نظریه زبان‌ها و ماشین‌ها

مؤلف : فرشید شیرافکن

دانشجوی دکترای بیوانفورماتیک دانشگاه تهران

ناشر: سازمان علمی آموزش فرادرس

بزرگترین پلتفرم آموزش آنلاین ایران

وب: www.faradars.org

تقدیم به:

روح پاک پدرم

- فرشید شیرافکن

سخن ناشر

در عین تمام نقدهای وارد شده به کنکور، هنوز راه‌حلی عملی که در جمیع جوانب، بهتر از سبک کوتاه و چند گزینه‌ای سؤالات باشد؛ ارائه نشده است. همین موضوع، کنکور را به ویژه کنکور کارشناسی ارشد به عنوان یک آزمون متمرکز و سراسری، از اهمیت دوچندانی برخوردار می‌کند.

یکی از آسیب‌های همراه با این آزمون سراسری این است که فضای رقابتی آن با ایجاد مؤسسات گوناگون، به سرعت از فضای یک رقابت علمی تبدیل به فضای رقابت اقتصادی می‌شود؛ به گونه‌ای که هزینه سرسام آور کلاس‌ها، دوره‌ها و منابع مرتبط با آزمون، از عهده بسیاری از دانشجویان خارج می‌شود. دانشجویانی که در عین استعداد تحصیلی بالا، در میدان رقابت مالی تحمیلی، در عین تمام شایستگی‌های خود، قدرت ادامه مسیر را از دست می‌دهند یا به نتیجه‌ای که در فضای مساوی مالی برای همه باید به آن می‌رسیدند، دست نمی‌یابند.

یکی از اهداف و آرمان‌های فرادرس به عنوان بزرگ‌ترین پروژه آموزش دانشگاهی اجرا شده بر بستر وب کشور، ایجاد دسترسی همگانی و یکسان به آموزش و دانش؛ مستقل از جغرافیا، زمان و سطح مالی دانشجویان بوده است. سیاست کاری فرادرس در راستای این آرمان، انتشار آموزش‌های ویدئویی تخصصی و دانشگاهی رایگان و یا بسیار کم هزینه، با تدریس مجرب‌ترین اساتید داخل و خارج کشور بوده است.

ما با انتشار رایگان این کتاب (به همراه نزدیک به ده کتاب رایگان دیگر) یکی از گام‌های دیگر خود را در راستای آرمان فرادرس برداشتیم. کتاب حاضر که حاصل نزدیک به دو دهه تدریس و پژوهش و تألیف مؤلف و مدرس فرادرس می‌باشد؛ در عین هزینه‌های بالای تألیف و آماده‌سازی، به جای انتشار و فروش؛ با تأمین مالی و سرمایه‌گذاری فرادرس به عنوان ناشر، به صورت کاملاً رایگان منتشر می‌شود. ما در گام‌های بعدی نیز تلاش خواهیم کرد که تا هر جا بتوانیم، حتی شده یک کتاب مرجع دیگر و بیشتر را با پرداخت هزینه، آزادسازی کرده و به صورت رایگان منتشر کنیم.

مؤلفین و ناشرینی که تمایل به واگذاری حق انتشار کتاب خود به فرادرس را دارند، می توانند با ایمیل ebooks@faradars.org مکاتبه نمایند. ما این کتاب‌ها را با پرداخت هزینه تألیف به مدرس و ناشر، به صورت رایگان منتشر خواهیم کرد تا همه دانشجویان مستقل از سطح مالی، به منابع مفید آزمون دسترسی داشته باشند. همچنین اگر ایده و نظری در خصوص کتاب‌های رایگان فرادرس داشته باشید، خوشحال می‌شویم که آن را با ایمیل ebooks@faradars.org مطرح نمایید.

سازمان علمی آموزش فرادرس

بزرگترین پلتفرم آموزش آنلاین ایران

وب: www.faradars.org

منبع مطالعاتی تکمیلی مرتبط با این کتاب

آموزش نظریه زبان‌ها و ماشین‌ها

در این درس با سه موضوع "زبان، گرامر و ماشین" آشنا می‌شوید. این درس پیش نیاز درس طراحی کامپایلر است. با یادگیری زبان‌ها و گرامرها می‌توانید نحوه کار کامپایلر و همچنین طراحی زبان‌های برنامه‌سازی را متوجه شد. یادگیری این درس بدون مدرس کار ساده‌ای نمی‌باشد و ما در این آموزش تجربه حداقل پانزده سال تدریس این درس را در اختیار شما گذاشته‌ایم. به امید اینکه دعای خیری برای ما شود.

مدرس: مهندس فرشید شیر افکن

مدت زمان: ۹ ساعت

جهت مشاهده آموزش ویدئویی این آموزش – کلیک کنید **faradars.org /fvsft110**

درباره مدرس

مهندس فرشید شیرافکن کارشناس ارشد مهندسی کامپیوتر گرایش نرم‌افزار است و در حال حاضر دانشجوی دکترای بیوانفورماتیک دانشگاه تهران هستند. ایشان از مدرسین نمونه در زمینه ارائه و آموزش دروس دانشگاهی انتخاب شده‌اند.

ایشان مشاور کنکور هستند و بیش از ۳۰ کتاب در زمینه کنکور رشته کامپیوتر تألیف نموده‌اند. ایشان در حال حاضر به عنوان یکی از برترین مدرسین فرادرس از جهت کمیت و کیفیت دروس ارائه شده، نزدیک به ۲۰ عنوان درسی را در قالب آموزش ویدئویی از طریق فرادرس منتشر کرده‌اند. این مجموعه دروس تا کنون مورد استفاده ده‌ها هزار دانشجوی سراسر کشور قرار گرفته اند.

مشاهده همه آموزش های تدریسی و تالیفی توسط مؤلف کتاب – کلیک کنید.

شماره همراه مدرس: ۰۹۱۲۱۹۷۲۰۲۸ Email: fshirafkan51@gmail.com

کتب رایگان دیگر از این مجموعه آموزشی

۱. **آموزش زبان برنامه سازی ++C – کلیک کنید (+)**
۲. **آموزش شیءگرایی در سی پلاس پلاس – کلیک کنید (+)**
۳. **آموزش پایگاه داده‌ها – کلیک کنید (+)**
۴. **آموزش ساختمان داده‌ها – کلیک کنید (+)**
۵. **آموزش سیستم عامل – کلیک کنید (+)**
۶. **آموزش ذخیره و بازیابی اطلاعات – کلیک کنید (+)**

برای دانلود رایگان این مجموعه کتب، به لینک زیر مراجعه کنید:

http://faradars.org/computer-engineering-exam

دسته‌بندی موضوعی آموزش‌های فرادرس، در ادامه آمده است:

برنامه‌نویسی برنامه نویسی – کلیک (+)	آموزش‌های دانشگاهی و تخصصی آموزش های دانشگاهی و تخصصی – کلیک (+)	هوش مصنوعی و یادگیری ماشین هوش مصنوعی و یادگیری ماشین – کلیک (+)	مهندسی برق الکترونیک و روباتیک مهندسی برق الکترونیک و روباتیک – کلیک (+)
دروس دانشگاهی دروس دانشگاهی – کلیک (+)	مباحث مشترک مباحث مشترک – کلیک (+)	مهارت‌های دانشگاهی مهارت های دانشگاهی – کلیک (+)	نرم‌افزارهای تخصصی نرم افزارهای تخصصی – کلیک (+)
مهندسی نرم‌افزار مهندسی نرم افزار – کلیک (+)	نرم‌افزارهای عمومی نرم افزارهای عمومی – کلیک (+)	طراحی و توسعه وب طراحی و توسعه وب – کلیک (+)	آموزش‌های عمومی آموزش های عمومی – کلیک (+)

فهرست مطالب

ابهام در گرامر و زبان

ساده سازی گرامرهای مستقل از متن

حذف متغیرها و قوانین بی فایده

حذف قوانین

حذف قوانین واحد

فرم های نرمال گرامر مستقل از متن

فرم نرمال چامسکی

فرم نرمال گریباخ

فصل ۶ : اتوماتای پشته ای (DPDA,NPDA)

اتوماتای پشته ای نامعین

تابع انتقال

پیکر بندی لحظه ای

اتوماتای پشته ای معین

تشخیص مستقل از متن بودن یک زبان

زبان مستقل از متن معین

ساخت اتوماتای پشته ای با استفاده از گرامر در فرم گریباخ

فصل ۷ : ماشین های تورینگ(TM)

ماشین تورینگ استاندارد

ماشین تورینگ در نقش پذیرنده زبان

ماشین تورینگ به عنوان مترجم

مدل های دیگر ماشین تورینگ

سکون دار

با نوار نیمه نامتناهی

آف لاین

با حافظه پیچیده تر

چند نواره

چند بعدی

نامعین

آتاماتای کراندار خطی(LBA)

فصل ۸ : زبان های بازگشتی-گرامر بدون محدویت و حساس به متن

زبان های بازگشتی و بازگشتی شمارش پذیر

گرامر بدون محدودیت

گرامر حساس به متن

ارتباط بین زبان ها، گرامرها و ماشین ها

سلسله مراتب چامسکی

بررسی بسته بودن زبان ها تحت عملگرها

فصل ۹ : تصمیم پذیری – کاهش پذیری

زبان های تصمیم نا پذیر

زبان های تصمیم پذیر

تصمیم پذیری در زبان های منظم

برشمارنده-

کاهش پذیری

فصل ۱

عبارت منظم – زبان منظم

در این کتاب، سه مفهوم اصلی یعنی زبان، گرامر(ابزار تولید زبان) و ماشین(ابزار پذیرش زبان) بررسی خواهند شد. مطالبی که بررسی می شوند عبارتند از:

۱- زبان و گرامر منظم و ماشین پذیرنده رشته‌های زبان منظم یعنی ماشین متناهی(FA)

۲- زبان و گرامر مستقل از متن و ماشین پذیرنده رشته‌های زبان مستقل از متن یعنی ماشین پشته‌ای(PDA)

۳- زبان و گرامر حساس به متن و ماشین تورینگ(TM) و ....

عبارت منظم

قبل از ورود به بحث زبان، عبارت منظم را بررسی می کنیم. عبارت منظم، ترکیبی از سمبل‌ها(از قبیل الفبا، پرانتز، عملگر بستار ، عملگر الحاق) می باشد. عبارت منظم با انجام متوالی برخی قوانین بازگشتی روی اجزاء پایه ای ایجاد می شود.

به طور نمونه a^+b یک عبارت منظم در الفبای $\Sigma=\{a,b\}$ می باشد، که معرف رشته‌هایی است که با یک یا چند حرف a شروع شده و به b ختم می شوند. مانند ab ، aab و aaab .

به عنوان مثالی دیگر عبارت منظم ab^+ ، معرف رشته‌هایی است که با یک حرف a شروع شده و به یک یا چند b ختم می شوند. مانند ab ، abb و abb. اگر به جای بستار مثبت از بستار ستاره استفاده شده بود، یعنی عبارت منظم به صورت ab^* باشد، آنگاه، معرف رشته‌هایی است که با یک حرف a شروع شده و به صفر یا یک یا چند b ختم می شوند. یعنی از b می‌توان استفاده نکرد و رشته a را تولید کرد که توسط ab^+ قابل تولید نبود.

تعریف

Σرا الفبای مفروض در نظر می گیریم. آنگاه:

۱- ϕ ، λ و $a\in\Sigma$ همگی عبارات منظم هستند. این عبارات را عبارات منظم پایه می‌خوانیم.

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

۳- یک رشته فقط و فقط در صورتی عبارت منظم است که با بکارگیری تعداد محدودی از قوانین بند (۲)، از عبارات منظم پایه بدست آیند.

چنانچه Σ یک الفبا باشد، آنگاه از Σ^+ برای نمایش مجموعه رشته‌های بدست آمده از الحاق چند سمبل دیگر و از Σ^* برای نمایش مجموعه رشته‌های بدست آمده از الحاق صفر یا بیشتر سمبل دیگر استفاده می‌کنیم. مجموعه Σ^* همواره حاوی λ خواهد بود. به عبارتی: $\Sigma^* = \Sigma^+ + \{\lambda\}$.

همواره Σ^+ و Σ^* نامتناهی هستند.

مثال

اگر الفبا شامل یک حرف a باشد، یعنی $\Sigma = \{a\}$ ، آنگاه :

$\Sigma^+ = \{a, aa, aaa, ...\}$ و $\Sigma^* = \{\lambda, a, aa, aaa, ...\}$

چند نکته:

۱- در عبارت منظم اولویت ستاره از همه بیشتر است، سپس اولویت اتصال بیشتر بوده، و در نهایت اجتماع قرار دارد، مگر اینکه از پرانتز برای تغییر اولویت استفاده شود.

۲- تعداد سمبل های موجود در رشته را طول رشته می گویند. طول رشته w با $|w|$ نشان داده می‌شود.

۳- رشته تهی (λ)، رشته ای است که هیچ سمبولی ندارد. برای هر رشته w داریم: $\lambda.w = w.\lambda = w$

۴- طول رشته تهی که با λ نشان داده می شود، برابر صفر است. ($|\lambda| = 0$)

۵- اگر r یک عبارت منظم باشد، آنگاه : $r.\lambda = r$

۶- اگر r یک عبارت منظم باشد، آنگاه : $r + \phi = r$

مثال

آیا رشته‌های aa و $abbba$ و $abab$ را می‌توان از عبارت منظم ab^*a تولید کرد؟

حل: aa: بله- به علت استفاده از بستار ستاره برای b ، می‌توان از b استفاده نکرد.

$abbba$: بله- به علت استفاده از بستار ستاره برای b ، می‌توان چند بار از b استفاده کرد.

$abab$: خیر.

مثال

آیا رشته‌های aa و aba و $abbbba$ را می‌توان از عبارت منظم $a(bb)^*a$ تولید کرد؟

حل: aa : بله- از bb داخل پرانتز، استفاده نمی کنیم.

aba : خیر- تعداد b های تولید شده توسط عبارت منظم، باید زوج باشد.

$abbbba$: بله- از bb داخل پرانتز، دو مرتبه استفاده می کنیم. ■

مثال

رشته‌های تولید شده با حداکثر طول ۴ ، توسط عبارت منظم a^+b و a^*b عبارتند از :

$$a^+b = \{ab, aab, aaab\} \ , \quad a^*b = \{b, ab, aab, aaab\}$$

■

مثال

رشته‌های تولید شده با طول ۳ توسط عبارت منظم $(a+b)^+$ عبارتند از :

$$(a+b)^+ = \{aaa, aab, aba, abb, baa, bab, bba, bbb\}$$

■

مثال

آیا رشته $aaaabbaa$ از عبارت منظم $(ab^*aa)^+$ قابل تولید است؟

حل: بله – به علت استفاده از بستار مثبت برای کل عبارت داخل پرانتز، می‌توان چند بار از این عبارت استفاده کرد. در بار اول aaa و در بار دوم abbaa را تولید می‌کنیم.■

مثال

نحوه تولید رشته‌های $abaa$ و $aaabaab$ و $aaababbba$ به کمک عبارت منظم $(ab+ab^*a)^+$ را مشخص کنید.

حل: عبارت داده شده دارای دو قسمت ab و ab^*a می‌باشد که به علت استفاده از بستار مثبت می‌توان چندین بار از آن استفاده کرد.

$abaa$: انتخاب ab از قسمت اول عبارت منظم و aa از قسمت دوم.

$aaabaab$: انتخاب aa از قسمت دوم، سپس aba از قسمت دوم و ab از قسمت اول .

$aaabababba$: انتخاب aa از قسمت دوم، سپس $abab$ با دو بار استفاده از قسمت اول و در نهایت $abba$ از قسمت دوم.

روابط مقابل برقرارند: $(a^+)^* = (a^*)^+ = (a^*)^* = a^*$

روابط زیر برقرارند:

$$\phi^+ = \phi \,, \phi^* = \{\lambda\} \,,\ \lambda^* = \lambda \,,\ \lambda^* - \phi^* = \phi \ ,\ \lambda - \phi^* = \phi,\ \lambda^*.\phi^* = \lambda$$

به جای $(\alpha+\beta)^*$ هر یک از عبارت‌های زیر را می توان قرار داد:

$$(\alpha^* + \beta)^* \ ,\ (\alpha + \beta^*)^* \ ,\ (\alpha^* + \beta^*)^* \ ,\ \alpha^*(\alpha + \beta)^* \ ,\alpha^*(\alpha+\beta)^*\beta^*$$

$$(\alpha^*\beta^*)^* \ ,\ \ \beta^*(\alpha\beta^*)^*,\ \ \alpha^*(\beta\alpha^*)^*,(\beta^*\alpha^*)^*$$

مثال

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

حل: می توان به جای $(a^*+b)^*$ و $(a^*+b^*)^*$ از $(a+b)^*$ استفاده کرد:

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

مثال

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

حل: می‌توان به جای $(a^*b^*)^*$ و $(b^*a^*)^*$ از $(a+b)^*$ استفاده کرد:

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

مثال

آیا رابطه $(ab)^*a=a(ba)^*$ برقرار است؟

حل: بله- تمام رشته‌های قابل تولید توسط هر کدام، به کمک دیگری نیز قابل تولید است.■

مثال

فرض کنید a و b عبارت‌های منظم باشند. آیا رابطه زیر برقرار است؟

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

حل: بله- تمام رشته‌های قابل تولید توسط هر کدام، به کمک دیگری نیز قابل تولید است.■

مثال

آیا رابطه $(a+b)^*=(a^*b+ab^*)^*$ برقرار است؟

حل: بله- تمام رشته‌های قابل تولید توسط هر کدام، به کمک دیگری نیز قابل تولید است.■

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

معکوس عبارت منظم

برای پیدا کردن معکوس(متمم) یک عبارت منظم، از روابط زیر استفاده می کنیم:

$$(\alpha^*)^R = (a^R)^* \quad (\alpha+\beta)^R = \alpha^R + \beta^R \qquad (\alpha\beta)^R = \beta^R \alpha^R$$

مثال

معکوس عبارت منظم $ab(c+de)$ را مشخص کنید.

حل: $[(ab).(c+de)]^R = (c+de)^R.(ab)^R = (c^R + (de)^R).(b.a) = (ed+c).ba$

تذکر: در واقع برای پیدا کردن معکوس یک عبارت منظم، می توان عبارت را از انتها به ابتدا نوشت.■

مثال

معکوس $(a^*+b)ba^*$ را مشخص کنید.

حل: کافی است عبارت را معکوس نوشت: $(a^*+b)ba^* \Rightarrow a^*b(b+a^*)$ ■

زبان

هر مجموعه از رشته‌های روی یک الفبای Σ را یک زبان می گویند. عملیات قابل انجام بر روی زبان ها عبارتنـد از: اجتمـاع، اشتراک، تفاضل، اتصال، معکوس، مکمل(متمم)، همریختی و تقسیم راست.

مثال

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

اجتماع و اشتراک

از آنجا که زبان ها، مجموعه هستند، اجتماع و اشتراک دو زبان، به راحتی قابل تعریف است.

مثال

اشتراک و اجتماع $L_1=\{a^nb^m : n \ge 0, m \ge 0\}$ و $L_2=\{a^nb^n : n \ge 0\}$ را مشخص کنید.

حل: چند جمله از هر زبان را مشخص می‌کنیم:

$$L_1=\{\lambda, ab, a^5b^2, a^7b^7,...\} \quad , \quad L_2=\{\lambda, ab, a^2b^2, a^5b^5, a^7b^7,...\}$$

واضح است که زبان L_2 زیر مجموعه L_1 است. در نتیجه اشتراک آنها L_2 و اجتماع آنها L_1 است.■

مثال

اشتراک و اجتماع دو زبان $L_1=\{a^nb^k : n\neq k\}$ و $L_2=\{a^nb^k : n\geq 0, k\geq 0\}$ را مشخص کنید.

حل: چند جمله از هر زبان را مشخص می‌کنیم:

$$L_1=\{\lambda, a^5b^2, a^7b^4, \ldots\}$$

$$L_2=\{\lambda, ab, a^7b^4, a^5b^5, a^4b^4, a^5b^2, \ldots\}$$

واضح است که زبان L_1 زیر مجموعه L_2 است $(L_1\subseteq L_2)$،پس حاصل $L_1\cap L_2$ همان L_1 و حاصل $L_1\cup L_2$ همان L_2 است.■

مثال

اشتراک L_1 و L_2 را مشخص کنید.

$$L_1=\{a^nb^nc^m : n\geq 0, m\geq 0\}$$

$$L_2=\{a^nb^mc^m : n\geq 0, m\geq 0\}$$

حل: چند جمله از هر زبان را مشخص می‌کنیم:

$$L_1=\{\lambda, abc, abc^5, a^6b^6c^2, a^4b^4c^4, \ldots\}$$

$$L_2=\{\lambda, abc, a^4b^5c^5, a^4b^4c^4, a^3b^6c^6, \ldots\}$$

بنابراین اشتراک این دو زبان، شامل جمله هایی است که در آنها تعداد a و b و c برابر است، یعنی: $L_1\cap L_2=\{a^nb^nc^n : n\geq 0\}$

مثال

اجتماع دو زبان زیر را مشخص کنید.

$$L_1=\{a^nb^mc^n : n\geq m\geq 0\}$$

$$L_2=\{a^nb^mc^n : m> n\geq 0\}$$

حل: اجتماع دو زبان برابر است با:

$$L_1 \bigcup L_2 = \{a^n b^m c^n : n \ge 0, m \ge 0\}$$

■

مثال

اشتراک سه زبان زیر را مشخص کنید.

$$L_1 = \{a^n b^m a^n b^p : n, m, p \ge 0\}$$

$$L_2 = \{a^n b^n a^m b^p : n, m, p \ge 0\}$$

$$L_3 = \{a^n b^m a^p b^n : n, m, p \ge 0\}$$

حل: اشتراک این سه زبان برابر است با: $L_1 \bigcap L_2 \bigcap L_3 = \{a^n b^n a^n b^n : n \ge 0\}$

■

مثال

اشتراک $L_1 = \{a^n b^m : n \ge 0, m \ge 0\}$ و $L_2 = \{b^n a^n : n \ge 0\}$ را مشخص کنید.

