هر w در Σ^+ توقف کند. به عبارت دیگر یک زبان بازگشتی است اگر و فقط اگر یک الگوریتم عضویت برای آن وجود داشته باشد.

قضیه : فرض کنید S یک مجموعه شمارای نامتناهی باشد، آنگاه مجموعه توانی آن یعنی 2^S شمارا نیست.

قضیه : برای هر ۴ غیرتهی، زبانهایی وجود دارند که شمارش‌پذیر بازگشتی نیستند.

قضیه : یک زبان شمارش‌پذیر بازگشتی وجود دارد که مکمل آن شمارش‌پذیر بازگشتی نیست.

قضیه : اگر یک زبانL و مکمل آن $\bar{L}$ هر دو شمارش‌پذیر بازگشتی باشند، آنگاه هر دو زبان بازگشتی هستند. اگر L بازگشتی باشد، آنگاه $\bar{L}$ نیز بازگشتی است و در نتیجه هر دو شمارش پذیر بازگشتی هستند.

قضیه : یک زبان شمارش‌پذیر بازگشتی وجود دارد که بازگشتی نیست، یعنی خانواده زبان‌های بازگشتی زیر مجموعه محضی از خانواده زبان‌های شمارش‌پذیر بازگشتی هستند.

مجموعه همه اعداد حقیقی شمارا نیستند.

مجموعه همه زبان هایی که شمارش پذیر بازگشتی نیستند، شمارا نمی‌باشند.

فرض کنید L یک زبان متناهی باشد، آنگاه L^+ شمارش‌پذیر بازگشتی است.

اگر یک زبان شمارش پذیر بازگشتی نباشد، مکمل آن نمی‌تواند بازگشتی باشد.

خانواده زبان‌های شمارش پذیر بازگشتی تحت اجتماع و اشتراک بسته است.

خانواده زبان های شمارش پذیر بازگشتی و بازگشتی تحت عمل معکوس کردن بسته هستند.

مکمل یک زبان مستقل از متن باید بازگشتی باشد.

فرض کنید L_1 بازگشتی و L_2 شمارش پذیر بازگشتی باشد. نشان دهید که $L_2 - L_1$ لزوما شمارش پذیر بازگشتی است.

فرض کنید L به گونه ای باشد که یک ماشین تورینگ برای شمارش عناصر L به ترتیب مناسب وجود داشته باشد این امر به معنای بازگشتی بودن L می‌باشد.

اگر L بازگشتی باشد L^+ نیز لزوما بازگشتی است.

فرض کنید S_1 یک مجموعه شمارا باشد، S_2 یک مجموعه شمارا نباشد و $S_1 \subset S_2$ باشد. آنگاه S_2 باید شامل تعداد نامتناهی عناصری باشد که در S_1 نیستند. و در حقیقت $S_2 - S_1$ نمی‌تواند شمارا باشد.

فرض کنید S یک مجموعه شمارای متناهی باشد. آنگاه این استدلال که مجموعه توانی آن یعنی 2^S شمارا نیست با شکست روبرو می‌شود.

تعریف : یک گرامر $G=(V,T,S,P)$ بدون محدودیت نامیده می‌شود اگر همه قوانین آن به شکل $u \to v$ باشند که u در $(V \cup T)^+$ و v در $(V \cup T)^*$ است. در این گرامر فقط یک محدودیت وجود دارد : λ مجاز نیست که در سمت چپ یک قانون باشد.

قضیه : هر زبان تولید شده بوسیله یک گرامر بدون محدودیت، شمارش‌پذیر بازگشتی است.

قضیه : برای هر زبان شمارش‌پذیر بازگشتی L، یک گرامر بدون محدودیت G وجود دارد بطوریکه $L = L(G)$ باشد.

زبان $L = \{a^{n+1}b^{n+k}, n \geq 1\ ,\ k = -1,1,3,...\}$ توسط گرامر بدون محدودیت

$S \to S_1B\ ,\ S_1 \to aS_1b\ , bB \to bbbB\ , aS_1b \to aa\ , B \to \lambda$ مشتق می‌شود.