حل: $L_1 \bigcap L_2 = \{a^n : n \ge 0\} \bigcup \{b^m : m \ge 0\}$ ■

اتصال

اتصال(الحاق) دو زبان به صورت زیر تعریف می شود:

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

مثال

تعداد اعضای اتصال $L_1 = \{10,1\}$ و $L_2 = \{01,011,11\}$ را تعیین کنید.

$$L_1L_2 = \{1001,10011,1011,101,1011,111\}$$

بنابراین L_1L_2 دارای پنج عضو می باشد. (1011 که دو بار تکرار شده را یکبار شمارش می کنیم.)

L^n به معنای اتصال L به تعداد n بار با خودش می‌باشد و $L^0 = \{\lambda\}$.

مثال

اگر $L = \{a^nb^n : n \geq 0\}$، آنگاه L^2 برابر است با :

$$L^2 = \{a^nb^na^kb^k : n \geq 0, k \geq 0\}$$

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

مثال

با فرض اینکه $L = \{awa : w \in \{a,b\}^*\}$، آنگاه L^2 برابر است با :

$$L^2 = \{aw_1aaw_2a : w_1, w_2 \in \{a,b\}^*\}$$

■

اتصال دو رابطه خاصیت جابجایی ندارد، یعنی : $L_1L_2 \neq L_2L_1$

عمل اتصال نسبت به اجتماع خاصیت پخشی دارد اما نسبت به اشتراک ندارد:

$$L_1(L_2 \cup L_3) = L_1L_2 \cup L_1L_3 \quad , \quad L_1(L_2 \cap L_3) \neq L_1L_2 \cap L_1L_3$$

معکوس

معکوس زبان L به صورت $L^R=\{w^R : w\in L\}$ تعریف می شود.روابط زیر برقرار می باشند:

الف- $(L^R)^R=L$ ب- $(L^R)^n=(L^n)^R$ ج- $((L^R)^n)^R=L^n$ د- $(L_1L_2)^R={L_2}^R{L_1}^R$

مثال

معکوس $L=\{a^nb^n : n\geq 0\}$، برابر است با:

$$L^R=\{b^na^n : n\geq 0\}$$

■

مکمل

مکمل(متمم) زبان L به صورت $\overline{L}=\Sigma^*-L$ تعریف می شود.

مثال

مکمل زبان $L=\{aa,bb\}$، را مشخص کنید. ($\Sigma=\{a,b\}$)

حل: زبان داده شده دارای دو رشته به طول ۲ است. مکمل این زبان، شامل همه رشته ها با طول صفر، یک، دو (به غیر از aa وbb)، سه و بزرگتر از سه است. این زبان را به صورت زیر نمایش می دهیم.

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

مثال

با فرض $\Sigma=\{a,b\}$، مکمل زبان $L=\{a^nb^n : n\in N\}$ برابر است با: $\overline{L}=\{a^xb^y : x\neq y\}$ ■

مثال

با فرض $\Sigma=\{0,1\}$ مکمل $L=\{0^n1^n0^n : n\in N\}$ برابر است

$$\overline{L}=\{0^n1^m0^k : n\neq m\neq k\}$$

بستار

برروی زبان L دو عمل بستار ستاره ای و بستار مثبت قابل انجام است:

$$L^*=L^0UL^1UL^2...=\bigcup_{i\geq 0}L^i$$

$L^{+} = L^{1} U L^{2} U L^{3} \ldots = \bigcup_{i\ge 1} L^{i}$

چند نکته:

۱- عملگر بستار نسبت به هیچیک از عملگرهای "اجتماع ، اشتراک ، تفاضل و الحاق" خاصیت پخشی ندارد. به عنوان نمونه : $(L_1 \cup L_2)^* \neq L_1^* \cup L_2^*$.

۲- برای هر زبان L داریم $\lambda \in L^*$ بنابراین : $\lambda \notin \overline{L^*}$

۳- هیچ زبانی وجود ندارد که در آن $\overline{L^*} = \overline{L}^*$.

۴- برای هر زبان L داریم: $\overline{L}^* \subset \overline{L^*}$.

هم ریختی

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

مثال

بـا فـرض $\Sigma=\{a,b\}$ و $\Gamma=\{a,b,c\}$ ، تـابع h را بصـورت $h(a)=ab$ و $h(b)=bbc$ تعریـف مـی‌کنیم. مطلـوب است $h(aba)$.

حل: کافی است در aba به جای a رشته ab و به جای b رشته bbc را قرار دهیم: $h(aba)=abbbcab$ ■

مثال

با فرض $\Sigma=\{a,b\}$ و $\Gamma=\{b,c,d\}$ ، h را به صورت $h(a)=bc$ و $h(b)=bdc$ تعریف می‌کنیم. اگر L زبان منظمی باشد که عبارت منظم $r=a^*(a+b^*)$ آن را معرفی کند، آنگاه عبارت منظمی کـه زبـان مـنظم $h(L)$ را معرفـی کنـد را مشخص کنید.

حـل: کـافی اسـت در عبـارت مـنظم داده شـده، بـه جـای a از رشـته bc و بـه جـای b از رشـته bdc اسـتفاده کـرد:
$(bc)^*(bc+(bdc)^*)$

خـانواده زبانهـای مـنظم تحـت همریختـی بسـته اسـت. بـه عبـارتی اگـر h یـک همریختـی و L یـک زبـان مـنظم باشد،آنگاه $h(L)$ ، منظم خواهد بود.

تقسیم راست

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

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

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

مثال

حاصل $\frac{L_1}{L_2}$ را تعیین کنید.

$L_1 = \{001101, 0101010, 011100, 101010\}$

$L_2 = \{01, 10\}$

حل: باید در رشته های L_1،پیشوندهایی را انتخاب کنیم که به 01 یا 10 ختم شوند. در زیر این پیشوندها خط کشیده شده است:

$L_1 = \{\underline{0011}01 \ , \ \underline{01010}10 \ , \ 011100 \ , \ \underline{1010}10\}$

بنابراین خواهیم داشت:

$L_1 / L_2 = \{0011, 01010, 1010\}$

■

مثال

حاصل تقسیم راست L_1 / L_2 را تعیین کنید.

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

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

پ- $L_1 = L(a^*b^*c)$, $L_2 = L(b^*c)$

حل: باید در رشته های L_1،پیشوندهایی را انتخاب کنیم که به L_2 ختم شوند.

الف: $\{a^*ba^*\}$ ب: $\{a^*\}$ پ: $\{a^*b^*\}$ ■

اگر L_1 و L_2 زبان های منظم باشند، آنگاه L_1 / L_2 نیز منظم خواهد بود. پس خانواده زبانهای منظم تحت تقسیم راست، بسته است.

زبان منظم

اگر بتوان برای زبانی، یک عبارت منظم نوشت،آن زبان منظم است. به ازای هر زبان منظم، یک عبارت منظم و به ازای هر عبارت منظم، یک زبان منظم وجود دارد. به عبارتی اگر L یک زبان منظم باشد، آنگاه عبارت منظمی به ازای r وجود خواهد داشت، بطوریکه L=L(r) .

برای بیان عبارت منظم یک زبان، دو شرط زیر باید برقرار باشد:

۱- هر رشته ای که عبارت منظم بیان می کند، شرط توصیف را داشته باشد.

۲- هر رشته ای که شرط توصیف را دارد، توسط عبارت منظم قابل بیان باشد.

با فرض اینکه r_1 و r_2 عبارت منظم و $L(r)$، زبان مربوط به عبارت منظم r باشد، خواهیم داشت:

الف- $L(r_1^*) = (L(r_1))^*$

ب- $L(r_1.r_2) = L(r_1).L(r_2)$

ج- $L(r_1 + r_2) = L(r_1) \cup L(r_2)$

مثال

عبارت منظم $r=(a+b)^*(a+bb)$ ، در الفبای $\sum=\{a,b\}$، به چه زبانی اشاره دارد؟

حل: بخش $(a+b)^*$ به معنای هر رشته ای از aها و bها است. بخش $(a+bb)$، بیانگر یک a یا دو b می‌باشد. در نتیجـه ، L(r) مجموعه تمام رشته هایی است که به یک a یا یک bb ختم می شوند.

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

مثال

عبارت منظم $r=(aa)^*(bb)^*b$ ، در الفبای $\sum=\{a,b\}$، به چه زبانی اشاره دارد؟

حل: به مجموعه تمام رشته هایی با تعداد زوج a که قبل از تعداد فردی از b ها می‌آیند:

$$L(r)=\{a^{2n}b^{2k+1}:n\geq 0,k\geq 0\}$$

مثال

عبارت منظم $r=a^{+}(bb)^{+}$ ، در الفبای $\Sigma=\{a,b\}$، چه زبانی را توصیف می کند؟

$$L(r)=\{a^{n}b^{2k}:n\geq 1,k\geq 1\}$$

■

مثال

عبارت منظم $r=abbb^{+}(a+b)^{+}$ ، در الفبای $\Sigma=\{a,b\}$، به چه زبانی اشاره دارد؟

حل: $L=\{ab^{n}w:n\geq 3,w\in\{a,b\}^{+}\}$ ■

مثال

عبارت منظم زبان $L=\{a^{n}b^{m}:(n+m)\ \ is\ \ even\}$ را بنویسید.

حل: برای اینکه مجموع تعداد a ها و b ها زوج باشد، باید تعداد هر دو زوج و یا تعداد هر دو فرد باشد.

$$r=(aa)^{*}(bb)^{*}+(aa)^{*}a(bb)^{*}b$$

■

مثال

عبارت منظم زبان $L=\{a^{n}b^{m}:(n+m)\ \ is\ \ odd\}$ را بنویسید.

حل: برای اینکه مجموع تعداد a ها و b ها فرد باشد، باید تعداد a فرد و b زوج و یا تعداد a زوج و b فرد باشد.

$$r=(aa)^{*}a(bb)^{*}+(aa)^{*}(bb)^{*}b$$

مثال

عبارت منظمی برای $L=\{w:|w|\ \ \mathrm{mod}\ \ 3=0\}$ ، در الفبای $\Sigma=\{a,b\}$، بنویسید.

حل: برای اینکه طول رشته مضرب ۳ باشد، تمام رشته های ممکن با طول سه را ایجاد و تکرار می کنیم:

$$r=((a+b)(a+b)(a+b))^{*}$$

می توان عبارت را به صورت $(\Sigma\Sigma\Sigma)^{*}$ نیز نشان داد.■

مثال

عبـــارت منظمـــی بـــرای $L=\{w_1bw_2 : |w_1| \bmod 2 = |w_2| \bmod 2, w_1,w_2 \in \{a\}^*\}$ ، در الفبـــای $\Sigma=\{a,b\}$، بنویسید.

حل: رشته های این زبان شامل یک b و تعداد زوجی a هستند:

$$r=(aa)^*b(aa)^*+a(aa)^*ba(aa)^*$$

مثال

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

حل: مساله را به حالتهای m=0,1,2,3 تقسیم می کنیم. تعداد ۴ یا بیشتر a تولید کرده و پس از آن به تعداد لازم b قرار می دهیم: $r=aaaaa^*(\lambda+b+bb+bbb)$ ■

مثال

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

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

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

برای هر زبان، تعداد نامحدودی عبارت منظم وجود دارد.

مثال

عبارت های منظمی در الفبای $\Sigma=\{a,b\}$ برای هر یک از حالات زیر بنویسید.

الف- با aa شروع یا به bb ختم شود.

ب- با aa شروع و به bb ختم شود.

پ- به b یا ba ختم شوند.

ت- سومین نماد از راست b باشد.

ث- تعداد a بر ۳ بخش پذیر باشد.

ج- قبل یا بعد از هر a یک b باشد.

حل: منظور از x در حل این مثال، همان $(a+b)$ است.

الف: $aax^* + x^*bb$ ب : aax^*bb

پ: $x^*(b+ba)$ ت: x^*bxx

ث : $b^*(ab^*ab^*ab^*)^+$ ج: $(ab+ba+b+aba)^*$ ■

مثال

عبارت‌های منظمی در الفبای $\Sigma = \{0,1\}$ برای هر یک از حالات زیر بنویسید.

الف- شامل 0 نباشد.

ب- دقیقا دو تا 1 داشته باشد.

پ- با 1 شروع شده و شامل دو 0 متوالی نباشد.

ت- تعداد زوجی 0 داشته باشد.

ث- دارای هیچ دو 0 متوالی نباشد.

حل:

الف : 1^*

ب : $0^*10^*10^*$

پ : $(1+10)^*$

ت : $(1^*01^*01^*)^* + 1^*$

ث : $(1+01)^*(0+1^*)$ یا $(1+01)^*(0+\lambda)$

مثال

عبارت‌های منظمی در الفبای $\Sigma=\{0,1\}$ برای هر یک از حالات زیر بنویسید.

الف- حداقل یک عدد 1 داشته باشد.

ب- حداقل دو تا 1 داشته باشد.

پ- حداقل دو تا 1 متوالی داشته باشد.

ت- دارای زیر رشته 001 باشد.

ث- به 101 ختم شود.

ج- تعداد 0 و تعداد 1 ، زوج باشد.

حل: منظور از x همان $(0+1)$ است.

الف : x^*1x^* ب : $x^*1x^*1x^*$ پ : x^*11x^*

ت : x^*001x^* ث : x^*101 ج : $(00+11+(01+10)(00+11)^*(01+10))^*$

مثال

یک عبارت منظم برای هر یک از حالتهای زیر در الفبای $\Sigma=\{a,b,c\}$ بنویسید.

الف- طول رشته زوج است. ب- طول رشته مضرب سه است.

پ- نماد شروع و پایان یکی باشد. ت- حداقل شامل یکی از سمبل‌های الفبا باشد.

حل: در حل این مثال، منظور از x همان $(a+b+c)$ است.

الف : $(xx)^*$

ب : $(xxx)^*$

پ : $a+b+c+ax^*a+bx^*b+cx^*c$

ت:

$x^*ax^*bx^*cx^* + x^*ax^*cx^*bx^* + x^*bx^*ax^*cx^* + x^*bx^*cx^*ax^* + x^*cx^*ax^*bx^* + x^*cx^*bx^*ax^*$ ■

مثال

یک عبارت منظم بنویسید که نمایش یک عدد صحیح یا اعشاری بدون علامت باشد.

حل: با فرض D={0,1,2,3,4,5,6,7,8,9} ، عبارت به صورت $D^+ + D^+.D^* + D^*.D^+$ است. مثال هایی از این رشته ها عبارتند از: 04. , .3 , 2.254 , 12 ■

مجموعه تمام اعداد حقیقی در زبان برنامه نویسی C ، یک زبان منظم است.

مثال

آیا زبان $L = \{a^n b^k : n,k \geq 0\}$ منظم است؟

حل: بله- چون زبان L همان a^*b^* است. ■

مثال

کدام یک از زبان‌های زیر منظم هستند؟

$L_1 = \{uww^R v : u,v,w \in \{a,b\}^+\}$

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

حل: فقط زبان L_1 منظم است. چون می توان یک عبارت منظم برای آن نوشت:

$r_1 = (a+b)^+(aa+bb)(a+b)^+$

■

مثال

کدام یک از زبان‌های زیر منظم هستند؟

$L_1 = \{ww^R : w \in \{a\}^*\}$

$L_2 = \{ww^R : w \in \{a,b\}^*\}$

حل: زبان L_1 منظم است.

■

مثال

کدام یک از زبان‌های زیر منظم هستند؟

$L_1 = \{a^n b^m a^k : n + m + k > 5\}$

$L_2 = \{a^n b^m a^k : n > 5, m > 3, k \leq m\}$

حل: فقط زبان L_1 منظم است. برای اثبات منظم بودن زبان L_1، کافی است آن را به حالت هایی مثل $m = 0, k = 0, n > 5$ تقسیم کرد. در اینصورت می توان برای هر یک عبارات منظمی ارائه داد. در ادامه منظم نبودن L_2 را به کمک لم تزریق نشان می دهیم. ■

چند نکته:

۱- این مساله که یک زبان به مقدار نامحدود حافظه نیاز دارد، الزاما به معنی نامنظم بودن آن نیست. زبان‌هایی وجود دارند که تعداد نامحدودی موقعیت دارند ولی منظم هستند. مانند:

$L = \{w:$ w دارای تعداد مساوی از زیر رشته های 01 و 10 می‌باشد. $\}$

۲- اگر L منظم باشد، آنگاه $L-\{\lambda\}$ منظم است.

۳- اگر L منظم باشد، آنگاه $L \bigcup \{a\}$ منظم است.(به ازای تمام $a \in \sum$)

۴- هر زبان متناهی، منظم است.

بسته بودن زبان های منظم

در حالت کلی یک مجموعه نسبت به عملی بسته است اگر اجرای آن عمل روی اعضای آن مجموعه،عضوی از همان مجموعه را نتیجه بدهد. مسئله بسته بودن یک زبان منظم روی یک عملگر، یعنی آیا زبان حاصل در اثر اعمال عملگر، باز هم منظم خواهد بود؟

اگر L زبان منظمی باشد، آنگاه $\overline{L}$ و L^* و L^R منظم خواهند بود. یعنی خانواده زبان های منظم تحت مکمل گیری ، بستار ستاره ای و معکوس، بسته هستند.

اگر L_1 و L_2 منظم باشند، آنگاه $L_1 \bigcup L_2$ ، $L_1 \bigcap L_2$ ، $L_1.L_2$ ، $L_1 - L_2$ منظم خواهند بود. یعنی خانواده زبان های منظم تحت اجتماع، اشتراک، الحاق و تفاضل بسته هستند.

مثال

نشان دهید خانواده زبان‌های منظم تحت تفاضل بسته هستند.

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

مثال

با فرض اینکه L_1, L_2 زبان‌های منظم هستند، کدام یک از زبان‌های $L_3 = L_1 \cap L_2^R$ و $L_4 = \overline{L_1} \cap \overline{L_2}$ منظم است؟

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

مثال

با فرض اینکه زبان‌های L_1, L_2 روی الفبای Σ تعریف شده باشند، آنگاه کدام گزاره درست است؟

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

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

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

حل: از آنجا که ϕ و Σ ، زبان‌های منظم هستند، دلیلی نادرستی گزاره ها عبارتند از:

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

ب: با فرض $L_1 = \Sigma$، آنگاه $L_1 \cup L_2 = \Sigma$ است و L_2 می‌تواند هر نوع زبانی باشد.

پ : با فرض $L_1 = \phi$، آنگاه $L_1 - L_2 = \phi$ است و L_2 می‌تواند هر نوع زبانی باشد.■

مثال

آیا اجتماع دو زبان نامنظم، می‌تواند منظم باشد؟

حل: بله- به طور نمونه، زبان های $L_1 = \{a^n b^m : n \le m\}$ و $L_2 = \{a^n b^m : n > m\}$ هر دو نامنظم هستند، اما اجتماع آنها یعنی $L(a^* b^*)$، منظم است.■

اگر L_1 و L_2 زبان های نامنظمی باشند، آنگاه $L_1 \cup L_2$ ممکن است منظم باشد.

مثال

با فرض $L_i = a^i b^i$، آیا حاصل اجتماع نامتناهی L_i به ازای $(i = 0,1,2,3,\ldots)$، منظم است؟

حل: خیر- اجتماع این زبان ها برابر $L=\{a^nb^n : n\geq 0\}$ است که منظم نیست. (بـه ازای هـر i ، L_i متنـاهی و در نتیجـه منظم است.) ■

مجموعه زبان‌های منظم، نسبت به اجتماع نامتناهی بسته نیستند.

مثال

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

حل: خیر- می‌دانیم که زبان‌های منظم تحت مکمل گیری بسته هستند. بنابراین اگر L منظم باشد، باید مکمل آن هم منظم باشد. اما چون $\overline{L}=\{w : n_a(w)=n_b(w)\}$، منظم نیست، بنابراین L نیز منظم نمی‌باشد. ■

مثال

اگر L_1 و L_2 زبان های منظمی باشند، آیا $L=\{w : w\in L_1, w^R\in L_2\}$ لزوما منظم است؟

حل: زبان L را می توان به صورت $L=\{w : w\in L_1, w\in {L_2}^R\}$ نوشـت کـه همـان $L_1\bigcap {L_2}^R$ اسـت. بنـابراین چـون زبان‌های منظم تحت معکوس و اشتراک بسته هستند،زبان $L_1\bigcap {L_2}^R$ منظم است.■

مثال

آیا زبان $L=\{w_1cw_2 : w,w_2\in\{a,b\}^*, w_1\neq w_2\}$ منظم است؟

حل: فرض کنیم L منظم باشد، آنگاه متمم آن نیز منظم است. از آنجا که زبان‌های منظم تحت اشتراک بسته هسـتند، پـس $\overline{L}\bigcap\{a^*ca^*\}$ باید منظم باشد. اما حاصل این اشتراک زبان $\{a^nca^n\}$ است که منظم نیست. پـس فـرض خلـف نادرسـت است و زبان L منظم نیست.■

لم تزریق

یک زبان در صورتی منظم است که در جریان پردازش رشته ها، اطلاعاتی که باید در هر مرحله به خاطر آورده شـوند، کـاملا محدود باشد. این گفته درست است، اما باید قبل از هر نوع استفاده ای به دقت اثبات شود. یکـی از راه هـای اثبـات نـامنظم بودن یک زبان، استفاده از لم تزریق(پمپاژ) است.

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

به طور خلاصه:

تمامی رشته‌های طولانی L را می‌توان به سه بخش تقسیم نمود، بطوریکه تعداد دلخواهی از تکرارهای بخش میانی، رشته دیگری را در L ایجاد کند. (می گوییم رشته میانی تزریق شده است.)

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

- لم pumping یک شرط کافی برای منظم نبودن یک زبان ارائه می دهد.

مثال

آیا زبان $L = \{a^n b^n : n > 0\}$ منظم است؟

حل: فرض کنیم که L منظم باشد. حال رشته $w = a^n b^n$ متعلق به L را به ۳ قسمت تجزیه می‌کنیم:

$x = a^{n-1}, y = a, z = b^n$

حال باید به ازای تمام $i \geq 0$ ، رشته $xy^i z$، متعلق به L باشد، ولی مثلا به ازای i=2 این چنین نیست: $a^{n-1}a^2b^n = a^{n+1}b^n \notin L$

بنابراین زبان داده شده منظم نیست.■

مثال

آیا زبان $L = \{a^n b^m c^k : k = n + m\}$ منظم است؟

حل: فرض کنیم که L منظم باشد. حال رشته $w = a^n b^{n+1} c^{2n+1}$ متعلق به L را به ۳ قسمت تجزیه می‌کنیم:

$x = a^{n-1}$ و $y = a$ و $z = b^{n+1}c^{2n+1}$

حال باید به ازای تمام $i \geq 0$ ، رشته $xy^i z$، متعلق به L باشد، ولی مثلا به ازای i=2 این چنین نیست:

$a^{n-1}a^2b^{n+1}c^{2n+1} = a^{n+1}b^{n+1}c^{2n+1} \notin L$

بنابراین زبان داده شده منظم نیست. چون: $(n+1)+(n+1) \neq 2n+1$ ■

مثال

آیا زبان $L=\{a^n : n \geq 2,$ عددی اول است $n\}$ منظم است؟

حل: فرض کنیم که L منظم باشد. حال رشته $w=a^k$ با فرض اول بودن k، متعلق به L را به ۳ قسمت تجزیه می‌کنیم:

$x=a^{k-t-h}$ و $y=a^t$ و $z=a^h$

حال باید به ازای تمام $i \geq 0$ ، رشته xy^iz ، متعلق به L باشد، ولی مثلا به ازای i=k+1 این چنین نیست: $a^{k-t-h}a^{t(k+1)}a^h=a^ka^{tk}=a^{(1+t)k} \notin L$

مقدار (1+t)k اول نیست، چون به k و 1+t بخش پذیر است. بنابراین زبان داده شده منظم نیست.■

تمرین‌های فصل ۱

۱- رشته‌های تولید شده توسط عبارت منظم $(a+bb)^*(b+aa)^*$ با طول ۳ یا کمتر، را بنویسید.

۲- در الفبای $\Sigma=\{a,b\}$، درستی تساوی های زیر را بررسی کنید.

الف- $(a+b)^*=a^*(ba)^*$

ب- $(a+b)^*=a^*+b^*$

۳- در الفبای $\Sigma=\{0,1\}$، درستی تساوی های زیر را بررسی کنید.

الف- $1+0(0+10)^*11=(00^*1)1^*$

ب- $0(0+10)^*11=(0^*10)^*1$

پ- $1^*0^*+(0+1)^*01(0+1)^*=0^*1^*$

۴- هر یک از عبارات منظم زیر در الفبای $\Sigma=\{0,1\}$، چه نوع رشته هایی را تولید می کنند؟

الف- $(0+1)^*(0+11)+1+\lambda$

ب- $0^*(1+1000^*)^*(\lambda+10)$

پ- $(1+01)^*(0+\lambda)$

۵- هر یک از عبارات منظم زیر، در الفبای $\Sigma=\{a,b\}$، چه نوع رشته هایی را تولید می کنند؟

الف- $(a+ba^*b)^*ba^*$

ب- $(b+ab+aab)^*(a+aa+\lambda)$

۶- عبارت منظم $1^*(000)100$، نشان دهنده چه مجموعه ای است؟

۷- آیا دو عبارت منظم زیر برابرند؟

$r_1=b^*(\lambda+ab^*ab^*)(\lambda+ab^*)$

$r_2 = (\lambda + b^* a)(\lambda + b^* ab^* a)b^*$

۸- عبارت منظمی بنویسید که زبان $L = \{a^n b^{3m} c^{2k} : n \geq 1, m \geq 1, k \geq 1\}$ را توصیف کند.

۹- عبارت منظمی بنویسید که زبان $L = \{a^n b^m : n \geq 2, m \leq 3\}$ را توصیف کند.

۱۰- اگر زبان L به وسیله عبارت منظم $ba(b^*(ab)^* + a^*(bb)^*)ba$ توصیف شده باشد، وارون L را مشخص کنید.

۱۱- منظم بودن زبان‌های زیر را بررسی کنید.

الف- $\{a^n b^n (a+b)^* : n \geq 0\}$

ب- $\{a^* a^n b^n b^* : n \geq 0\}$

پ- $\{a^n b^m : n, m \geq 0\} - \{a^n b^m : n \neq m\}$

ت- $\{a^n b^m : n \geq 0, m \geq 0\} \cup \{a^n : n \geq 0\}$

ث- $\{w : n_a(w) \neq n_b(w)\} \cap \{a^* b^*\}$

۱۲- اگر L زبان نامنظم روی الفبای Σ باشد، آیا $(\Sigma^* - L)$ منظم است؟

۱۳- اگر L_1 و L_2 زبان های نامنظم روی الفبای Σ باشند، آیا $L_1 \cup L_2$ می تواند منظم باشد؟

۱۴- با توجه به زبان های زیر، عبارت $(L_1 \cup L_2)L_3^*$ معرف چه زبانی است؟

$L_1 = \{a, b\}$

$L_2 = \{aa, ab, ba, bb\}$

$L_3 = \{aaa, aab, aba, abb, baa, bab, bba, bbb\}$

۱۵- تابع $f : N \rightarrow N$ به صورت زیر می باشد. آیا زبان $L_f = \{0^n 1^{f(n)} : n \in N\}$ ، منظم است؟ ($L_f \subseteq \{0,1\}^*$)

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

۱۶- آیا به ازای هر زبان منظم A ، زبان نامنظم B وجود دارد به گونه‌ای که $B \subseteq A$ ؟

۱۷- آیا به ازای هر زبان منظم A ، زبان نامنظم B وجود دارد به گونه‌ای که $A \subseteq B$ ؟

۱۸- اگر L_1 و L_2 زبان هایی منظم باشند و $L_1 \subseteq L \subseteq L_2$. آیا L منظم است؟

۱۹- آیا زبان منظم، می تواند ذاتا مبهم باشد؟

۲۰- لم تزریق برای منظم نبودن یک زبان شرط کافی است یا لازم؟

۲۱- نشان دهید $(L_1.L_2)^* \subset L_2^*$.

$$L_1 = \{a^n : n = 9k, k = 0,1,2,\ldots\}$$
$$L_2 = \{a^n : n = 3k, k = 0,1,2,\ldots\}$$

۲۲- اگر $L - \sum^* = \varphi$ باشد، آنگاه L چه زبانی می تواند باشد؟

۲۳- اگر $L_1 = \{10,1\}$ و $L_2 = \{011,11\}$ باشد، حاصل $L_1.L_2$ را مشخص کنید.

۲۴- حاصل تقسیم راست L_1 / L_2 را تعیین کنید.

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

۲۵- اگر L یک زبان روی مجموعه Σ باشد، نمایشی برای $((L^R)^n)^R$ بنویسید.

۲۶- آیا اگر $s \in palindrome$ باشد، آنگاه $s^n \in palindrome$ است؟

$$palindrome = \{\lambda \ \ and \ \ all \ strings \ \ s \ \ such \ that \ reverse(s) = s\}$$

۲۷- درستی یا نادرستی گزاره های زیر را مشخص کنید.

الف- اگر L یک زبان منظم فاقد λ باشد، آنگاه truncate(L) منظم خواهد بود.

ب - اگر L منظم باشد، $chop2(L) = \{w : vw \in L, |v| = 2\}$ هم منظم خواهد بود.

پ- اگر L منظم باشد، آنگاه $even(L) = \{even(w) : w \in L\}$ هم منظم خواهد بود.

۲۸- عبارت منظمی برای زبان $L = \{w : n_a(w) \bmod 3 > 0\}$ روی الفبای $\Sigma = \{a,b\}$ بنویسید.

۲۹- عبارت منظمی روی الفبای $\Sigma = \{a,b,c,d\}$ بنویسید که بیانگر تمام رشته هایی باشد کـه طـول تمـام دنبالـه‌های a آنها، مضرب ۳ باشد.

۳۰- عبارت منظمی روی الفبای $\Sigma = \{a,b\}$ بنویسید که بیانگر تمام رشته هایی باشد که شامل دنباله ای از a ها بـا طـول بیشتر از ۲ نباشد.

۳۱- عبارت منظمی برای زبان $L = \{a^n b^m : n \geq 1, m \geq 1, nm \geq 3\}$ بنویسید.

۳۲- عبارت منظمی روی الفبای $\Sigma = \{a,b,c\}$ بنویسید که بیانگر تمام رشته هایی باشد که حداکثر دو تا a دارند.

۳۳- به کمک لم تزریق نشان دهید زبان $L = \{a^n b^p : n \leq p\}$ منظم نیست؟

۳۴- به کمک لم تزریق نشان دهید که زبان $L = \{n_a(w) = n_b(w)\}$ منظم نیست؟

حل تمرین فصل ۱

۱- رشته های تولید شده عبارتند از:

$\lambda, a, bb, ab, aaa, bbb, b, aa, aab, abb, bba$

۲- هر دو گزاره نادرست هستند:

الف: رشته b توسط سمت چپ تولید می شود اما توسط سمت راست تولید نمی شود.

ب: رشته aba توسط سمت چپ تولید می شود اما توسط سمت راست تولید نمی شود.

۳- هر سه گزاره نادرست هستند:

الف: رشته 1 توسط سمت چپ تولید می شود اما توسط سمت راست تولید نمی شود.

ب: رشته 011 توسط سمت چپ تولید می شود اما توسط سمت راست تولید نمی شود.

پ: رشته 10 توسط سمت چپ تولید می شود اما توسط سمت راست تولید نمی شود.

۴- حل هر گزاره در زیر آورده شده است:

الف: همه رشته هایی که به 01 ختم نمی‌شوند.

ب: تمام رشته هایی که شامل 101 نباشد.

پ: دارای یک زوج صفر متوالی نباشند.

۵- حل هر گزاره در زیر آورده شده است:

الف: دارای تعدادی فرد از کاراکتر b می‌باشند.

ب: همه رشته هایی که شامل سه تا a پشت سرهم نباشند.

۶- عبارت داده شده نشان دهنده مجموعه $\{<8^n+1>_2 : n \geq 1\}$ می باشد. به طور نمونه:

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

۷- دو عبارت منظم داده شده برابرند:

$$r_1 = b^*+b^*ab^*+b^*ab^*ab^*+b^*ab^*ab^*ab^*$$
$$r_2 = b^*+b^*ab^*ab^*+b^*ab^*+b^*ab^*ab^*ab^*$$

۸- با توجه به زبان داریم:

الف- حداقل یک a باید داشت.

ب- حداقل تعداد b ها باید سه یا مضربی از سه باشد.

ج- حداقل تعداد c ها باید دو یا مضربی از دو باشد.

بنابراین عبارت منظمی که زبان L را توصیف کند برابر است با: $a^+(bbb)^+(cc)^+$

۹- جواب $aaa^*(\lambda+b+bb+bbb)$ است.

۱۰- برای بدست آوردن وارون، عبارت را از انتها به ابتدا می‌خوانیم:

$$L=ba(b^*(ab)^*+a^*(bb)^*)ba \Rightarrow L^R=ab((bb)^*a^*+(ba)^*b^*)ab$$

۱۱- بررسی گزاره ها:

الف: بله- از الحاق a^nb^n با $(a+b)^*$، زبان $(a+b)^*$ حاصل می شود که منظم است.

ب: بله- عبارت a^*a^n برابر a^* و عبارت b^nb^* برابر b^* می باشد. پس زبان a^*b^* بوده که منظم است.

پ: خیر- حاصل $\{a^nb^n : n \geq 0\}$ است که منظم نمی‌باشد.

ت : بله- زبان به صورت $a^*b^*+a^*=a^*b^*$ است، که منظم است.

ث: خیر- این زبان برابر $\{a^n b^m : n \neq m\}$ است که منظم نیست، چون باید نابرابری a ها و b ها چک شود.

۱۲- خیر- چون اگر L نامنظم باشد، آنگاه مکمل آن یعنی $\Sigma^* - L$ حتما نامنظم است.

۱۳- بله- زبان $L_1 = \{a^n b^k : n \neq k\}$ نامنظم و زبان $L_2 = \{a^n b^k : n,k \geq 0\}$ منظم است، اما اجتماع آنها یعنی $\{a^n b^k : n,k \geq 0\}$، منظم است.

مثالی دیگر: زبان $L_1 = \{a^n b^n : n \geq 0\}$ نامنظم و در نتیجه مکمل آن یعنی $L_2 = \overline{L_1}$ هم نامنظم خواهد بود. اما اجتماع آن ها یعنی $a^* b^*$ ، منظم است.

۱۴- با توجه به زبان‌های داده شده، می‌توان نوشت:

$$L_1 = \Sigma, L_2 = \Sigma^2, L_3 = \Sigma^3$$

از آنجا که عبارت $(\sum \cup \sum^2)(\sum^3)^*$ معرف مجموعه رشته‌هایی است که طول آنها مضرب ۳ نمی‌باشند، می توان عبارت داده شده را به صورت زیر بیان کرد:

$$L = \{w : w \in \{a.b\}^*, |w| \bmod 3 \geq 1\}$$

۱۵- خیر- تابع f(n) یک به یک است، پس زبان منظم نیست.

۱۶- خیر- فرض کنید A برابر تهی باشد که زبانی منظم است. هیچ زبانی نمی توان پیدا کرد که زیر مجموعه تهی باشد.

۱۷- خیر- فرض کنید A برابر Σ^* باشد که زبانی منظم است. هیچ زبانی نمی توان پیدا کرد که Σ^*، زیر مجموعه آن باشد.

۱۸- خیر- چون با فرض $L_1 = \phi$ و $L_2 = \Sigma^*$، برای زبان L چه منظم باشد و چه نامنظم، رابطه $L_1 \subseteq L \subseteq L_2$ برقرار است.

۱۹- خیر- زبان های منظم نمی‌توانند ذاتاً مبهم باشند.

۲۰- لم تزریق، یک شرط کافی برای منظم نبودن یک زبان است. چون اگر زبانی در لم تزریق صدق نکند، آنگاه حتما نامنظم است.

۲۱- زبان‌های L_1 و L_2 را می توان به صورت زیر نوشت:

$$L_1 = \{\lambda, a^9, a^{18}, ...\}$$

$$L_2 = \{\lambda, a^3, a^6, a^9, ...\}$$

نتیجه الحاق این دو زبان برابر است با:

$$L_1.L_2 = \{\lambda, a^3, a^6, a^9, a^{12}, a^{15}, a^{18}, ...\}$$

بنابراین : $(L_1.L_2)^* \subset L_2^*$

۲۲- عبارت $L - \Sigma^*$ را می توان به صورت $L \cap \phi$ نشان داد:

$$L - \Sigma^* = L \cap (\overline{\Sigma^*}) = L \cap \phi$$

که طبق صورت سئوال این مقدار برابر تهی است. از آنجا که اشتراک هر زبانی با زبان تهی، زبانی تهی می باشد، بنابراین زبان L می تواند هر زبانی باشد.

۲۳- نتیجه الحاق این دو زبان برابر است با:

$$L_1.L_2 = \{10,1\}\{011,11\} = \{10011, 1011, 101, 111\}$$

۲۴- $\{\lambda, 0, 1, 11\}$

۲۵- از آنجا که روابط $(L^R)^R = L$ و $(L^R)^n = (L^n)^R$ برقرار می باشند، داریم:

$$((L^R)^n)^R = ((L^R)^R)^n = L^n$$

که L^n به صورت $\{w_1 w_2 ... w_n \mid w_i \in L\}$ نمایش داده می شود.

۲۶- بله- رشته پالیندروم، رشته ای است که با معکوس اش یکی باشد. بنابراین اگر s یک رشته پالیندروم باشد، s^n نیز یک رشته پالیندروم است و برعکس. مثلا $s = radar$ و $s^2 = radarradar$.

۲۷- هر سه گزاره درست هستند.

الف- زبان truncate(L) ، از حذف آخرین سمبل سمت راست هر رشته از زبان L بوجود می آید.

ب- زبان chop2(L)، از حذف دو سمبل انتهایی سمت چپ هر رشته ای از زبان L بوجود می آید.

پ- زبان even(w) ، از استخراج حروف واقع در موقعیت های زوج w بدست می آید.

۲۸- $r=(b^*ab^*ab^*ab^*)^*(a(a+\lambda)b^*ab^*)$

۲۹- $r=(aaa+b+c+d)^*$

۳۰- $r=(b^*+(a+\lambda)(a+\lambda)b)^*.(a+\lambda).(a+\lambda)$

۳۱- $r=a^+bbb^++aaa^+b^++aa^+bb^+$

۳۲- $r=(b+c)^*(a+\lambda)(b+c)^*(a+\lambda)(b+c)^*$

۳۳- فرض کنیم که L منظم باشد. حال رشته $w=a^mb^m$ متعلق به L را به ۳ قسمت تجزیه کرد:

$$w=a^{m-t-k}a^ta^kb^m \Rightarrow x=a^{m-t-k}, y=a^t, z=a^k.b^m$$

حال باید به ازای تمام $i\geq 0$ ، رشته xy^iz، متعلق به L باشد، ولی مثلا به ازای i=2 این چنین نیست: $a^{m-t-k}(a^t)^2a^kb^m=a^{m+t}b^m \notin L$

بنابراین زبان داده شده منظم نیست. $(t\geq 1)$

۳۴- فرض کنیم که L منظم باشد. حال رشته $w=a^mb^m$ متعلق به L را به ۳ قسمت تجزیه کرد:

$$w=a^{m-t-k}a^ta^kb^m \Rightarrow x=a^{m-t-k}, y=a^t, z=a^k.b^m$$

حال باید به ازای تمام $i\geq 0$ ، رشته xy^iz، متعلق به L باشد، ولی مثلا به ازای i=0 این چنین نیست: $a^{m-t-k}(a^t)^0a^kb^m=a^{m-t}b^m \notin L$

بنابراین زبان داده شده منظم نیست.

فصل ۲

گرامر- گرامر منظم

گرامر

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

V : مجموعه متناهی از اشیاء به نام متغیرها

T : مجموعه متناهی از اشیاء به نام سمبل‌های پایانی(ترمینال)

S : سمبل ویژه ای به نام متغیر شروع $(S \in V)$

P : مجموعه متناهی از قوانین

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

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

مثال

مجموعه V و T را در گرامر با قوانین، را مشخص کنید.

$$S \to Aab$$
$$A \to Aab$$
$$A \to a$$

حل: مجموعه V یا همان متغیرها برابر $V = \{S, A\}$ است. مجموعه T یا همان ترمینال‌ها برابر $T = \{a, b\}$ است. متغیرها با حروف بزرگ و ترمینال‌ها با حروف کوچک نمایش داده می‌شوند. ■

انواع گرامر

گرامرها را به چهار دسته تقسیم می شوند:

۱- منظم

گرامری که همه قواعد آن به صورت $A \to Bx \mid x$ یا $A \to xB \mid x$ باشد. ($A, B \in V$ و $x \in T^*$)

۲- مستقل از متن

گرامری که در سمت چپ کلیه قواعد آن، فقط یک متغیر باشد.

۳- حساس به متن

گرامری که تمامی قوانین آن به فرم $x \to y$ باشند که در آن x و y عضو $(V+T)^+$ باشند و $|x| \le |y|$.

۴- بدون محدودیت

هیچ شرط و محدودیتی برای قواعد تولید ندارد. تنها محدودیت این است که λ نباید در سمت چپ قواعد تولید باشد.

مثال

نوع گرامر زیر را مشخص کنید.

$1.S \to AB$
$2.A \to aAb$
$3.bB \to bbbB$
$4.aAb \to aa$
$5.B \to \lambda$

حل: گرامر داده شده از نوع بدون محدودیت است.

این گرامر مستقل از متن نیست، چون در سمت چپ همه قواعد باید فقط یـک متغیـر باشـد. در دو قـانون ۳و۴ این چنین نیست.