تعریف : یک گرامر G=(V,T,S,P) حساس به متن نامیده می‌شود اگر همهٔ قوانین آن به شکل $x \to y$ باشند که $x, y \in (V \cup T)^+$, $|x| \le |y|$

$xAy \to xvy$ معادل با آن است که بگوییم $A \to v$

تعریف : یک زبان L را حساس به متن گویند اگر یک گرامر حساس به متن G وجود داشته باشد به طوریکه L=L(G) یا $L = L(G) \cup \{\lambda\}$ باشد.

قضیه : برای هر زبان حساس به متن L که شامل λ نباشد، ماشین کراندار خطی M وجود دارد بطوریکه L=L(M) می‌باشد.

قضیه : اگر یک زبان L بوسیله ماشین کراندار خطی M پذیرفته شود، آنگاه یک گرامر حساس به متن وجود دارد که L را تولید نماید.

قضیه : هر زبان حساس به متن L بازگشتی است.

قضیه : یک زبان بازگشتی وجود دارد که حساس به متن نیست.

ماشین‌های کراندار خطی دارای قدرت کمتری از ماشین‌های تورینگ هستند زیرا آنها فقط زیر مجموعه محضی از زبان‌های بازگشتی را می‌پذیرند. ماشین‌های کراندار خطی قویتر از ماشین های پشته ای هستند. زبان‌های مستقل از متن زیر مجموعه‌ای از زبان-های حساس به متن هستند. هر زبان پذیرفته شده توسط یک ماشین پشته‌ای بوسیله یک ماشین کراندار خطی نیز پذیرفته می‌شود، ولی زبان‌هایی وجود دارند که بوسیله ماشین‌های کراندار خطی پذیرفته می‌شوند اما برای آن‌ها ماشین پشته‌ای وجود ندارد.

چند زبان حساس به متن را مشاهده می‌کنید :

الف) $L = \{a^{n+1}b^n c^{n-1} : n \ge 1\}$ ب) $L = \{a^n b^n a^{2n} : n \ge 1\}$

ج) $L = \{a^n b^m c^n d^m : n \ge 1, m \ge 1\}$ د) $L = \{ww : w \in \{a,b\}^+\}$

ذ) $L = \{w : n_a(w) = n_b(w) = n_c(w)\}$ ر) $L = \{w : n_a(w) = n_b(w) < n_c(w)\}$

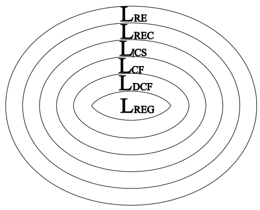
خانواده زبان‌های حساس به متن تحت عملیات اجتماع و معکوس کردن بسته اند.

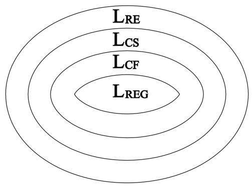
یک زبان حساس به متن : $L = \{wuw : w, u \in \{a,b\}^+\}$

سلسله مراتب چامسکی : زبان های زیر را در نظر بگیرید :

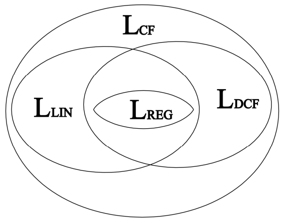
(L_{RE}) زبان‌های شمارش پذیر بازگشتی، (L_{CS})زبان‌های حساس به متن، (L_{CF})زبان های مستقل از متن، (L_{REG})زبان‌های منظم،(L_{DCF})زبان‌های مستقل از متن قطعی، (L_{REC})زبان های بازگشتی.

در نتیجه وابستگی‌های زیر این زبان‌ها وجود دارد :





مثال : ما قبلا زبان مستقل از متن $L=\{w: n_a(w)=n_b(w)\}$ را معرفی کردیم و نشان دادیم که قطعی است ولی خطی نیست. از طرف دیگر زبان $L=\{a^n b^n\} \cup \{a^n b^{2n}\}$ خطی می‌باشد اما قطعی نیست. این نشان می‌دهد که ارتباطی بین زبان‌های منظم، خطی، مستقل از متن قطی و مستقل از متن غیر قطعی وجود دارد که در زیر مشاهده می‌کنید :



فصل ۱۲. محدودیت‌های محاسبات الگوریتمی

۱) قضیه : مسئله توقف ماشین تورینگ تصمیم‌ناپذیر است.