این گرامر چون مستقل از متن نیست، پس حتما منظم نمی باشد.

این گرامر حساس به متن نیست، چون طول سمت چپ باید از سمت راست بیشتر باشـد. در دو قـانون ۴و ۵ این چنین نیست. ■

زبان تولید شده توسط گرامر

با استفاده از گرامرها می توان بوسیله بکار بردن قوانین با ترکیب های مختلف، رشته های متعددی تولید کرد. مجموعـه ایـن رشته های پایانی، زبانی است که بوسله گرامر تولید می شود.

فرض کنیم که $G = (V, T, S, P)$ یک گرامر باشد. آنگاه مجموعه $L(G) = \{w \in T^* : S \overset{*}{\Rightarrow} w\}$، زبانی است که توسـط G تولید می شود.

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

مثال

گرامر منظم $S \to aS \mid \lambda$، زبان منظم a^* را تولید می‌کند. نحوه تولید رشته aa به کمک این گرامر:

$S \Rightarrow aS \Rightarrow aaS \Rightarrow aa$

■

مثال

گرامر منظم $S \to aS \mid a$، زبان منظم a^+ را تولید می‌کند.نحوه تولید رشته aa به کمک این گرامر:

$$S \Rightarrow aS \Rightarrow aa$$

■

مثال

گرامر منظم $S \to abS \mid a$ ، زبان منظم $(ab)^*a$ را تولید می‌کند. نحوه تولید رشته $ababa$:

$$S \Rightarrow abS \Rightarrow ababS \Rightarrow ababa$$

مثال

گرامر منظم زیر، چه زبانی را تولید می‌کند؟

$$S \to Aab$$
$$A \to Aab \mid a$$

حل: زبان منظم $a(ab)^+$ را تولید می‌کند.■

مثال

گرامر مستقل از متن $S \to aSb \mid \lambda$، چه زبانی را تولید می کند؟

حل: زبان $\{a^n b^n : n \geq 0\}$ یعنی رشته‌های شروع شونده با a که تعداد a و b در آنها برابر است تولید می‌شود. نحوه تولید رشته $aabb$ به کمک این گرامر:

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aabb$$

■

مثال

زبان گرامر مستقل از متن زیر را مشخص کنید.

$$S \to XY$$
$$X \to aXb \mid \lambda$$
$$Y \to cY \mid \lambda$$

حل: قوانین خط ۲ رشته هایی به فرم $a^n b^n$ و قوانین خط ۳ رشته هایی به فرم c^m تولید می‌کنند. در نهایت به علت وجود قانون ۱، رشته هایی به فرم $a^n b^n c^m$ تولید می شود. بنابراین :

$$L = \{a^n b^n c^m : n \geq 0, m \geq 0\}$$

نحوه تولید رشته $a^2 b^2 c$ توسط این گرامر: (n=2,m=1)

$$S \Rightarrow XY \Rightarrow aXbY \Rightarrow aaXbbY \Rightarrow aabbY \Rightarrow aabbcY \Rightarrow aabbc$$

■

مثال

گرامر حساس به متن زیر چه زبانی را تولید می‌کند؟

$$S \to abc \mid aAbc$$
$$Ab \to bA$$
$$Ac \to Bbcc$$
$$bB \to Bb$$
$$aB \to aa \mid aaA$$

حل: زبان حساس به متن $L = \{a^n b^n c^n : n \geq 1\}$ را تولید می کند. نحوه تولید رشته $a^2 b^2 c^2$:

$$S \Rightarrow aAbc \Rightarrow abAc \Rightarrow abBbcc \Rightarrow aBbbcc \Rightarrow aabbcc$$

■

گرامر منظم (regular)

گرامری که همه قواعد آن به صورت $A \to Bx \mid x$ باشد را خطی از چپ و گرامری کـه همـه قواعـد آن بـه صورت $A \to xB \mid x$ باشد را خطی از راست می گویند. ($x \in T^*$ و $A, B \in V$) . گرامری که خطی از راسـت یا خطی از چپ باشد را گرامر منظم می گویند.

مثال

زبان منظم تولید شده توسط هر یک از گرامرهای منظم را مشخص کنید.

الف	ب
$S \to aA \mid ab$ $A \to bS$	$S \to aA \mid \lambda$ $A \to bS$

حل:الف: $(ab)^+$ ب: $(ab)^*$

مثال

زبان منظم تولید شده توسط هر یک از گرامرهای منظم را مشخص کنید.

الف	ب
$S \to aA \mid a$ $A \to aA \mid bA \mid b$	$S \to aA \mid a$ $A \to aA \mid bA \mid a \mid b$

حل: الف: $a + a(a+b)^*b$ ب: $a(a+b)^*$

مثال

زبان منظم تولید شده توسط هر یک از گرامرهای منظم را مشخص کنید.

الف	ب	پ	ت
$S \to aaS \mid aA \mid bA$ $A \to bA \mid \lambda$	$S \to bS \mid aA$ $A \to bA \mid \lambda$	$S \to aS \mid aB \mid bB \mid \lambda$ $B \to bB \mid \lambda$	$S \to aS \mid aB$ $B \to bB \mid b$

حل: الف: $(aa)^*(a+b)b^*$ ب: b^*ab^* پ: a^*b^* ت: a^+b^+

مثال

زبان منظم تولید شده توسط هر یک از گرامرهای منظم را مشخص کنید.

الف	ب	پ
$S \to aaS \mid aX$ $X \to bX \mid bb$	$S \to aS \mid abX$ $X \to bX \mid \lambda$	$S \to aS \mid abS \mid \lambda$

حل: الف: $(aa)^*ab^+b$ ب: a^*abb^* پ: $(a+ab)^*$

■

مثال

زبان منظم تولید شده توسط هر یک از گرامرهای منظم را مشخص کنید.

الف	ب	پ	ت
$S \to aaS \mid abbX$ $X \to baX \mid \lambda$	$S \to aaaS \mid aX$ $X \to bbbbX \mid bbbbb$	$S \to aaS \mid aaaX$ $X \to bbX \mid b$	$S \to aaS \mid aaabbX$ $X \to baX \mid \lambda$

حل:

الف: $(aa)^*abb(ba)^*$ ب: $(aaa)^*a(bbbb)^+b$

پ: $(aa)^+a(bb)^*b$ ت: $(aa)^+abb(ba)^*$

■

مثال

گرامر منظمی برای زبان $L=\{a^nb^m : n\geq 2, m\geq 3\}$ بنویسید.

حل:

$$S \to aaA$$
$$A \to aA \mid bbbB$$
$$B \to bB \mid \lambda$$

■

مثال

در الفبای $\Sigma=\{a\}$، گرامر تولید کننده زبان $L=\{w : |w| \bmod 3 = 0\}$ را بنویسید.

حل: گرامر تولید کننده رشته هایی با حرف a که طول آنها مضرب ۳ باشد برابر است با:

$S \to aaaS \mid \lambda$

■

مثال

در الفبای $\Sigma = \{a\}$، گرامر تولید کننده زبان $L = \{w : |w| \bmod 3 = 2\}$ را بنویسید.

حل: $S \to aaaS \mid aa$ ■

مثال

گرامر تولید کننده زبان $L = \{w : |w| \bmod 3 > 0\}$ را بنویسید. $(\Sigma = \{a\})$

حل: باقیمانده تقسیم صحیح هر عدد به 3 ، برابر 0 یا 1 یا 2 است. با توجه به شرط، حالت 0 مطرح نیست. توسط $S \to S_1$ حالتی که باقیمانده برابر 1 است و توسط $S \to S_2$، حالتی که باقیمانده برابر 2 است، تولید می شود.

$S \to S_1 \mid S_2$

$S_1 \to aaaS_1 \mid a$

$S_2 \to aaaS_2 \mid aa$

■

مثال

گرامر منظمی بنویسید که زبان $L = \{a^n b^m :$ هستند هر دو زوج m و n $\}$ را تولید کند.

حل:

$S \to aaS \mid X$
$X \to bbX \mid \lambda$

■

مثال

گرامر منظمی بنویسید که زبان $L = \{a^n b^m :$ هر دو فرد هستند m و n $\}$ را تولید کند.

حل:

$S \rightarrow aaS \mid aY$
$Y \rightarrow bbY \mid b$

■

مثال

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

حل: برای اینکه جمع n و m زوج باشد، باید هر دو فرد یا هر دو زوج باشند. این دو حالت در مثالهای قبل بررسی شدند و کافی است با دستور $S \rightarrow S_1 \mid S_2$ ، آنها را با هم ترکیب کرد.

$S \rightarrow S_1 \mid S_2$
$S_1 \rightarrow aaS_1 \mid X$
$X \rightarrow bbX \mid \lambda$
$S_2 \rightarrow aaS_2 \mid aY$
$Y \rightarrow bbY \mid b$

■

تمرین‌های فصل۲

۱- نحوه تولید رشته $w = acaacabbdebde$ توسط گرامر زیر را نشان دهید.

$S \rightarrow acBdeA \mid BAB$

$B \rightarrow aSb \mid ae \mid \lambda$

$A \rightarrow aAb \mid b \mid \lambda$

۲- نحوه تولید رشته $w = a^5$ توسط گرامر زیر را مشخص کنید.

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

۳- نحوه تولید رشته $w = a^6$ توسط گرامر زیر را مشخص کنید.

$S \rightarrow ABC$

$B \rightarrow DBE \mid \lambda$

$DE \to EFaa$

$aE \to Ea$

$aC \to Ca$

$FE \to EFa$

$FC \to C$

$AE \to A$

$AC \to \lambda$

۴- نحوه تولید رشته $(((:a)),a)$ توسط گرامر زیر را مشخص کنید.

$S \to F \mid (S,SH)$

$F \to ((:F)) \mid a$

$H \to ,SH \mid \lambda$

۵- نحوه تولید رشته $babba$ توسط گرامر زیر را مشخص کنید.

$S \to aA \mid Aa$
$A \to aA \mid bA \mid \lambda$

۶- کدام گرامرمنظم می‌باشند؟

$G1: S \to aSb \mid ab$

$G2: S \to Ab \mid ab$

$G3: S \to aA \mid ab \ , \qquad A \to Sa$

۷- عبارت منظمی معادل با هر یک از گرامرهای زیر بنویسید.

الف	ب	پ

$S \to bS \mid aA \mid \lambda$ $A \to bA \mid aB$ $B \to bB \mid aS$	$S \to aB \mid cB$ $B \to abB \mid cbB \mid acB \mid \lambda$	$S \to aA \mid bA$ $A \to aA \mid bA \mid \lambda$

۸- عبارت منظمی معادل با هر یک از گرامرهای زیر بنویسید.

الف	ب	پ
$S \to 00S \mid X$ $X \to 11X \mid \lambda$	$S \to 0B \mid A$ $A \to 1A \mid S$ $B \to 1S \mid 1$	$S \to A$ $A \to 0B \mid 0$ $B \to 1C$ $C \to 1A$

۹- عبارت منظمی معادل با هر یک از گرامرهای زیر بنویسید.

الف	ب	پ
$S \to bS \mid aA \mid \lambda$ $A \to bA \mid aB$ $B \to bB \mid aS$	$S \to aaS \mid abbX$ $X \to baX \mid bbY$ $Y \to aY \mid \lambda$	$S \to T \mid D$ $T \to aT \mid bD$ $D \to bD \mid aT \mid \lambda$

۱۰- آیا هر گرامر خطی، گرامر منظم است؟

۱۱- یک گرامر خطی راست برای زبان $(aab^*ab)^*$ بنویسید.

پاسخ تمرین فصل ۲

۱- نحوه تولید:

$$
\begin{aligned}
S &\Rightarrow acBdeA\\
&\Rightarrow acaSbdeA\\
&\Rightarrow acaacBdeAbdeA\\
&\Rightarrow acaacaSbdeAbdeA\\
&\Rightarrow acaacaBABbdeAbdeA\\
&\Rightarrow acaacaAbdeAbdeA\\
&\Rightarrow acaacabbdebde
\end{aligned}
$$

۲- اگر اشتقاق زیر را دنبال کنیم به $aaaaa$ می رسیم:

$$S \Rightarrow ACaB \Rightarrow AaaCB \Rightarrow AaaDB \Rightarrow AaDaB \Rightarrow ADaaB \Rightarrow ACaaB$$

$$\Rightarrow AaaCaB \Rightarrow AaaaaCB \Rightarrow AaaaaE \Rightarrow AaaaEa \Rightarrow AaaEaa \Rightarrow AaEaaa$$
$$\Rightarrow AEaaaa \Rightarrow aaaaa$$

۳- نحوه تولید:

$$
\begin{aligned}
&S \Rightarrow ABC \Rightarrow ADBEC \Rightarrow ADDBEEC \Rightarrow ADDEEC\\
&\Rightarrow ADEFaaEC \Rightarrow AEFaaFaaEC \Rightarrow AFaaFaaEC\\
&\Rightarrow AFaaFaEaC \Rightarrow AFaaFEaaC \Rightarrow AFaaEFaaaC\\
&\Rightarrow AFaEaFaaaC \Rightarrow AFEaaFaaaC \Rightarrow AEFaaaFaaaC\\
&\Rightarrow AFaaaFaaaC \Rightarrow AFaaaFaaCa \Rightarrow AFaaaFaCaa\\
&\Rightarrow AFaaaFCaaa \Rightarrow AFaaaCaaa \Rightarrow AFaaCaaaa\\
&\Rightarrow AFaCaaaaa \Rightarrow AFCaaaaaa \Rightarrow ACaaaaaa\\
&\Rightarrow aaaaaa
\end{aligned}
$$

زبان گرامر به صورت $\{a^{n(n+1)} : n \geq 0\}$ است.

۴- نحوه تولید:

۵- نحوه تولید:
$$S \Rightarrow (S,SH) \Rightarrow (F,SH) \Rightarrow (((:F)),SH) \Rightarrow (((:F)),FH) \Rightarrow (((:a)),FH) \Rightarrow (((:a)),a)$$

$$S \Rightarrow Aa \Rightarrow bAa \Rightarrow baAa \Rightarrow babAa \Rightarrow babbAa \Rightarrow babba$$

۶- گرامر G2 منظم است. در گرامر منظم، همه قواعد خطی راست یا خطی چپ می باشند.

۷- عبارت منظم هر یک از گرامرها برابرند با:

الف- $(b^*ab^*ab^*a)^*b^*$

ب- $(a+c)(ab+cb+ac)*$

پ- $(a+b)(a+b)*$

۸- عبارت منظم هر یک از گرامرها برابرند با:

الف- $(00)^*(11)^*$

ب- $1^*(01^+)^+$

پ- $(011)^*0$

۹- عبارت منظم هر یک از گرامرها برابرند با:

الف- $(a+b)^2(a+b)^+$

ب- $(aa)^*abb(ba)^*bba^*$

پ- $(a^*b+\lambda)b^*(aa^*bb^*)^*$

۱۰- خیر- گرامری منظم است که همه قواعد آن، خطی چپ یا خطی راست باشند. در گرامر خطی، ممکن است تعدادی از قواعد خطی چپ و تعدادی خطی راست باشند.

۱۱- گرامر :

$S \rightarrow aaA \mid \lambda$
$A \rightarrow bA \mid abS$

فصل ۳

اتوماتای متناهی

ماشین‌ها(اتوماتا) ابزارهایی هستند برای تشخیص رشته‌های زبان که رشته را از چپ به راست بررسی کرده و نهایتاً اعلام می‌کنند که آیا رشته متعلق به زبان هست یا نه. ماشین‌ها را می‌توان به عنوان مدلهای ریاضی برای کامپیوترهای واقعی در نظر گرفت.

انواع ماشین

یک دسته بندی برای اتوماتا به صورت زیر است:

۱- متناهی(FA)

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

۲- پشته‌ای (PDA)

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

۳- کراندار خطی (LBA)

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

۴- تورینگ (TM)

ماشینی دارای حافظه نامحدود با قابلیت خواندن و نوشتن است.

نکات:

۱- ماشین متناهی، قادر به پذیرش زبان منظم است.

۲- ماشین پشته ای، قادر به پذیرش زبان مستقل از متن است.

۳- ماشین کراندار خطی، قادر به پذیرش زبان حساس به متن است.

۴- ماشین تورینگ تشخیص دهنده، قادر به پذیرش زبان بازگشتی شمارش پذیر است.

در این فصل ماشین متناهی را بررسی می کنیم. ماشین های دیگر در فصل های بعدی بررسی خواهند شد.

ماشین های متناهی

یکی از ساده ترین انواع ماشین‌ها، ماشین متناهی می باشد که از آن در شناخت زبان‌های منظم اسـتفاده می‌شـود. اتوماتـای متناهی مدل مناسبی برای کامپیوتر با محدودیت شـدید حافظـه اسـت. ماشـین‌های متنـاهی(FA) ، بـه دو دسـته معـین (DFA) و نامعین (NFA) تقسیم می‌شوند.

DFA : Deterministic Finite Accepter

NFA : Nondeterministic Finite Accepter

پذیرنده متناهی معین (DFA)

ماشین متناهی قطعی(معین) بوسیله پنج تایی $M = (Q, \sum, \delta, q_0, F)$ تعریف می شود که در آن:

Q : مجموعه متناهی از حالات داخلی

$\sum$: مجموعه متناهی از علائمی به نام الفبای ورودی

δ : تابع انتقال $(Q \times \sum \to Q)$

q_0 : حالت شروع

F : مجموعه حالات پایانی

یک dfa ، مانند همه اتوماتاها دارای حالت های داخلی، قوانینی برای انتقال از یک حالت بـه حالـت دیگـر، تعـدادی ورودی و همچنین روشهایی برای تصمیم گیری هستند.

نحوه عملکرد ماشین:

اتوماتا ابتدا در حالت شروع و هد خواندن از نوار ورودی، روی آخرین سمبل از سمت چپ رشته ورودی قرار دارد. با هر یک از حرکت های اتوماتا ، هد یک موقعیت به راست می رود. اگر با رسیدن به پایان رشته، اتوماتا در یکی از حالت های پایانی قـرار داشته باشد، رشته پذیرفته می شود.

انتقال از یک حالت داخلی به حالت داخلی دیگر، توسط تابع انتقال انجام می شود. بعنوان مثال $\delta(q_i, a) = q_j$، یعنی اگـر dfa در حالت q_i باشد و هد بر روی a باشد، آنگاه ماشین به حالت q_j تغییر حالت می دهد.

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

مثال

نمودار تغییر وضعیت DFA زیر را به کمک شکل نشان دهید.

$\sum=\{a,b\}\quad , Q=\{q_0,q_1\}\ ,\ \ F=\{q_1\}$

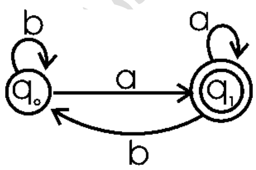
$\delta(q_0,a)=q_1$

$\delta(q_0,b)=q_0$

$\delta(q_1,b)=q_0$

$\delta(q_1,a)=q_1$

حل: این ماشین دارای دو حالت q_0 و q_1 است که q_0 حالت شروع و q_1 حالت پایانی است. طبق تابع انتقال‌های داده شده، ماشین اگر در حالت q_0 باشد و هد a را بخواند به حالت q_1 می رود و اگر b را بخواند در همان حالت q_0 باقی می‌ماند. همچنین اگر ماشین در حالت q_1 باشد و هد b را بخواند به حالت q_0 رفته و اگر a را بخواند در همان حالت q_1 باقی می ماند. نمودار تغییر وضعیت DFA این مثال در شکل زیر نشان داده شده است:

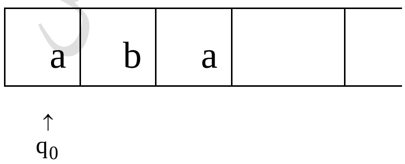


حالت نهایی به صورت دوایری دو خطی نمایش داده می شود. ■

مثال

آیا رشته W=aba توسط ماشین مثال قبل، پذیرفته می‌شود؟

حل: بله- نمایش تعلق رشته aba به زبان ماشین:



a	b	a		

$\uparrow$
q_1

a	b	a		

$\uparrow$
q_0

a	b	a		

$\uparrow$
$q_1 \in F$

نحوه پذیرش رشته توسط ماشین: با شروع از حالت q_0، ابتدا سمبل a را می خواند. با توجه به یال های گراف ماشین به حالت q_1 می رود. سپس، سمبل b خوانده شده و ماشین به حالت q_0 می رود. در نهایت سمبل a خوانده شده و ماشین به حالت q_1 می رود. در این لحظه، هم در پایان رشته و هم در حالت پایانی قرار داریم. بنابراین رشته aba پذیرفته می شود.

نشان دهید ماشین رشته abb را نمی پذیرد. ■

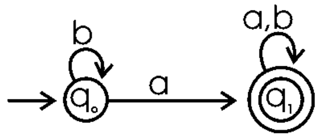
زبان ها و dfa ها

زبان مجموعه ای از تمام رشته های پذیرفته شده توسط اتومات می باشد. زبان پذیرفته شده توسط dfa ،$M=(q,\Sigma,\delta,q_0,F)$ مجموعه تمام رشته های روی Σ است که توسط M پذیرفته می شوند. فرم صوری آن به صورت $L(M)=\{w\in\Sigma^* : \delta^*(q_0,w)\in F\}$ است. dfa پس از پردازش هر یک از رشته های Σ^* ، یا آنها را می پذیرد و یا رد می کند. عدم پذیرش به این معنی است که dfa در یکی از حالت های غیر پایانی متوقف شود.