۲) قضیه : اگر مسئله توقف تصمیم‌پذیر بود آنگاه هر زبان شمارش‌پذیر بازگشتی، بازگشتی می بود. در نتیجه مسئله توقف تصمیم ناپذیر است.

۳) گوییم مسئله A به مسئله B کاهش می‌یابد اگر تصمیم پذیری A از تصمیم پذیری B نتیجه شود. آنگاه اگر بدانیم که A تصمیم ناپذیر است، می‌توانیم نتیجه بگیریم که B نیز تصمیم ناپذیر است.

۴) هیچ الگوریتمی وجود ندارد که تعیین کند آیا یک ماشین تورینگ دلخواه روی همه ورودی‌ها متوقف می‌شود یا خیر.

۵) هیچ الگوریتمی برای تعیین این که آیا دو ماشین تورینگ M_1 و M_2 زبان یکسانی را می‌پذیرند وجود ندارد.

۶) هر مسئله ای که دامنه اش متناهی باشد تصمیم پذیر است.

۷) قضیه : فرض کنید G یک گرامر بدون محدودیت باشد. در این صورت مسئله تعیین این که $L(G)=\varnothing$ هست یا خیر تصمیم ناپذیر است.

۸) قضیه : فرض کنید M یک ماشین تورینگ باشد. در این صورت این مسئله که L(M) متناهی است یا خیر تصمیم ناپذیر است.

۹) برای یک ماشین تورینگ دلخواه M با $\Sigma=\{a,b\}$ ، مسئله" L(M) شامل دو رشته مختلف با طول یکسان است" تصمیم ناپذیر است.

۱۰) قضیه رایس بیان می‌کند که هر خاصیت غیر جزئی از یک زبان شمارش پذیر بازگشتی تصمیم ناپذیر است.

۱۱) این دو مسئله تصمیم ناپذیرند : L(M) شامل رشته‌ای به طول ۵ است و L(M) منظم است.

۱۲) فرض کنید M_1 و M_2 ماشین های تورینگ دلخواهی باشند، در این صورت مسئله $L(M_1)\subseteq L(M_2)$ تصمیم ناپذیر است.

۱۳) فرض کنید G_1 یک گرامر بدون محدودیت و G_2 یک گرامر منظم باشد. مسئله $L(G_1)\cap L(G_2)=\varnothing$ تصمیم ناپذیر است.

۱۴) قضیه : مسئله پس تناظر تغییر یافته تصمیم ناپذیر است.

۱۵) قضیه : مسئله پس تناظر تصمیم ناپذیر است.

۱۶) قضیه : هیچ الگوریتمی برای تعیین این که آیا هر گرامر مستقل از متن داده شده مبهم است وجود ندارد.

۱۷) قضیه : هیچ الگوریتمی در مورد تعیین این که آیا $L(G_1)\cap L(G_2)=\varnothing$ هست یا خیر برای گرامرهای دلخواه مستقل از متن G_1و G_2 وجود ندارد.

فصل ۱۳. مدل‌های دیگر محاسبات

۱) تعریف : یک تابع بازگشتی اولیه نامیده می‌شود اگر و فقط اگر بتواند از روی توابع اساسی P_K و s و z بوسیله ترکیب و بازگشت پذیری اولیه متوالی ساخته شود.

۲) قضیه : فرض کنید F مجموعه همه توابع از I به I باشد. آنگاه تابعی در F وجود دارد که بازگشتی اولیه نیست.

۳) قضیه : فرض کنید C مجموعه همه توابع قابل محاسبه عمومی از I به I باشد. در این صورت تابعی در C وجود داردکه بازگشتی اولیه نیست.

۴) قضیه : تابع آکرمن بازگشتی اولیه نیست.

۵) تعریف : یک تابع، بازگشتی μ گفته می‌شود اگر بتواند از روی توابع اساسی بوسیله دنباله‌ای از کاربردهای عملگر μ و عملیات ترکیب و بازگشت‌پذیری اولیه ساخته شود.