مثال

عبارت منظم مربوط به DFA زیر را مشخص کنید؟

(q_0 حالت شروع و q_1 حالت پایانی است.)

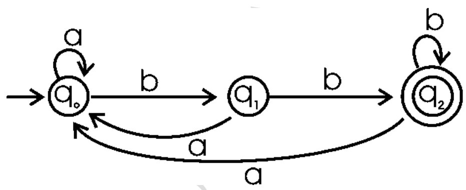


حل: $b^* a(a+b)^*$ ■

مثال

برای عبارت منظم $(a+b)^* bb$، یک DFA رسم کنید. $\Sigma = \{a,b\}$

حل:

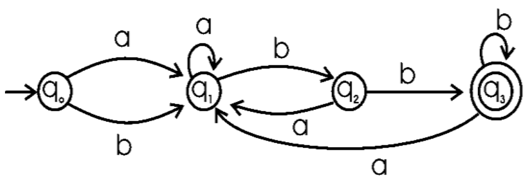


■

مثال

برای عبارت منظم $(a+b)^+ bb$، یک DFA رسم کنید. $\Sigma = \{a,b\}$

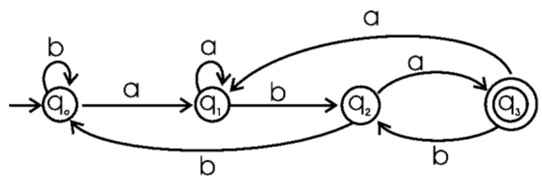
حل:



■

مثال

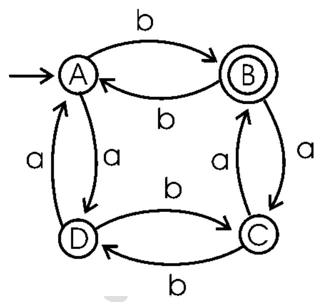
عبارت منظم مربوط به DFA زیر را مشخص کنید؟ $\Sigma = \{a,b\}$



حل: $(a+b)^* aba$ (رشته هایی که به aba ختم می شوند.)

مثال

ماشین زیر چه نوع رشته هایی را می پذیرد؟ $\Sigma = \{a,b\}$

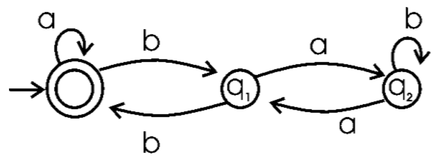


حل: رشته هایی با تعداد زوجی a و تعداد فردی b .

این مثال را با تغییر حالت پایانی به حالت های دیگر، بررسی کنید. به طور نمونه اگر C حالت پایانی بود(به جای B)، آنگاه چه زبانی را می پذیرفت؟

مثال

عبارت منظم مربوط به DFA زیر را مشخص کنید؟ $\Sigma = \{a,b\}$

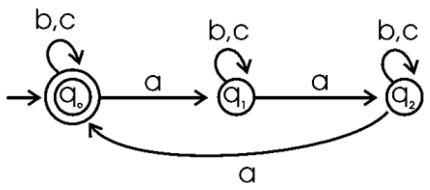


حل: $(a+b(ab^*a)^*b)^*$.

رشته λ نیز پذیرفته می‌شود، چون حالت شروع، حالت پایانی نیز می‌باشد.

مثال

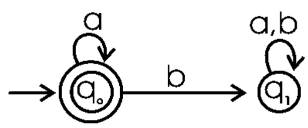
ماشین DFA زیر چه نوع رشته هایی را می‌پذیرد؟ $\Sigma = \{a,b,c\}$



حل: رشته هایی که تعداد a در آنها، مضرب 3 باشد. ■

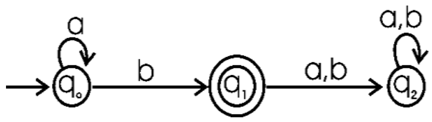
حالت دام یا تله (trap state)

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



مثال

در DFA زیر، حالت trap را مشخص کنید.

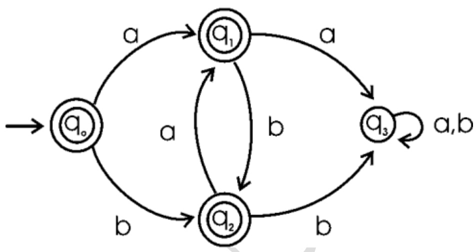


حل: حالت q_2. (زبان ماشین a^*b است.)

مثال

DFA پذیرنده زبانی که رشته های آن شامل aa یا bb نباشد را رسم کنید.

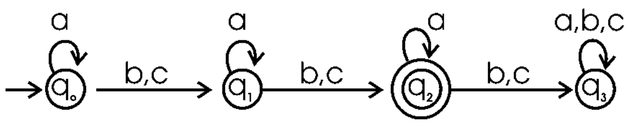
حل: حالت q_3، trap است.



مثال

در الفبای $\Sigma = \{a,b,c\}$ ، DFA پذیرنده زبانی که در رشته های آن تعداد کل b ها و c ها برابر ۲ باشد را رسم کنید.

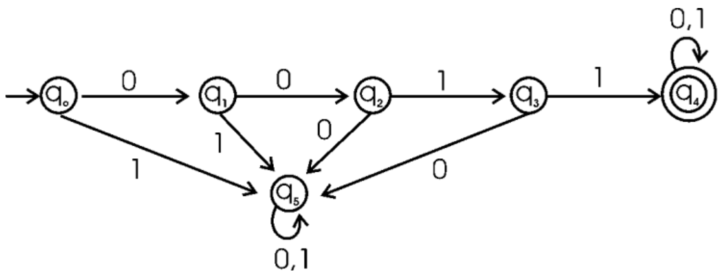
حل: حالت q_3، trap است.



مثال

در الفبای $\Sigma = \{0,1\}$، DFA پذیرنده زبانی که رشته های آن با زیر رشته 0011 آغاز شود را رسم کنید.

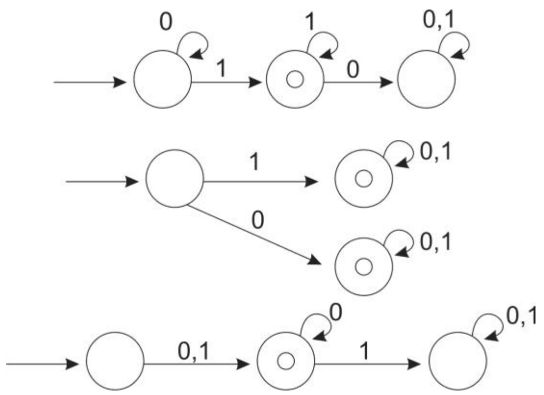
حل: حالت q_5، trap است.



مثال

برای هر یک از زبان های $(0+1)^*1$ و $1(0+1)^*$ و $(0+1)0^*$ یک DFA رسم کنید.

حل: dfa هر کدام به ترتیب از بالا به پاینن رسم شده است:



اصلاح: در شکل وسط، حالت پایینی، حالت trap است.

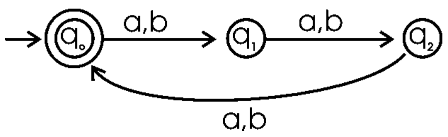
مکمل DFA

برای رسم مکمل یک DFA، کافی است که کلیه حالتهای غیر پایانی را به حالت پایانی و کلیه حالات پایـانی را بـه حـالات غیر پایانی تبدیل کنیم. (جهت یالها و مقدار برچسب آنها تغییری نمی‌کند.)

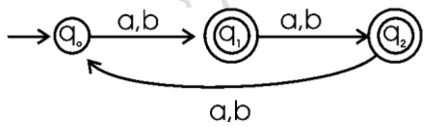
مثال

dfa ای رسم کنید که رشته‌هایی که طول آنها مضربی از 3 نباشد را بپذیرد. ($\Sigma=\{a,b\}$)

حل: ابتدا DFA ای رسم می کنیم که رشته هایی را بپذیرد که طول آنها مضربی از ۳ باشد:



سپس DFA بالا را مکمل می کنیم:

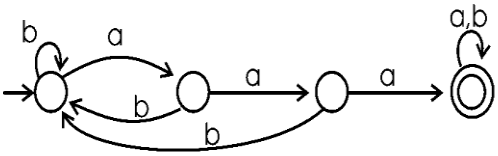


■

مثال

DFA ای رسم نمایید که زبان $L=\{w:$ در آن وجود ندارد $aaa\}$ را بپذیرد. ($\Sigma=\{a,b\}$)

حل: ابتدا DFA ای رسم می کنیم که زبان $L=\{w:$ باشد w زیر رشته‌ای از $aaa\}$ را بپذیرد:



سسپس DFA بالا را مکمل می کنیم. ■

اگر $M=(Q,\Sigma,\delta,q_0,F)$ و $\hat{M}=(Q,\Sigma,\delta,q_0,Q-F)$ دو DFA باشند، آنگاه:

$$\overline{L(M)} = L(\hat{M})$$

برای به دست آوردن مکمل یک NFA از روی نمودار حالت، بایـد ابتـدا NFA را بـه DFA تبـدیل کرده و سپس مکمل DFA را بدست آورد.

پذیرنده های متناهی نامعین(NFA)

اگر به ازای هر حالت ماشین و هر نماد ورودی به صورت منحصر بفردی حالت بعدی مشخص باشد به آن معین (Deterministic) می گویند. در یک ماشین نامعین (Nondeterministic) در هر لحظه ممکن است چندین انتخاب مختلف موجود باشد. پذیرنده‌های متناهی نامعین(غیرقطعی)، پیچیده تر از انواع معین خود هستند. ماشین نامعین می تواند به ازای دریافت یک ورودی در هر حالت، به چندین حالت مختلف تغییر حالت دهد.

تعریف

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

سه تفاوت عمده بین تعریف NFA و تعریف DFA وجود دارد. در NFA:

۱- محدوده تابع δ در مجموعه توانی 2^Q است. مثلاً اگر وضعیت فعلی q_0 باشد و حرف a خوانده شود، آنگاه هریک از حالتهای q_2,q_1 می‌تواند وضعیت بعدی باشد:

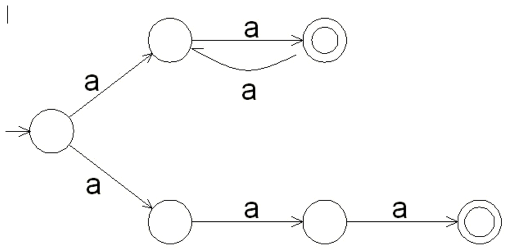
$$\delta(q_0,a)=\{q_1,q_2\}$$

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

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

مثال

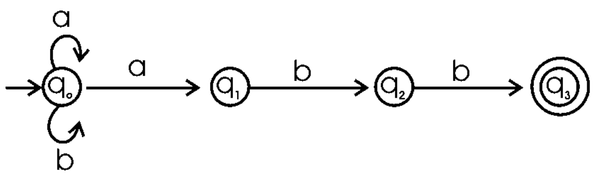
آیا ماشین زیر یک NFA می باشد؟



حل: بله- چون دو انتقال با برچسب a از حالت شروع دارد. اولین انتخاب منجر به پذیرش تمام رشته های دارای تعداد زوجی از a می شود و دومین انتخاب منجر به پذیرش رشته a^3 می شود. زبان پذیرفته شده توسط این ماشین، $L=\{a^{2n}:n\geq 1\}\bigcup\{a^3\}$ می باشد. ■

مثال

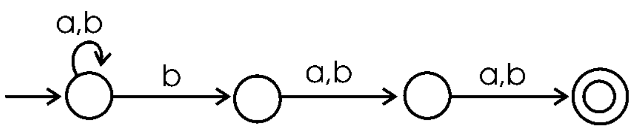
ماشین NFA زیر چه زبانی را می پذیرد؟ $\Sigma=\{a,b\}$



حل: رشته‌هایی را که به زیر رشته abb ختم شوند.(این ماشین dfa نیست. به طور نمونه، در حالت q_1، دریافت a پیش بینی نشده است یا در q_0 با دریافت a ، دو تغییر حالت ممکن است.) ■

مثال

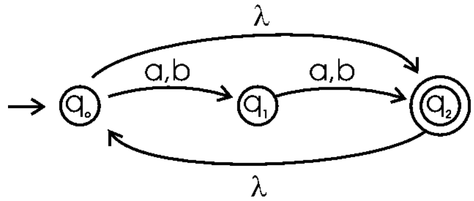
ماشین NFA زیر چه زبانی را می پذیرد؟



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

مثال

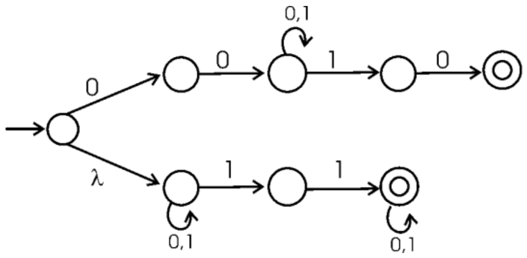
ماشین NFA زیر چه زبانی را می پذیرد؟



حل: ماشین داده شده، رشته‌هایی با طول زوج را می پذیرد. (در تعریف NFA اجازه داشتن یال با برچسب λ را داریم.به عبارتی λ بعنوان ورودی قابل قبول است. یعنی ماشین می‌تواند بدون استفاده از سمبل ورودی، دست به انتقال بزند. هد می تواند در بعضی انتقال ها حرکت نکند.) ■

مثال

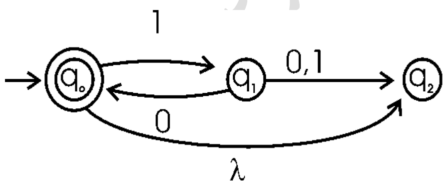
ماشین NFA زیر چه زبانی را می پذیرد؟ $\Sigma = \{0,1\}$



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

مثال

زبان پذیرفته شده توسط اتومات شکل زیر چیست؟ $\Sigma = \{0,1\}$



حل: اتومات زبان $L = \{(10)^n : n \geq 0\}$ را می پذیرد. ■

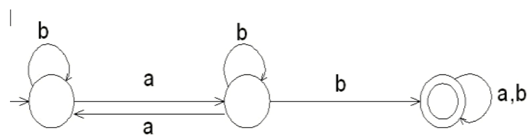
مثال

می دانیم که یک DFA داده شده با 4 حالت، رشته ای به طول 6 را می پذیرد. آیا تعداد اعضای زبان پـذیرش شـده توسط این ماشین، متناهی است؟

حل: خیر- چون طول رشته پذیرفته شده، از تعداد حالتهای ماشین بیشتر است، بنابراین دارای حلقه می باشد. ماشینی که دارای حلقه است، تعداد اعضای زبان پذیرش شده آن، نامتناهی می باشد.■

مثال

پذیرنده حالت متناهی غیر قطعی زیر چه زبانی را می‌پذیرد.



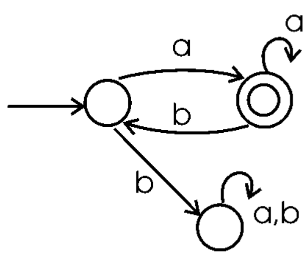
حل: $(b+ab^*a)^*ab^*b(a+b)^*$ ■

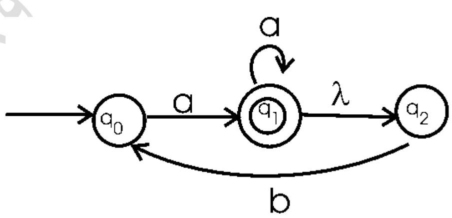
هم ارزی DFA و NFA

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

مثال

در شکل زیر دو ماشین هم ارز نشان داده شده است(شکل سمت راست dfa و سمت چپ nfa است)





چند نکته:

۱- به ازای هر زبانی که توسط یک NFA پذیرفته می شود، یک DFA هم وجود دارد که آن را می پذیرد.

۲- کلاس های DFA ها و NFA ها دارای قدرت یکسان می باشند.

۳- برای هر NFA با هر تعداد دلخواه حالت پایانی، یک DFA با فقط یک حالت پایـانی، هـم ارز بـا آن NFA وجود دارد.

۴- اگر L یک زبان غیر تهی باشد بطوریکه هر w عضو L، دارای حداقل طول n باشد، آنگاه هر DFA کـه L را بپـذیرد، باید حداقل n+1 حالت داشته باشد.

ارتباط گرامر منظم با ماشین متناهی

می توان با داشتن انتقالات یک ماشین متناهی، گرامر منظم مربوط به زبان تولید شده توسط ماشین را مشـخص کـرد(و بـر عکس). به طور نمونه انتقال $\delta(q_i, a) = q_j$ به قانون $q_i \to aq_j$ تبدیل می شود.

مثال

گرامر منظمی برای زبان $L = \{w :$ هر دو زوج هستند $n_b(w), n_a(w)\}$ روی $\{a,b\}$ که DFA آن به صورت زیر اسـت، بنویسید. (q_0 حالت شروع و پایانی است.)

$$\delta(q_0, a) = q_1, \delta(q_0, b) = q_2$$
$$\delta(q_1, a) = q_0, \delta(q_1, b) = q_3$$
$$\delta(q_2, a) = q_3, \delta(q_2, b) = q_0$$
$$\delta(q_3, a) = q_2, \delta(q_3, b) = q_1$$

حل: از روی انتقالات ماشین، گرامر را می نویسیم:

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

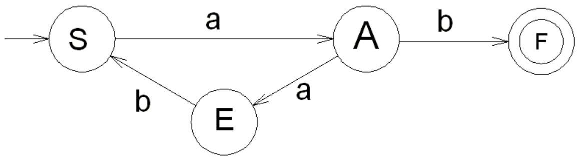
مثال

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

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

حل:

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



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

مثال

ماشین متناهی متناظر با گرامر منظم زیر را بدست آورید.

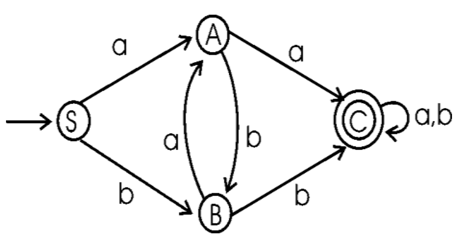
$S \to aA \mid bB$

$A \to aC \mid bB \mid a$

$C \to aC \mid bC \mid a \mid b$

$B \to aA \mid bC \mid b$

حل:



روال تبدیل NFA به DFA

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

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

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

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

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

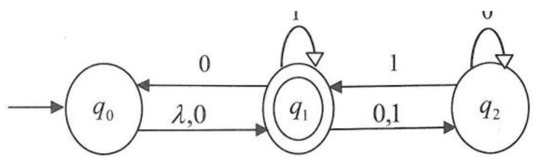
د- در صورت عدم وجود راسی با برچسب $\{q_1, q_m, ..., q_n\}$ در G_D ، این راس را ایجاد کنید. یالی از $\{q_i, q_j, ..., q_k\}$ بـه $\{q_1, q_m, ..., q_n\}$ به نام a به G_D اضافه کنید.

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

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

مثال

NFA زیر را به DFA تبدیل کنید.



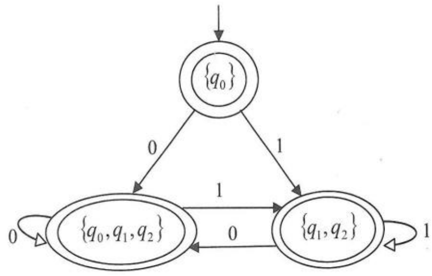
حل:

$$\delta(\{q_0\},0) = \{q_0, q_1, q_2\} \quad , \quad \delta(\{q_0\},1) = \{q_1, q_2\}$$

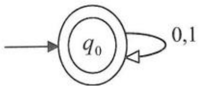
$$\delta(\{q_1, q_2\},0) = \{q_0, q_1, q_2\} \quad , \quad \delta(\{q_1, q_2\},1) = \{q_1, q_2\}$$

$$\delta(\{q_0, q_1, q_2\},0) = \{q_0, q_1, q_2\} \quad , \quad \delta(\{q_0, q_1, q_2\},1) = \{q_1, q_2\}$$

بنابراین داریم:



اما nfa حاصل λ را هم می پذیرد. پس q_0 هم باید وضعیت نهایی باشد در این صورت dfa حاضر همه رشته ها را می پذیرد و به طور خلاصه به شکل روبه رو در می آید:



■

کاهش تعداد حالات در ماشین های متناهی

هر dfa یک زبان منحصر بفرد را تعریف می کند، اما عکس این جمله صحیح نیست،یعنی برای یک زبان ممکن است چند dfa وجود داشته باشد. در عمل ممکن است از بین چند dfa که برای یک زبان وجود دارد، یکی را انتخاب کرد. معمولا این dfa دارای حالات کمتری می باشد. در dfa می توان حالتی که دسترس پذیر نباشد را حذف کرد و بعضی از حالتها را که ادغام پذیر هستند را با هم ادغام کرد.

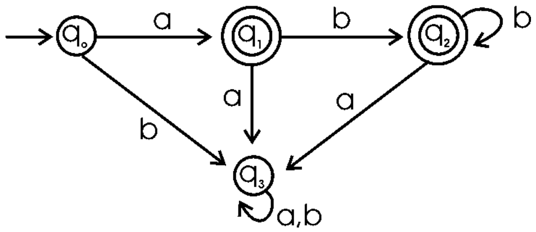
دو حالت p و q یک dfa ، در صورتی ادغام پذیر هستند که به ازای هر $w \in \Sigma^*$ داشته باشیم:

$$\delta^*(q,w) \in F \Rightarrow \delta^*(p,w) \in F \quad , \quad \delta^*(q,w) \notin F \Rightarrow \delta^*(p,w) \notin F$$

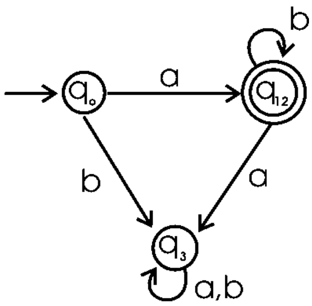
اگر p و q ادغام پذیر بوده و q و r هم ادغام پذیر باشند، آنگاه p و r ادغام پذیر هستند و در نتیجه هر سه حالت هم ادغام پذیر هستند. به عبارتی ادغام پذیر بودن یک رابطه هم ارزی است. (ادغام ناپذیر بودن، اینطور نیست.)

مثال

DFA زیر را کمینه نمایید.

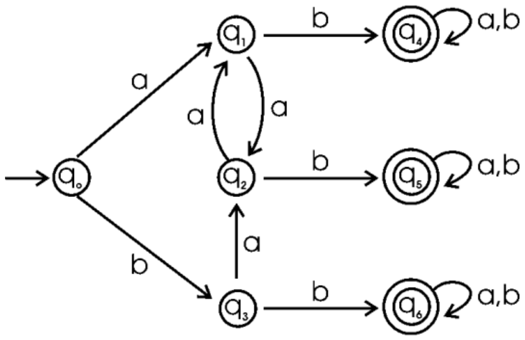


حل: حالت q_1 و q_2 ادغام پذیرند و به جای آنها حالت q_{12} را قرار می‌دهیم:

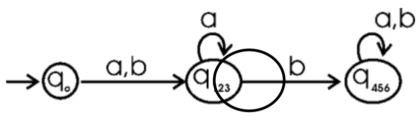


مثال

DFA زیر را کمینه نمایید.



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



■

نحوه تشخیص منظم بودن یک زبان

در فصل های قبل برای اینکه نشان دهیم یک زبان منظم است، عبارت منظم یا گرامر منظم برای آن ارائـه دادیم. یک راه دیگر این است که بتوان برای آن یک ماشین متناهی پیدا کرد.

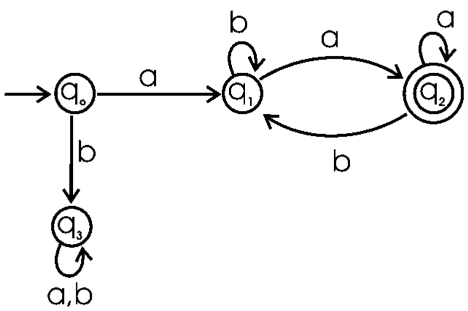
زبان L منظم است اگر و فقط اگر یک DFA مانند M وجود داشته باشد به طوریکه L = L(M).

فرض کنیم r یک عبارت منظم باشد. آنگاه یک NFA وجود دارد که L(r) را می پذیرد. در نتیجه L(r) یک زبـان منظم است.

مثال

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

حل: برای اثبات منظم بودن زبان داده شده، کافی است که یک DFA برای آن پیدا کنیم. زبان L رشته هایی را می‌پـذیرد که با حرف a شروع شده و با حرف a نیز تمام می شود. بین این دو حرف a می تواند هیچ یا تعداد نامحدودی حرف a یا b ظاهر شود.

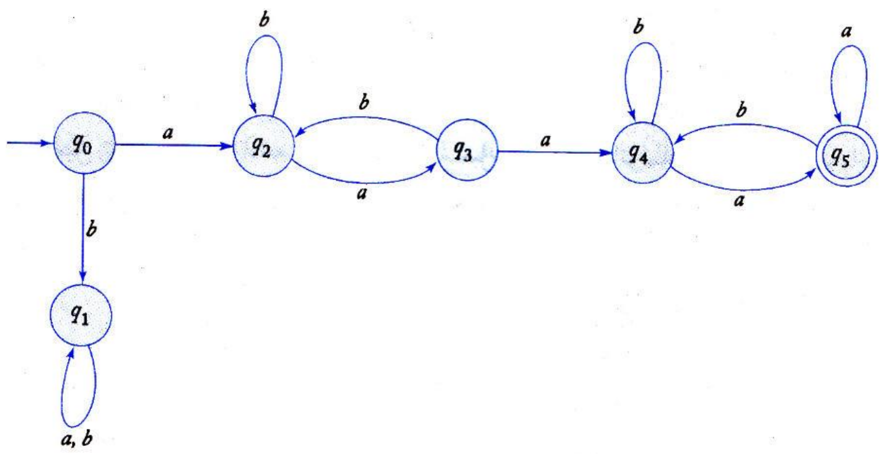


وضعیت q_3 یک تله است و اگر رشته با حرف b شروع شود، ماشین به این حالت می رود. ■

مثال

با فرض اینکه $L=\{awa : w\in\{a,b\}^*\}$، نشان دهید که زبان L^2 منظم است.

حل: زبان L^2 برابر $L^2=\{aw_1aaw_2a : w_1,w_2\in\{a,b\}^*\}$ است. با رسم یک DFA برای این زبان می توان نشان داد که L^2 نیز منظم است.

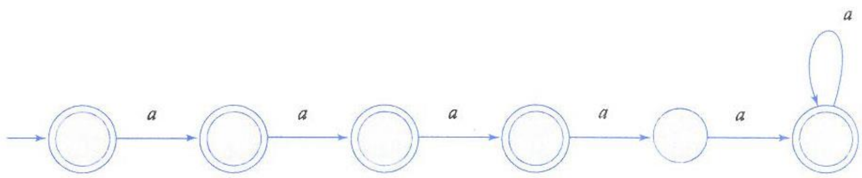


اگر زبان مفروض L منظم باشد، L^2 و L^3 و ... هم منظم خواهد بود.

مثال

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

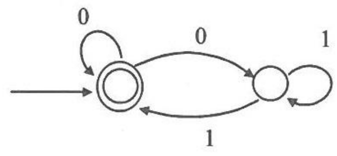
حل: برای اثبات منظم بودن زبان داده شده، کافی است که یک DFA برای آن پیدا کنیم:



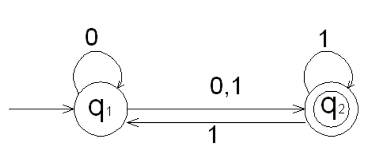
تمرین‌های فصل ۳

۱- عبارت منظمی معادل زبان هر یک از ماشین‌های زیر را بنویسید.

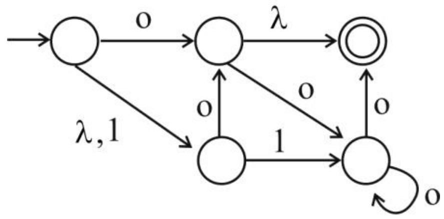
الف:



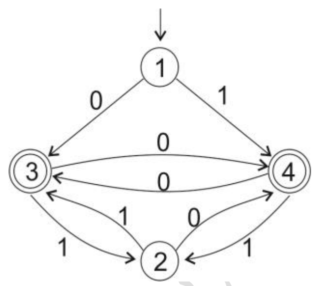
ب:



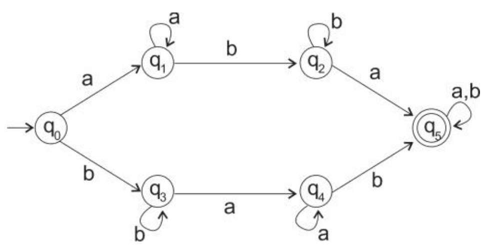
پ:



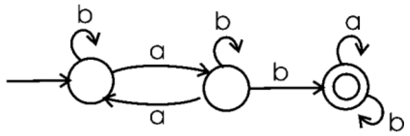
ت:



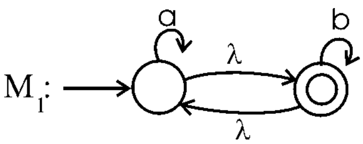
ث:

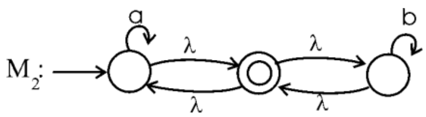


ج:

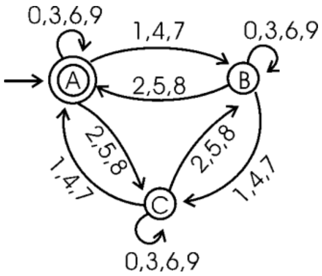


۲- چه رابطه ای بین $L(M_1)$ و $L(M_2)$ است؟



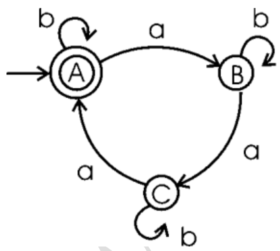


۳- چه رشته‌های توسط ماشین زیر پذیرفته می شوند؟

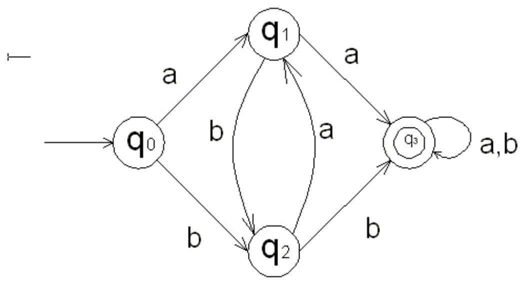


۴- یک DFA برای زبان منظم $(a^*(b+\lambda)a^*)^*$ رسم کنید.

۵- گرامر هم ارز با پذیرنده زیر را بنویسید.



۶- گرامر متناظر با ماشین زیر را بنویسید.



۷- آیا زبان $\{a^nb^ma^k : n+m+k>3\}$ منظم است؟

۸- کدام یک از زبان‌های زیر منظم هستند؟

$L_1 = \{a^n :$ n یا ضریب سه است یا ضریب پنج $\}$

$L_2 = \{a^n :$ n ضریب سه است و ضریب پنج نیست $\}$

$L_3 = \{a^n : n = 2^k, k \le 2000\}$

۹- آیا زبان زیر منظم است؟

$L=\{w\in(0,1)^*:$ تعداد 0 ها و 1ها برابر مقدار ثابت $n\geq 0$ باشد.$\}$

۱۰- آیا زبان $L=\{a^n : n=i+jk\}$ منظم است؟ (i و k ثابت و $j=0,1,2,...$)

۱۱- آیا زبان زیر منظم است؟

$L=\{w\in L(A):$ A یک DFA است و در مسیر پذیرش w از چند حالت معین A عبور نمی شود.$\}$

۱۲- آیا زبان زیر منظم است؟

$L=\{x^n y^n : x\in(0+1)^*, y\in(0+1)^*, n\geq 0\}$

۱۳- اگر $m(L)$ تعداد حالات DFA مینیمال متناظر با زبان $L\subseteq\{0,1\}^*$ باشد، $m(L)$ چه رابطه ای با $m(L^R)$ دارد؟

۱۴- آیا هر زبان منظمی، معین(قطعی) است؟

۱۵- برای $L(a\phi)$ و $L(\phi^*)$ یک NFA رسم کنید.

۱۶- ثابت کنید که تمام زبانهای متناهی، منظم هستند.

۱۷- منظور از dfa ناقص چیست؟

۱۸- کدام گزاره صحیح است؟

الف- اگر L یک زبان منظم فاقد λ باشد، یک NFA بدون انتقال λ و فقط با یک حالت پایانی وجود دارد که L را می پذیرد.

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

پ- به ازای هر NFA با چندین حالت شروع، یک NFA با دقیقا یک حالت شروع وجود دارد که همان زبان را می‌پذیرد.

۱۹- رشته‌های تولید شده توسط گرامر زیر چه خاصیتی دارند؟

$S \to aA \mid bB$

$A \to bbA \mid baB \mid aS \mid b$

$B \to abA \mid aaB \mid bS \mid a$

تمرین:

۱- یک dfa طراحی کنید که زبان پذیرفته شده توسط گرامر زیر را تولید کند.

$$S \to abA$$
$$A \to baB$$
$$B \to aA \mid bb$$

۲- در الفبای $\Sigma = \{a,b\}$ برای هر یک از موارد زیر یک dfa طراحی کنید.

الف- تمام رشته هایی که حداقل یک a و دقیقا دو تا b دارند.

ب- سمت چپ ترین علامت با سمت راست ترین علامت یکی نباشد.

پ – طول هر دنباله از a ها برابر ۲ یا ۳ باشد.

۳- در الفبای $\Sigma = \{a,b\}$ برای هر یک از زبان‌های زیر یک ماشین متناهی طراحی کنید.

الف- $L = \{a^3\} \bigcup \{a^{2n} : n \ge 1\}$

ب- $L = \{aba^n : n \ge 0\} \bigcup \{abab^n : n \ge 0\}$

پ- $L = \{a^n b : n \ge 0\} \bigcup \{b^n a : n \ge 1\}$

ت- $L = \{a^n b^m : n \ge 2, m \le 2\}$

۴- در الفبای $\Sigma = \{a,b\}$ برای هر یک از زبان‌های زیر یک ماشین متناهی طراحی کنید.

الف- $L = \{ab^3wb^2 : w \in \{a,b\}^*\}$

ب- $L = \{w : w = b^n a + ub : u \in \Sigma^+, n_a(u) \ne 0, n \ge 0\}$

پ- $L = \{w : n_a(w) \bmod 3 > n_b(w) \bmod 3\}$

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

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

پاسخ تمرین فصل ۳

۱- عبارت منظم معادل با هر ماشین برابر است با:

الف- $(0+01^*1)^*$

ب- $0^*(0+1)1^*(10^*(0+1)1^*)^*$

پ- $0+(00+1+11)0^*0$

ت- $(0+1)(0+10+11)^*$

ث- $(a^+b^+a+b^+a^+b)(a+b)^*$

ج- $(b+ab^*a)^*ab^*b(a+b)^*$

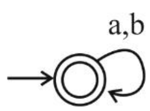
۲- هر دو ماشین، زبان $(a+b)^*$ را می پذیرند.

۳- ماشین اعداد بخش پذیر بر ۳ را می‌پذیرد. مانند ۹۷۵ ، ۶۴۷۱ ، ۳۱۵

۴- زبان داده شده را می توان ساده کرد:

$$(a^*(b+\lambda)a^*)^* = ((a^*b+a^*)a^*)^* = (a^*ba^*+a^*)^* = (a+b)^*$$

بنابراین ماشین به صورت زیر است:



۵- گرامر به صورت زیر است:

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

$B \rightarrow bB \mid aC$

$C \rightarrow bC \mid aA$

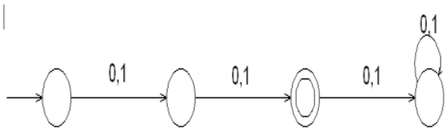
۶- گرامر به صورت زیر است:

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

۷- بله- چون می توان برای آن یک NFA رسم کرد.

۸- همه زبان های داده شده منظم هستند،چون برای هر یک از آنها می توان یک DFA رسم کرد.

۹- بله- چون مقدار n ثابت است و می توان برای آن یک DFA طراحی کرد. به طور مثال اگر n=2، به کمک DFA زیر می توان رشته های زبان را پذیرفت:



۱۰- بله– می توان dfa برای آن رسم کرد. چون i و k ثابت هستند، بنابراین تعداد وضعیت ها مشخص و عددی ثابت و متناهی می باشد.($a^{i+jk} = a^i . a^{jk}$)

۱۱- بله- چون برای آن یک DFA طراحی شده است.

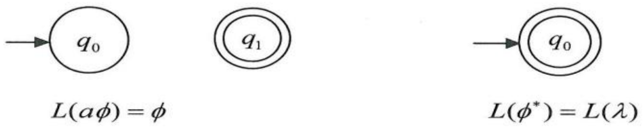
۱۲- بله- چند رشته از زبان:

$$L = \{\lambda, 0^5 1^5, 1^2 0^2, (011)^2 (10)^2, (10)^3 (0)^3, \ldots\}$$

۱۳- اگر ماشین DFA ای که زبان L را می پذیرد، به ماشینی که معکوس زبان L (یعنی L^R) را بپذیرد، تبدیل کنیم، ممکن است ماشین به NFA تبدیل شود. بنابراین برای قطعی کردن ماشین، تعداد حالات یعنی $m(L)$ به صورت توانی زیاد می شود. بنابراین $m(L) \le 2^{m(L^R)}$.

۱۴- بله- چون می توان برای آن یک DFA رسم کرد.

۱۵- ماشین های NFA به صورت زیر است:

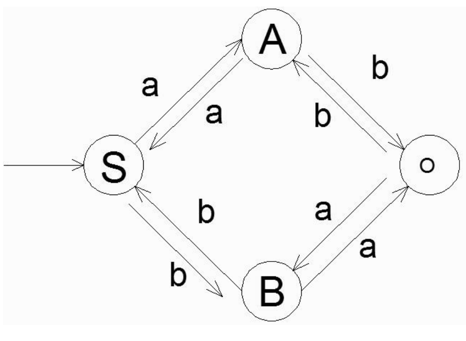


۱۶- کافی است یک NFA برای آنها طراحی کنیم. یک q_0 مشترک برای همه رشته های عضو وجود دارد و سپس هر رشته به صورت مجزا برای خود یک وضعیت نهایی دارد که مسیر رسیدن با این وضعیت نهایی معادل خود رشته خواهد بود.

۱۷- nfa ای که در آن هیچ انتقال λ وجود ندارد و به ازای هر $q \in Q$ و هر $a \in \Sigma$، $\delta(q,a)$ حاوی حداکثر یک عضو باشد را dfa ناقص می گویند. در این dfa ، برای برخی انتقال ها نمی توان حرکتی کرد.

۱۸- همه گزاره ها صحیح هستند.

۱۹- شامل تعداد فردی a و تعداد فردی b هستند. البته می توان FA را رسم کرد و با توجه به آن زبان را تشخیص داد.)



منتخبی از عناوین آموزشی منتشر شده بر روی فرادرس

برنامه‌نویسی	
عنوان آموزش	مدت زمان تقریبی
مبانی برنامه نویسی - کلیک کنید (+)	۳ ساعت
برنامه نویسی C - کلیک کنید (+)	۱۳ ساعت
آموزش برنامه نویسیC++	۲۰ ساعت
برنامه نویسی کاربردی سی شارپ - کلیک کنید (+)	۱۴ ساعت
آموزش جامع شی گرایی در سی شارپ - کلیک کنید (+)	۱۴ ساعت
برنامه نویسی جاوا - کلیک کنید (+)	۲۳ ساعت
آموزش برنامه نویسی PHP - کلیک کنید (+)	۲۸ ساعت
آموزش فریمورک PHP کدایگنایتر (CodeIgniter) - کلیک کنید (+)	۷ ساعت
آموزش اسکریپت برنامه نویسی jQuery - کلیک کنید (+)	۷ ساعت
آموزش ویژوال بیسیک دات نت (Visual Basic.NET) - کلیک کنید (+)	۱۳ ساعت
آموزش تکمیلی ویژوال بیسیک دات نت (Visual Basic.NET) - کلیک کنید (+)	۱۶ ساعت
آموزش برنامه نویسی با روش سه لایه به زبان VB.Net - کلیک کنید (+)	۴ ساعت
برنامه نویسی اسمال بیسیک یا Small Basic - کلیک کنید (+)	۱۶ ساعت
آموزش ساخت بازی ساده در ویژوال بیسیک - کلیک کنید (+)	۲ ساعت
آموزش کاربردی SQL Server - کلیک کنید (+)	۱۱ ساعت
آموزش آشنایی با LINQ to SQL در C# - کلیک کنید (+)	۲ ساعت
آموزش برنامه نویسی با روش سه لایه به زبان سی شارپ - کلیک کنید (+)	۴ ساعت
آموزش برنامه نویسی تحت شبکه با سی شارپ در قالب پروژه - کلیک کنید (+)	۱ ساعت
آموزش Cryptography در دات نت - کلیک کنید (+)	۳ ساعت
برنامه‌نویسی (ادامه از صفحه قبل)	

مدت زمان تقریبی	عنوان آموزش
۴ ساعت	آموزش قفل نرم افزاری در سی شارپ از طریق رجیستری
۱۳ ساعت	آموزش ساخت اپلیکیشن کتاب و کار با دادهها در اندروید - کلیک کنید (+)
۱۴ ساعت	آموزش ارتباط با دیتابیس سمت سرور در اندروید - کلیک کنید (+)
۱۶ ساعت	آموزش ساخت روبات و کنترل آن با اندروید - کلیک کنید (+)
۷ ساعت	آموزش ساخت اپلیکیشن دیکشنری صوتی دو زبانه با قابلیت تشخیص صوت کاربر - کلیک کنید (+)
۹ ساعت	آموزش مدیریت بانک اطلاعاتی اوراکل - کلیک کنید (+)
۷ ساعت	آموزش مدیریت بانک اطلاعاتی اوراکل پیشرفته - کلیک کنید (+)
۱ ساعت	آموزش راه اندازی اوراکل ۱۲c در لینوکس
۳ ساعت	آموزش دیتاگارد در اوراکل - کلیک کنید (+)
۹ ساعت	برنامه نویسی متلب - کلیک کنید (+)
۱۴ ساعت	متلب برای علوم و مهندسی - کلیک کنید (+)
۷ ساعت	برنامه نویسی متلب پیشرفته - کلیک کنید (+)
۸ ساعت	طراحی رابط های گرافیکی (GUI) در متلب - کلیک کنید (+)
۷ ساعت	آموزش برنامه نویسی R و نرمافزار RStudio - کلیک کنید (+)
۵ ساعت	آموزش تکمیلی برنامه نویسی R و نرمافزار RStudio - کلیک کنید(+)
۲۰ ساعت	آموزش برنامه نویسی پایتون ۱ - کلیک کنید (+)
۵ ساعت	آموزش برنامه نویسی پایتون ۲ - کلیک کنید (+)
۱۶ ساعت	آموزش گرافیک کامپیوتری با OpenGL - کلیک کنید(+)