۶) قضیه : یک تابع، بازگشتی μ است اگر و فقط اگر محاسبه‌پذیر باشد.

ضمیمه۲. لیستی از زبان های نوع اول، دوم و سوم

الف : تعدادی زبان منظم (نوع ۳)

۱. $L=\left\{x\in\{0,1\}^*:|x|_0 \mod 2=0\right\}$

۲. $L=\left\{a^i:i\geq 3\right\}\cup\left\{a^iba^j:i\geq 0,j\geq 1\right\}$

۳. $L=\left\{0^n1^m:n\geq 2,m\geq 3\right\}$

۴. $L=\left\{a^nb^m:n,m\geq 0\right\}$

۵. $L=\left\{a^{2i}b^{3j}:i,j\geq 1\right\}$

۶. $L=\left\{(ab)^n:n\geq 0\right\}$

۷. $L=\left\{a^nb^m:n\leq m,m\leq 2^{1024}\right\}$

۸. $L=\left\{a^nb^mc^kd^l:n+m+k+l\geq 2^{10}\right\}$

۹. $L=\left\{awa:w\in\{a,b\}^*\right\}$

۱۰. $L=\left\{aw_1aaw_2a:w_1,w_2\in\{a,b\}^*\right\}$

۱۱. $L=\left\{w:n_a(w) \mod 3>n_b(w) \mod 3\right\}$

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

۱۳. $L=\left\{a^n:n\geq 0,n\neq 4\right\}$

۱۴. $L=\left\{a^n:n=i+jk,i,j\ \text{ثابت},j=0,1,2,....\right\}$

۱۵. $L=\left\{a^n:n\geq 1\right\}\cup\left\{b^ma^k:m\geq 0,k\geq 0\right\}$

۱۶. $L=\left\{a^nb:n\geq 0\right\}\cup\left\{b^na:n\geq 1\right\}$

۱۷. $L=\left\{a^nb^m:a\geq 4,m\leq 3\right\}$

۱۸. $L=\left\{a^nb^m:n\geq 1,m\geq 1\ ,\ nm\geq 3\right\}$

۱۹. $L=\left\{w:\ |n_a(w)-n_b(w)|\ \text{فرد می‌باشد}\right\}$

۲۰. $L=\{a^n b^l a^k : n+l+k>5\}$

۲۱. $L=\{uww^R v : u,v,w\in\{a,b\}^+\}$

ب : تعدادی زبان مستقل از متن (نوع ۲)

۱. $L=\{ww^R : w\in\{a,b\}^*\}$

۲. $L=\{a^n b^n : n\ge 0\}$

۳. $L=\{ab(bbaa)^n bba(ba)^n : n\ge 0\}$

۴. $L=\{a^n b^n : n\ne m\}$

۵. $L=\{w\in\{a,b\}^* : n_a(w)=n_b(w), n_a(v)\ge n_b(v)$ که v هر پیشوندی از w می‌باشد$\}$