راه اندازی و مدیریت وبسایت‌ها و سرورها	
عنوان فرادرس	**مدت زمان تقریبی**
آموزش برنامه نویسی PHP - کلیک کنید (+)	۲۸ ساعت
آموزش فریمورک PHP کدایگنایتر (CodeIgniter) - کلیک کنید (+)	۷ ساعت
آموزش طراحی وب با HTML - کلیک کنید (+)	۳ ساعت
آموزش طراحی وب با CSS - کلیک کنید (+)	۵ ساعت
آموزش پروژه محور HTML و CSS - کلیک کنید – کلیک کنید (+)	۴ ساعت
آموزش جاوا اسکریپت (JavaScript) - کلیک کنید (+)	۹ ساعت
آموزش کار با cPanel - کلیک کنید (+)	۱ ساعت
آموزش مدیریت هاست با DirectAdmin - کلیک کنید – کلیک کنید (+)	۱ ساعت
راه اندازی سایت و کار با وردپرس - کلیک کنید (+)	۷ ساعت
راه اندازی فروشگاه دیجیتال با وردپرس و Easy Digital Downloads - کلیک کنید (+)	۱ ساعت
آموزش راه اندازی سایت شخصی با وردپرس - کلیک کنید (+)	۱ ساعت
آموزش ترجمه قالب وردپرس – کلیک کنید (+)	۲ ساعت
آموزش راه اندازی سایت خبری با وردپرس - کلیک کنید (+)	۲ ساعت

علوم کامپیوتر	
عنوان آموزش	**مدت زمان تقریبی**
ساختمان داده‌ها -کلیک کنید (+)	۱۰ ساعت
آموزش ساختمان داده‌ها (مرور – تست کنکور ارشد) - کلیک کنید (+)	۲۰ ساعت
آموزش نظریه زبان‌ها و ماشین‌ها -کلیک کنید (+)	۹ ساعت
آموزش نظریه زبان‌ها و ماشین (مرور – تست کنکور ارشد) - کلیک کنید (+)	۸ ساعت
آموزش سیستم های عامل - کلیک کنید (+)	۱۱ ساعت
آموزش سیستم عامل (مرور اجمالی و تست کنکور) - کلیک کنید (+)	۱۲ ساعت
آموزش پایگاه داده‌ها -کلیک کنید (+)	۸ ساعت
آموزش پایگاه داده‌ها (مرور – تست کنکور ارشد) - کلیک کنید (+)	۵ ساعت
آموزش طراحی و پیاده سازی زبان های برنامه سازی - کلیک کنید (+)	۱۰ ساعت
آموزش طراحی و پیاده سازی زبان های برنامه سازی (مرور – تست کنکور ارشد) - کلیک کنید (+)	۱۲ ساعت
آموزش روش های حل روابط بازگشتی - کلیک کنید (+)	۴ ساعت
آموزش روش تقسیم و حل در طراحی الگوریتم - کلیک کنید (+)	۲ ساعت
آموزش ذخیره و بازیابی اطلاعات - کلیک کنید (+)	۸ ساعت
آموزش ساختمان گسسته با رویکرد حل مساله - کلیک کنید (+)	۱۶ ساعت
آموزش جامع مدارهای منطقی - کلیک کنید (+)	۱۰ ساعت
آموزش معماری کامپیوتر با رویکرد حل مسأله - کلیک کنید (+)	۲۰ ساعت
آموزش ساختمان گسسته (مرور و حل تست های کنکور کارشناسی ارشد) - کلیک کنید (+)	۱۲ ساعت
آموزش طراحی الگوریتم - کلیک کنید (+)	۸ ساعت

آموزش شبکه های کامپیوتری ۱ - کلیک کنید (+)	۱۹ ساعت

علوم کامپیوتر (ادامه از صفحه قبل)	
عنوان آموزش	**مدت زمان تقریبی**
آموزش نظریه گراف و کاربردها - کلیک کنید (+)	۱۴ ساعت
آموزش نتورک پلاس - (Network+) کلیک کنید (+)	۱۰ ساعت
آموزش مدل سازی UML با نرم‌افزار Rational Rose - کلیک کنید (+)	۳ ساعت
آموزش پردازش ویدئو - کلیک کنید (+)	۳ ساعت
پردازش تصویر در متلب - کلیک کنید (+)	۱۶ ساعت
آموزش پردازش تصویر با OpenCV - کلیک کنید (+)	۱۰ ساعت

هوش مصنوعی	
عنوان آموزش	**مدت زمان تقریبی**
الگوریتم ژنتیک در متلب - کلیک کنید (+)	۱۴ ساعت
الگوریتم PSO در متلب - کلیک کنید (+)	۱۰ ساعت
الگوریتم ازدحام ذرات (PSO) گسسته باینری - کلیک کنید (+)	۲ ساعت
ترکیب الگوریتم ژنتیک و PSO در متلب - کلیک کنید (+)	۱ ساعت
حل مسأله فروشنده دوره گرد با استفاده از الگوریتم ژنتیک - کلیک کنید (+)	۲ ساعت
الگوریتم مورچگان در متلب - کلیک کنید (+)	۶ ساعت
الگوریتم رقابت استعماری در متلب - کلیک کنید (+)	۱۳ ساعت
طراحی سیستم های فازی عصبی یا ANFIS با استفاده از الگوریتم های فرا ابتکاری و تکاملی - کلیک کنید (+)	۲ ساعت
الگوریتم فرهنگی یا Cultural Algorithm در متلب - کلیک کنید (+)	۲ ساعت

هوش مصنوعی (ادامه از صفحه قبل)	
عنوان آموزش	**مدت زمان تقریبی**
شبیه سازی تبرید یا Simulated Annealing در متلب - کلیک کنید (+)	۴ ساعت
جستجوی ممنوع یا Tabu Search در متلب - کلیک کنید (+)	۲ ساعت
الگوریتم کرم شب تاب یا Firefly Algorithm در متلب - کلیک کنید (+)	۱ ساعت
بهینه سازی مبتنی بر جغرافیای زیستی یا BBO در متلب - کلیک کنید (+)	۲ ساعت
جستجوی هارمونی یا Harmony Search در متلب - کلیک کنید (+)	۲ ساعت
کلونی زنبور مصنوعی یا Artificial Bee Colony در متلب - کلیک کنید (+)	۳ ساعت
الگوریتم زنبورها یا Bees Algorithm در متلب - کلیک کنید (+)	۲ ساعت
الگوریتم تکامل تفاضلی - کلیک کنید (+)	۱ ساعت
الگوریتم بهینه سازی علف هرز مهاجم یا IWO در متلب - کلیک کنید (+)	۲ ساعت
الگوریتم بهینه سازی مبتنی بر و یادگیری یا - TLBO کلیک کنید (+)	۱ ساعت
الگوریتم بهینه سازی جهش قورباغه یا SFLA در متلب - کلیک کنید (+)	۴ ساعت
بهینه سازی چند هدفه در متلب - کلیک کنید (+)	۱۹ ساعت
بهینه سازی مقید در متلب - کلیک کنید (+)	۹ ساعت
شبکه های عصبی مصنوعی در متلب - کلیک کنید (+)	۲۸ ساعت
آموزش کاربردی شبکه های عصبی مصنوعی - کلیک کنید (+)	۹ ساعت
آموزش استفاده از شبکه عصبی مصنوعی با نروسولوشن - کلیک کنید (+)	۳ ساعت
شبکه عصبی GMDH در متلب - کلیک کنید (+)	۴ ساعت
شبکه های عصبی گازی به همراه پیاده سازی عملی در متلب - کلیک کنید (+)	۳ ساعت
طبقه بندی و بازشناسی الگو با شبکه های عصبی LVQ در متلب - کلیک کنید (+)	۳ ساعت
آموزش پیاده سازی الگوریتم های تکاملی و فراابتکاری در سی شارپ - کلیک کنید (+)	۸ ساعت

آمار و داده کاوی	
عنوان آموزش	**مدت زمان تقریبی**
گنجینه فرادرس های یادگیری ماشین و داده کاوی - کلیک کنید (+)	۸۸ ساعت
گنجینه فرادرس های محاسبات هوشمند - کلیک کنید (+)	۷۱ ساعت
آموزش یادگیری ماشین – کلیک کنید (+)	۲۴ ساعت
داده کاوی یا Data Mining در متلب - کلیک کنید (+)	۲۴ ساعت
آموزش داده کاوی در - RapidMiner کلیک کنید (+)	۲ ساعت
آموزش وب کاوی - کلیک کنید (+)	۱۷ ساعت
شبکه های عصبی مصنوعی در متلب - کلیک کنید (+)	۲۸ ساعت
آموزش کاربردی شبکه های عصبی مصنوعی - کلیک کنید (+)	۹ ساعت
شبکه عصبی GMDH در متلب - کلیک کنید (+)	۴ ساعت
شبکه های عصبی گازی به همراه پیاده سازی عملی در متلب - کلیک کنید (+)	۳ ساعت
طبقه بندی و بازشناسی الگو با شبکه های عصبی LVQ در متلب - کلیک کنید (+)	۳ ساعت
خوشه بندی با استفاده از الگوریتم های تکاملی و فراابتکاری - کلیک کنید (+)	۳ ساعت
تخمین خطای کلاسیفایر یا - Classifier کلیک کنید (+)	۲ ساعت
انتخاب ویژگی یا - Feature Selection کلیک کنید (+)	۲ ساعت
انتخاب ویژگی با استفاده از الگوریتم های فرا ابتکاری و تکاملی - کلیک کنید (+)	۴ ساعت
کاهش تعداد رنگ تصاویر با استفاده از روش های خوشه بندی هوشمند - کلیک کنید (+)	۱ ساعت
آموزش پردازش سیگنال های واقعی در متلب - کلیک کنید (+)	۴ ساعت
مبانی و کاربردهای راهبرد تلفیق داده یا - Data Fusion کلیک کنید (+)	۹ ساعت
آمار و احتمال مهندسی - کلیک کنید (+)	۱۳ ساعت
آزمون‌های فرض مربوط به میانگین جامعه نرمال در - SPSS کلیک کنید (+)	۳ ساعت

آمار و داده کاوی (ادامه از صفحه قبل)	
عنوان آموزش	مدت زمان تقریبی
آموزش محاسبات آماری در اکسل - کلیک کنید (+)	۲ ساعت
آموزش کنترل کیفیت آماری - کلیک کنید (+)	۵ ساعت
آموزش کنترل کیفیت آماری با - SPSS کلیک کنید (+)	۲ ساعت
آموزش مدل سازی معادلات ساختاری با - Amos کلیک کنید (+)	۷ ساعت
تجزیه و تحلیل اطلاعات با نرم‌افزار - SASکلیک کنید (+)	۴ ساعت

مهندسی برق	
عنوان آموزش	مدت زمان تقریبی
طراحی دیجیتال با استفاده از وریلوگ یا - Verilog کلیک کنید (+)	۷ ساعت
آموزش جامع مدارهای منطقی - کلیک کنید (+)	۱۰ ساعت
آموزش مروری طراحی و پیاده سازی مدارات منطقی - کلیک کنید (+)	۴ ساعت
آموزش میکروکنترلر AVR و نرم‌افزار - CodevisionAVRکلیک کنید (+)	۴ ساعت
آموزش تکمیلی میکروکنترلر AVR و نرم‌افزار - CodevisionAVRکلیک کنید (+)	۴ ساعت
آشنایی باPLC های ساخت شرکت های Omron و - Keyence کلیک کنید (+)	۶ ساعت
میکروکنترلر PIC با کامپایلر - CCS کلیک کنید (+)	۹ ساعت
آموزش تحلیل و طراحی مدارات الکترونیکی با - Proteus کلیک کنید (+)	۳ ساعت
آموزش شبیه سازی و تحلیل مدارهای الکتریکی و الکترونیکی با پی اسپایس (PSpice) -کلیک کنید (+)	۳ ساعت
آموزش مقدماتی - ADS کلیک کنید (+)	۳ ساعت

آموزش تکمیلی آنالیز مدار با نرم‌افزار - ADSکلیک کنید (+)	۲ ساعت
مهندسی برق (ادامه از صفحه قبل)	
عنوان آموزش	**مدت زمان تقریبی**
آموزش تحلیل ریاضی مدارات الکتریکی با - OrCAD کلیک کنید (+)	۲ ساعت
آموزش شبیه سازی مدارات الکترونیکی با - Orcad Capture کلیک کنید (+)	۲ ساعت
آموزش برنامه نویسی آردوینو (- (Arduinoکلیک کنید (+)	۸ ساعت
آموزش تکمیلی برنامه نویسی آردوینو - (Arduino) کلیک کنید (+)	۷ ساعت
آموزش طراحی برد مدار چاپی به کمک نرم‌افزار - Altium Designer کلیک کنید (+)	۷ ساعت
آموزش مبانی ربات های برنامه پذیر - کلیک کنید (+)	۵ ساعت
آموزش ساخت روبات و کنترل آن با اندروید - کلیک کنید (+)	۱۶ ساعت
آموزش مدارهای الکتریکی ۱ - کلیک کنید (+)	۹ ساعت
آموزش مدارهای الکتریکی ۲ - کلیک کنید (+)	۱۱ ساعت
آموزش سیستم های کنترل خطی - کلیک کنید (+)	۱۰ ساعت
آموزش مکاترونیک کاربردی ۱ - کلیک کنید (+)	۱۳ ساعت
آموزش کامسول (مباحث منتخب) - کلیک کنید (+)	۳ ساعت
آموزش سینماتیک مستقیم و معکوس ربات‌ها -کلیک کنید (+)	۳ ساعت
آموزش تجزیه و تحلیل سیگنال‌ها و سیستم ها - کلیک کنید (+)	۲۷ ساعت
آموزش متلب با نگرش تحلیل آماری، تحلیل سری های زمانی و داده های مکانی - کلیک کنید (+)	۸ ساعت
پردازش سیگنال های دیجیتال با استفاده از نرم‌افزار متلب - کلیک کنید (+)	۴ ساعت
شبیه سازی سیستم با سیمیولینک - کلیک کنید (+)	۴ ساعت
آموزش سیستم های قدرت در سیمولینک و متلب - کلیک کنید (+)	۱۱ ساعت
آنالیز پایداری و کنترل سیستم های قدرت با استفاده از جعبه ابزارهای نرم‌افزار متلب - کلیک کنید (+)	۲ ساعت
آشنایی با SimPowerSystems در شبیه سازی سیستم های قدرت - کلیک کنید (+)	۳ ساعت

مهندسی برق (ادامه از صفحه قبل)	
عنوان آموزش	**مدت زمان تقریبی**
شبیه سازی ماشین های الکتریکی در تولباکس های Simulink و SimPowerSystem در نرمافزار متلب - کلیک کنید (+)	۴ ساعت
آموزش الکترونیک قدرت – شبیه سازی در متلب و سیمولینک - کلیک کنید (+)	۸ ساعت
آموزش شبیه سازی عملکرد انواع ماشین های الکتریکی در سیمولینک متلب - کلیک کنید (+)	۱۰ ساعت
برنامه های پاسخگویی بار - کلیک کنید (+)	۴ ساعت
آموزش نرمافزار ETAP برای تحلیل سیستم های قدرت - کلیک کنید (+)	۲۱ ساعت
آموزش مقدماتی نرمافزار GAMSبرای حل مسائل بازار برق - کلیک کنید (+)	۵ ساعت
آموزش پخش بار اقتصادی (دیسپاچینگ اقتصادی) در GAMS - کلیک کنید (+)	۲ ساعت
کاربرد فازی در سیستم های قدرت - کلیک کنید (+)	۲ ساعت
آموزش نرمافزار HFSS - کلیک کنید (+)	۵ ساعت
طراحی آنتن مایکرواستریپ به کمک نرمافزار HFSS - کلیک کنید (+)	۱ ساعت
آموزش طراحی و شبیه سازی آنتن های SIW با HFSS - کلیک کنید (+)	۱ ساعت
آموزش بررسی کامل آنتن های مایکرواستریپ و طراحی آن توسط CST - کلیک کنید (+)	۳ ساعت
آموزش تجزیه سیگنال به مولفه های مود ذاتی یا Empirical Mode Decomposition -کلیک کنید (+)	۴۰ دقیقه
نمونه برداری و بازسازی اطلاعات در سیستم های کنترل دیجیتال - کلیک کنید (+)	۳ ساعت
بررسی پاسخ ورودی پله در شناسایی فرآیندهای صنعتی - کلیک کنید (+)	۱ ساعت
مدل سازی و شناسایی سیستم های دینامیکی با استفاده از مدل ARX و شبکه فازی عصبی ANFIS - کلیک کنید (+)	۱ ساعت
طراحی و تنظیم ضرایب کنترل کننده PID با منطق فازی - کلیک کنید (+)	۲ ساعت
آموزش کنترل سیستم چهار تانک - کلیک کنید (+)	۲ ساعت

فصل ۴

زبان و گرامر مستقل از متن

برای ساخت برنامه‌های قدرتمندتر باید تا حدی از قید محدودیت‌های موجود در گرامرهای منظم رها شویم. از زبان‌های مستقل از متن در طراحی زبان‌های برنامه‌سازی و ساخت کامپایلر استفاده می‌شود.

گرامر مستقل از متن

گرامر مفروض $G=(V,T,S,P)$ در صورتی مستقل از متن خوانده می شود که تمام قوانین P به فرم $A \to x$ باشند که در آن $A \in V$ و $x \in (V \bigcup T)^*$. به طور کلی شرط مستقل از متن بودن این است که در سمت چپ قوانین، فقط یک متغیر وجود داشته باشد.

زبان L مستقل ازمتن نامیده می شود، اگر و تنها اگر گرامر مستقل ازمتنG وجود داشته باشد بطوریکه $L=L(G)$.

گرامرهای منظم، مستقل از متن نیز هستند. هر زبان منظمی، یک زبان مستقل از متن نیز می باشد.

خانواده زبان های منظم یکی از زیر مجموعه های محض خانواده زبان های مستقل از متن هستند.

مثال

گرامر تولید کننده زبان $L=\{a^n b^n : n \geq 1\}$ را بنویسید.

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

$$S \to aSb \mid ab$$

مثال

گرامر تولید کننده زبان $L=\{a^n b^n : n \geq 0\}$ را بنویسید.

حل: $S \to aSb \mid \lambda$

مثال

گرامر تولید کننده زبان $L=\{a^n b^{n+1}: n \ge 0\}$ را بنویسید.

حل: $S \to aSb \mid b$.

روش دوم: این زبان با زبان $\{a^n b^n : n \ge 0\}$ فقط در یک b تفاوت دارد($a^n b^n b$). بنابراین چون گرامر $\{a^n b^n : n \ge 0\}$ را با قانون $S \to aSb \mid \lambda$ می توان تولید کرد، خواهیم داشت:

$S \to Xb$

$X \to aXb \mid \lambda$

مثال

گرامر تولید کننده زبان $L=\{a^{n+1} b^n : n \ge 0\}$ را بنویسید.

حل: $S \to aSb \mid a$.

روش دوم: این زبان با $\{a^n b^n : n \ge 0\}$، فقط در یک a تفاوت دارد: $aa^n b^n$

$S \to aX$

$X \to aXb \mid \lambda$

مثال

گرامر تولید کننده زبان $L=\{a^{n+3} b^n : n \ge 0\}$ را بنویسید.

حل: $S \to aSb \mid aaa$.

روش دوم: این مثال با $\{a^n b^n : n \ge 0\}$ فقط در سه تا a تفاوت دارد: $aaaa^n b^n$

$S \to aaaX$

$X \to aXb \mid \lambda$

مثال

گرامر تولید کننده زبان $L=\{a^n b^{n-3}: n \ge 3\}$ را بنویسید.

حل:زبان را می توان به صورت $\{a^{k+3}b^k : k \geq 0\}$ نیز نشان داد. پس جواب، مانند مثال قبل است. ■

مثال

گرامر تولید کننده زبان $L = \{a^n b^{2n} : n \geq 0\}$ را بنویسید.

حل: $S \to aSbb | \lambda$

روش دوم:

$$
\begin{aligned}
&S \to aSA | \lambda \\
&A \to bB \\
&B \to b
\end{aligned}
$$

مثال

گرامر تولید کننده زبان $L = \{a^{n+2} b^{3n} : n \geq 0\}$ را بنویسید.

حل: $S \to aSbbb | aa$ ■

مثال

گرامر تولید کننده زبان $L = \{a^n b c^{2n} : n \geq 0\}$ را بنویسید.