۶. $L=\{a^{2n} b^m : n\ge 0, m\ge 0\}$

۷. $L=\{a^n b^m : n\le m+3, n\ge 0, m\ge 0\}$

۸. $L=\{a^n b^m : n\ne m-1, n\ge 0, m\ge 0\}$

۹. $L=\{a^n b^m : 2n\le m\le 3n, n\ge 0, m\ge 0\}$

۱۰. $L=\{w\in\{a,b\}^* : n_a(w)\ne n_b(w)\}$

۱۱. $L=\{w\in\{a,b\}^* : n_a(v)\ge n_b(v)$ و v هر پیشوندی از w است$\}$

۱۲. $L=\{w\in\{a,b\}^* : n_a(w)=2n_b(w)+1\}$

۱۳. $L=\{a^n b^m c^k : n=m$ یا $m\le k, n\ge 0, m\ge 0, k\ge 0\}$

۱۴. $L=\{a^n b^m c^k : n=m$ یا $m\ne k, n\ge 0, m\ge 0, k\ge 0\}$

۱۵. $L=\{a^n b^m c^k : k=|n-m|, n\ge 0, m\ge 0, k\ge 0\}$

۱۶. $L=\{w\in\{a,b,c\}^* : n_a(w)+n_b(w)\ne n_c(w)\}$

۱۷. $L=\{a^n b^n c^k : k\ge 3, n\ge 0\}$

۱۸. $L=\left\{a^n ww^R b^n : w\in\sum^*, n\ge 1\right\}$

۱۹. $L=\left\{uvwv^R : u,v,w\in\{a,b\}^+, |u|=|w|=2\right\}$

۲۰. $L=\left\{w_1cw_2 : w_1,w_2\in\{a,b\}^+, w_1\ne w_2^R\right\}$

۲۱. $L=\left\{\{a^nb^nc^m\}\cup\{a^nb^mc^m\} : n,m\ge 0\right\}$

۲۲. $L=\left\{a^nb^mc^{n+m} : n\ge 0, m\ge 0\right\}$

۲۳. $L=\left\{a^nb^{n+m}c^m : n\ge 0, m\ge 1\right\}$

۲۴. $L=\left\{w : 2n_a(w)\le n_b(w)\le 3n_a(w)\right\}$

۲۵. $L=\left\{w_1cw_2 : w_1,w_2\in\{a,b\}^*, w_1\ne w_2\right\}$

۲۶. $L=\left\{a^nb^na^mb^m : n\ge 0, m\ge 0\right\}$

۲۷. $L=\left\{w\in\{a,b,c\}^* : n_a(w)+n_b(w)=n_c(w)\right\}$

۲۸. $L=\left\{w\in\{a,b,c\}^* : n_a(w)\ne n_b(w)\ne n_c(w)\right\}$

۲۹. $L=\{x^Rcy : x,y\in\{a,b\}^*$, y زیررشته‌ای از x می‌باشد$\}$

۳۰. $L=\left\{uv^iwx^iy : i\ge 0, u,v,w,x,y\in\sum^*\right\}$

۳۱. $L=\{x\in\{a,b,c\}^* : |x|_a\ne|x|_b$ یا $|x|_b\ne|x|_c$ یا $|x|_a\ne|x|_c\}$

۳۲. $L=\{a^ib^jc^k : i=2j$ یا $j=2k\}$

۳۳. $L=\left\{a^nb^mc^{2(n+m)} : n,m\ge 0\right\}$

۳۴. $L=\{a^ib^jc^k : i,j,k\ge 0, (i=j$ یا $i=k)\}$

۳۵. $L=\left\{x\in\{0,1\}^* : x=x^R\right\}$

۳۶. $L=\left\{a^mb^nc^pd^q : m+n=p+q\right\}$

۳۷. $L=\left\{w\in\{a,b\}^* : |w|_a\le|w|_b\le 2|w|_a\right\}$

ج: تعدادی زبان وابسته به متن (نوع ۱)

۱. $L = \{a^n b^n c^n : n \ge 1\}$

۲. $L = \{a^{n!} : n \ge 0\}$

۳. $L = \{a^n : n = m^2, m \ge 1\}$

۴. $L = \{a^n :$ n یک عدد اول است $\}$

۵. $L = \{a^n :$ n یک عدد اول نیست $\}$

۶. $L = \{ww : w \in \{a,b\}^+\}$

۷. $L = \{w^n : w \in \{a,b\}^+, n \ge 1\}$

۸. $L = \{www^R : w \in \{a,b\}^+\}$

۹. $L = \{a^{n+1} b^n c^{n-1} : n \ge 1\}$

۱۰. $L = \{a^n b^n a^{2n} : n \ge 1\}$

۱۱. $L = \{a^n b^m c^n d^m : n \ge 1, m \ge 1\}$

۱۲. $L = \{w : n_a(w) = n_b(w) = n_c(w)\}$

۱۳. $L = \{w : n_a(w) = n_b(w) < n_c(w)\}$

۱۴. $L = \{wuw : w, u \in \{a,b\}^+\}$

۱۵. $L = \{ww^R w : w \in \{0,1\}^+\}$