حل: $S \to aScc | b$ ■

مثال

گرامر تولید کننده زبان $L = \{a^n b^n : n \geq 0\}$ را بنویسید. (n مضربی از ۳ نیست)

حل: $S \to aaaSbbb | aabb | ab$ ■

مثال

گرامر تولید کننده زبان $L = \{a^{2n+2} b^{n+2} : n \geq 0\}$ را بنویسید

حل:

$$
\begin{aligned}
&S \to aXb \\
&X \to aaXb | ab
\end{aligned}
$$

مثال

گرامر تولید کننده زبان $L=\{a^n bc^m : n \neq m\}$ را بنویسید.

$S \to aSc \mid aS \mid Sc \mid ab \mid bc$

مثال

گرامر تولید کننده زبان $L=\{a^n b^n a^k b^k : n \geq 0, k \geq 0\}$ را بنویسید.

حل:

$$S \to MM$$
$$M \to aMb \mid \lambda$$

■

مثال

گرامر تولید کننده زبان $L=\{a^n b^n : n \geq 0\} \cup \{b^n a^n : n \geq 0\}$ را بنویسید.

حل: کافی است با دستور $S \to S_1 \mid S_2$ ، گرامر $\{a^n b^n\}$ و گرامر $\{b^n a^n\}$ را با هم ترکیب کرد:

$$S \to S_1 \mid S_2$$
$$S_1 \to aS_1b \mid \lambda$$
$$S_2 \to bS_2a \mid \lambda$$

■

مثال

فرض کنید $L=\{a^n b^n : n \geq 0\}$ باشد. گرامری برای زبان L^* بنویسید.

حل:

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

■

مثال

گرامری برای زبان $L=\{a^n b^k : 2n \leq k \leq 3n\}$ بنویسید. $(n \geq 0, k \geq 0)$

حل: با $S \to aSbb \mid \lambda$ زبان $\{a^n b^{2n} : n \geq 0\}$ و با $S \to aSbbb \mid \lambda$ زبان $\{a^n b^{3n} : n \geq 0\}$ را تولید می‌کنیم:

$$S \to aSbb \mid aSbbb \mid \lambda$$

■

مثال

با توجه به مثال قبل، نحوه تولید رشته a^3b^7 را مشخص کنید.

حل:

$$S \Rightarrow aSbb \Rightarrow aaSbbbb \Rightarrow aaaSbbbbbbb \Rightarrow aaabbbbbbb$$

دو بار از $S \to aSbb$ و یکبار از $S \to aSbbb$ و در آخر از $S \to \lambda$ استفاده می کنیم.

■

مثال

گرامر تولید کننده زبان $L=\{a^n b^k : k > n\}$ را بنویسید. $(n \geq 0, k \geq 0)$

حل: توسط $X \to aXb$ به تعداد مساوی a و b و به کمک $B \to bB \mid b$ ، یک یا چند b تولید می شود:

$S \to XB$
$X \to aXb \mid \lambda$
$B \to bB \mid b$

■

مثال

گرامر تولید کننده زبان $L=\{a^n b^k : n>k\}$ را بنویسید. $(n \ge 0, k \ge 0)$

حل: توسط $A \to aA \mid a$ ، یک یا چند a و توسط $X \to aXb$ به تعداد مساوی a و b تولید می شود:

$S \to AX$
$A \to aA \mid a$
$X \to aXb \mid \lambda$

■

مثال

یک گرامر مستقل از متن برای زبان $L=\{a^n b^k : n \ne k\}$ بنویسید.

حل: کافی است دو حالت $n>k$ و $k>n$ که در دو مثال قبل بررسی شد را با هم ترکیب کرد.

$S \to AX \mid XB$

$X \to aXb \mid \lambda$

$A \to aA \mid a$

$B \to bB \mid b$

■

مثال

گرامری بنویسید که زبان $L=\{a^n b^n c^k : n>0, k>0\}$ را تولید کند.

حل: توسط $X \to aXb \mid ab$ زبان $\{a^n b^n : n>1\}$ و توسط $Y \to cY \mid c$ زبان $\{c^k : k>0\}$، را تولید می کنیم. با دستور $S \to XY$ این دو را کنار هم قرار می دهیم.

$S \to XY$

$X \to aXb \mid ab$

$Y \to cY \mid c$

■

مثال

گرامری بنویسید که زبان $L = \{a^k b^n c^n : n > 0, k > 0\}$ را تولید کند.

حل: توسط قانون $X \to aX \mid a$ زبان $\{a^k : k > 0\}$ و توسط $Y \to bYc \mid bc$ زبان $\{b^n c^n : n > 0\}$ را تولید می کنیم. با دستور $S \to XY$ این دو را کنار هم قرار می دهیم.

$S \to XY$

$X \to aX \mid a$

$Y \to bYc \mid bc$

مثال

گرامر مستقل از متنی برای زبان $L = \{a^n b^m c^k \ : \ m \le k\}$ بنویسید.

$(n \ge 0, m \ge 0, k \ge 0)$

حل: توسط $A \to aA \mid \lambda$ زبان a^* و توسط دیگر قوانین $\{b^m c^k : m \le k\}$، را می‌سازیم.

$S \to AX$
$A \to aA \mid \lambda$
$X \to bXc \mid Y$
$Y \to Yc \mid \lambda$

■

مثال

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

حل: می توان زبان را به صورت $a^n b^m c^m c^n$ نشان داد. قسمت $b^m c^m$ با X تولید می‌شود.

$S \rightarrow aSc \mid X$
$X \rightarrow bXc \mid \lambda$

■

مثال

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

حل: می‌توان زبان را به صورت $a^k a^m b^m c^k$ نشان داد. قسمت $a^m b^m$ با X تولید می‌شود.

$$S \to aSc \mid X$$
$$X \to aXb \mid \lambda$$

■

مثال

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

حل: می‌توان زبان را به فرم $a^n b^n b^k c^k$ نشان داد. قسمت $a^n b^n$ با X و قسمت $b^k c^k$ با Y تولید می‌شود.

$$S \to XY$$
$$X \to aXb \mid \lambda$$
$$Y \to bYc \mid \lambda$$

■

مثال

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

حل: شرط به معنی $k=n-m$ و $k=-(n-m)$ است. این دو حالت را می‌توان به ترتیب به صورت $n=k+m$ و $m=n+k$ نشان داد. این حالتها در مثالهای قبل بررسی شد و باید با هم ترکیب شوند.

■

مثال

گرامر مستقل از متنی برای زبان $L=\{a^n b^m c^k : k>n+m\}$ بنویسید. $(n\geq 0, m\geq 0, k\geq 1)$

حل: کافی است زبان را به صورت $a^n b^m c^+ c^m c^n$ نشان داد. قسمت $b^m c^+ c^m$ با X تولید می شود.

$$S \to aSc \mid X$$
$$X \to bXc \mid cY$$
$$Y \to cY \mid \lambda$$

■

مثال

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

حل: زبان را به صورت $a^n(a+b)^+ b^m c^m c^n$ نشان می‌دهیم.

$$S \to aSc \mid aS \mid aX \mid bX$$
$$X \to bXc \mid bX \mid \lambda$$

■

مثال

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

حل: شرط به معنی $k>n+m$ یا $k<n+m$ است. در نتیجه این دو حالت که در مثالهای قبل بررسی شد را با هم ترکیب می‌کنیم.

■

مثال

گرامر مستقل از متنی برای زبان $L=\{a^n b^m c^k : k=n.m\}$ بنویسید.

حل: نمی توان یک گرامر مستقل از متن برای این زبان نوشت. پس این زبان مستقل از متن نیست. ■

مثال

گرامر مستقل از متنی برای زبان $L=\{(ab)^n (cd)^n : n\geq 1\}$ بنویسید.

$$S \to aXd$$
$$X \to bSc \mid bc$$

■

مثال

گرامر مستقل از متنی برای زبان $L=\{(ab)^n (cde)^n : n\geq 1\}$ بنویسید.

$S \rightarrow aXde$
$X \rightarrow bSc \mid bc$

■

مثال

گرامر مستقل از متنی برای زبان $L = \{aa(bc)^n be(dde)^n : n \geq 0\}$ بنویسید.

$S \rightarrow aaX$
$X \rightarrow bYe$
$Y \rightarrow cXdd \mid \lambda$

■

مثال

گرامر مستقل از متنی برای زبان $L = \{ab(bbaa)^n bba(ba)^n \mid n \geq 0\}$ بنویسید.

حل: در جملات این زبان ، تعداد bbaa با ba برابر است.

$S \rightarrow abX$
$X \rightarrow bbYa$
$Y \rightarrow aaXb \mid \lambda$

■

مثال

گرامر مستقل از متنی برای زبان $L = \{a^n b^k c^k d^n : n \geq 0, k \geq 1\}$ بنویسید.

$S \rightarrow aSd \mid bXc \mid \lambda$
$X \rightarrow bXc \mid \lambda$

■

مثال

گرامر تولید کننده زبان $L = \{w : n_a(w) = n_b(w)\}$ را بنویسید.

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

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

■

مثال

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

حل: تعداد a ها یکی بیشتر از تعداد b ها می باشد.

$S \rightarrow XaX$

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

به طور نمونه رشته ababa را تولید می کند که شامل سه تا a و دو تا b است. نحوه تولید:

$S \Rightarrow XaX \Rightarrow aXbaX \Rightarrow abaX \Rightarrow ababXa \Rightarrow ababa$

■

مثال

گرامر تولید کننده زبان $L=\{w: n_a(w)>n_b(w)\}$ را بنویسید.

حل: تعداد a ها بیشتر از تعداد b ها می باشد.

$S \rightarrow SS \mid aSb \mid bSa \mid aS \mid Sa \mid a$

■

مثال

گرامر تولید کننده زبان $L=\{w: |w| \bmod 3 \geq |w| \bmod 2\}$ را بنویسید. ($\Sigma=\{a\}$)

حل: طول رشته باید 6k، 3k+1 و یا 3k+2 باشد.

$$S \rightarrow aX \mid aaX \mid Y$$
$$X \rightarrow aaaX \mid \lambda$$
$$Y \rightarrow aaaaaaY \mid \lambda$$

■

مثال

گرامر تولید کننده زبان $L=\{w\in\{a,b\}^*:ww^R\}$ را بنویسید.

حل: جمله های این زبان مانند abba است که نیمه دوم، معکوس نیمه اول است. $S\to aSa\,|\,bSb\,|\,\lambda$ ■

مثال

گرامر تولید کننده زبان $L=\{w\in\{a,b\}^+:ww^R\}$ را بنویسید.

حل: $S\to aSa\,|\,bSb\,|\,aa\,|\,bb$

■

مثال

گرامر تولید کننده زبان $\{w\in\{a,b\}^*:w=w^R\}$ را بنویسید.

حل: جمله های زبان این گرامر مانند aba است که از هر دو طرف یکسان خوانده می شوند.

$$S\to aX\,|\,bY\,|\,\lambda$$
$$X\to Sa\,|\,\lambda$$
$$Y\to Sb\,|\,\lambda$$

مثال

گرامر تولید کننده زبان $L=\{a^nww^Rb^n:n\geq 1, w\in\{a,b\}^*\}$ را بنویسید.

$$S\to aSb\,|\,aMb$$
$$M\to aMa\,|\,bMb\,|\,\lambda$$

■

مثال

گرامر تولید کننده زبان $L=\{uvwv^R:u,v,w\in\{a,b\}^+,|u|=|w|=2\}$ را بنویسید.

$$S\to XY$$
$$X\to aa\,|\,bb\,|\,ab\,|\,ba$$
$$Y\to aYa\,|\,bYb\,|\,aXa\,|\,bXb$$

■

مثال

گرامر مستقل از متن زیر چه زبانی را تولید می کند؟

$$S \rightarrow AB \mid \lambda$$
$$A \rightarrow 1A \mid S$$
$$B \rightarrow 0B \mid S$$

حل: زبان گرامر داده شده، زبان منظم $(1^*0^*)^+$ است. گرامر فوق یک گرامر مستقل از متن است که زبان منظم تولید می‌کند. ■

مثال

گرامر مستقل از متن زیر چه زبانی را تولید می کند؟

$$S \rightarrow XYZ \mid \lambda$$
$$X \rightarrow aX \mid S$$
$$Y \rightarrow bY \mid S$$
$$Z \rightarrow cZ \mid S$$

حل: زبان گرامر داده شده، زبان منظم $(a^*b^*c^*)^+$ است. گرامر فوق یک گرامر مستقل از متن است که زبان منظم تولید می‌کند.■

گرامر منظم فقط زبان منظم تولید می کند، اما گرامر مستقل از متن علاوه بر زبان مستقل از متن، می تواند زبان منظم هم تولید کند.

گرامر ساده (S–گرامر)

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

مثال

گرامر زیر یک گرامر ساده نمی باشد، چون زوج (S,a) در دو قانون ۱و۳ وجود دارد.

$$S \to aS$$
$$S \to bSS$$
$$S \to aSS$$
$$S \to c$$

■

مثال

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

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

■

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

بسیاری از ویژگی های زبان های برنامه سازی، بوسیله گرامرهای ساده قابل توصیف هستند. در کامپایلرها بیشتر از گرامرهای LL و RR استفاده می شود.

بسته بودن زبان های مستقل از متن

خانواده زبان‌های مستقل از متن تحت اجتماع، الحاق، بستار ستاره ای، معکوس و هم ریختی بسته است و تحت اشتراک، مکمل گیری و تفاضل بسته نیست.

مثال

نشان دهید که خانواده زبان‌های مستقل از متن،تحت اشتراک بسته نیستند.

حل: دو زبان L_1 و L_2 مستقل از متن هستند، چون برای آنها می توان گرامر مستقل از متن نوشت:

$L_1 = \{a^n b^n c^m : n \geq 0, m \geq 0\}$

$L_2 = \{a^n b^m c^m : n \geq 0, m \geq 0\}$

اما اشتراک این دو زبان یعنی $\{a^n b^n c^n : n \geq 0\}$ ، مستقل از متن نیست.

■

اگر L_1 مستقل از متن و L_2 یک زبان منظم باشد، آنگاه $L_1 \cap L_2$ مستقل از متن اسـت. ایـن خاصـیت، بسته بودن تحت اشتراک منظم خوانده می شود.

مثال

آیا زبان $L = \{a^n b^n : n \geq 0, n \neq 3\}$ مستقل از متن است؟

حل: می توان زبان L را به صورت $L = \{a^n b^n : n \geq 0\} \cap \overline{L_1}$ نوشت که $L_1 = \{a^3 b^3\}$ است. بنابراین زبـان L از اشتراک یک زبان مستقل از متن با زبان منظم تشکیل شده که با توجه به نکته قبل، زبان L مسـتقل از متن است. (علت منظم بودن $\overline{L_1}$: زبان L_1 متناهی است، بنابراین مـنظم نیـز هسـت.از طرفـی زبان‌هـای منظم تحت مکمل گیری بسته هستند.) ■

مثال

آیا زبان $L = \{w \in \{a,b,c\}^* : n_a(w) = n_b(w) = n_c(w)\}$ ، مستقل از متن است؟

حل: خیر- اشتراک این زبان با زبان منظم $L(a^* b^* c^*)$ برابر $\{a^n b^n c^n : n \geq 0\}$ مـی باشـد کـه مـی دانـیم مستقل از متن نمی باشد. بنابراین L مستقل از متن نیست. (با توجه به قضیه اشتراک منظم) ■

مثال

آیا مکمل زبان $L = \{w \in \{a,b,c\}^* : n_a(w) = n_b(w) = n_c(w)\}$ ، مستقل از متن است؟

حل: بله - مکمل این زبان از اجتماع چهار حالت زیر تشکیل شده:

۱- $n_a(w) < n_b(w)$ ۲- $n_a(w) > n_b(w)$ ۳- $n_a(w) < n_c(w)$ ۴- $n_a(w) > n_c(w)$

قبلا نشان داده شد که تمامی این چهار حالت مستقل از متن هستند(برای آنها گرامر مستقل از متن نوشته شد.) از آنجا که زبان‌های مستقل از متن تحت اجتماع بسته هستند، اجتماع این چهار حالت نیز مستقل از متن است.■

مثال

نشان دهید زبان‌های مستقل از متن تحت متمم بسته نیستند.

حل: زبان های زیر و متمم آنها مستقل از متن هستند:

$L_1 = \{a^n b^n c^m : n \ge 0, m \ge 0\}$ $L_2 = \{a^n b^m c^m : n \ge 0, m \ge 0\}$

بنابراین $L = \overline{L_1} \cup \overline{L_2}$ مستقل از متن است. متمم زبان L برابر است با:

$\overline{L} = \overline{\overline{L_1} \cup \overline{L_2}} = L_1 \cap L_2$

حاصل $L_1 \cap L_2$ برابر $\{a^n b^n c^n : n \ge 0\}$ است که مستقل از متن نیست.■

مثال

فرض کنید $L = \{a^n b^n : n \ge 0\}$، باشد که زبانی مستقل از متن است. آیا زبان L^* نیز مستقل از متن است؟

حل: بله- چون می‌توان برای آن یک گرامر مستقل از متن به صورت زیر نوشت:

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

■

برای اینکه ثابت کنیم یک زبان مستقل از متن نیست، کافی است حاصل اجتماع آن با یک زبان مستقل از متن، مستقل از متن نباشد. چون زبان‌های مستقل از متن نسبت به عمل اجتماع بسته هستند.

اگر L_1 مستقل از متن و L_2 یک زبان منظم باشد، آنگاه $L_1 - L_2$ مستقل از متن است. این خاصیت، بسته بودن تحت تفاضل منظم خوانده می شود.

اگر زبانی و متمم آن هر دو مستقل از متن باشند، آنگاه آن زبان لزوما منظم نیست.

لم تزریق برای زبان های مستقل از متن

به کمک لم تزریق می توان تشخیص داد که یک زبان مستقل از متن نیست.

لم تزریق: فرض کنید L یک زبان مستقل از متن نامتناهی باشد. آنگاه عدد صحیح و مثبت m وجود دارد، بطوریکه هر w متعلق به L با فرض $|w| \ge m$ را می توان به صورت $w = uvxyz$ با شرایط $|vxy| \le m$ و $|vy| \ge 1$ چنان تجزیه کرد که به ازای هر $i = 0,1,2,...$ داشته باشیم: $uv^i xy^i z \in L$

مثال

به کمک لم تزریق، نشان دهید که زبان $L = \{a^n b^n c^n : n > 0\}$ مستقل از متن نیست.

حل: فرض کنیم که L مستقل از متن باشد. حال رشته $w = a^n b^n c^n$ متعلق به L را به ۵ قسمت تجزیه کرد:

$x = a^n, y = b, z = b^{n-2}, u = b, v = c^n$

حال باید به ازای تمام $i \ge 0$ ، رشته $uv^i xy^i z \in L$، متعلق به L باشد، ولی به ازای i=2 این چنین نیست: $a^n b^i b^{n-2} b^i c^n = a^n b^2 b^{n-2} b^2 c^n = a^n b^{n+2} c^n \notin L$

تذکر مهم: با اعمال قوانین لم تزریق روی زبان $L = \{a^n b^n : n > 0\}$، متوجه می‌شویم که به ازای هر مقدار i ، رشته تزریق شده در L است. از این موضوع نمی‌توان نتیجه گرفت که L مستقل از متن است و فقط می‌توان گفت که از لم تزریق نتوانستیم نتیجه‌ای بگیریم.

مثال

توسط لم تزریق می‌توان نشان داد که زبانهای زیر مستقل از متن نمی‌باشند:

$\{a^n b^m :$ اول است n یا m اول است $\}$	$\{ww : w \in \{a,b\}^*\}$
$\{a^n b^m :$ اول است n و m اول نیست $\}$	$\{ww^R w : w \in \{a,b\}^*\}$
$\{a^n b^m :$ m,n هر دو اول هستند $\}$	$\{w : n_a(w) < n_b(w) < n_c(w)\}$

$\{a^n b^m : n = m^2\}$	$\{w : n_a(w)/n_b(w) = n_c(w)\}$
$\{a^n b^m : n \le m^2\}$	$\{w : n_a(w) = n_b(w) = n_c(w)\}$
$\{a^n b^m c^k : k = mn\}$	$\{w \in \{a,b,c\}^* : n_a^2(w) + n_b^2(w) = n_c^2(w)\}$
$\{a^n b^n c^m : n \ne m\}$	$\{a^n :$ یک عدد اول است $n\}$
$\{a^n b^m c^k : k > n, k > m\}$	$\{a^{n!} : n > 0\}$
$\{a^n b^m c^k : n < m, n \le k \le m\}$	$\{a^{n^2} : n \ge 0\}$
	$\{a^{nm} :$ هر دو اول هستند m,n $\}$
	$\{a^n b^m a^n b^m : n \ge 0, m \ge 0\}$

گرامر خطی

گرامر خطی،گرامر مستقل از متنی است که در سمت راست تمام قواعد آن، حداکثر یک متغیر وجود داشته باشد. زبان مستقل از متن L در صورتی خطی خوانده می شود که گرامر مستقل از متن خطی G وجود داشته باشد، بطوریکه $L = L(G)$ باشد.

مثال

آیا زبان $\{a^n b^m : m \le n \le 2m-1\}$ خطی است؟

حل: بله- چون می توان یک گرامر خطی برای آن نوشت:

$$S \to aAb \mid aaBb$$
$$A \to aAb \mid \lambda$$
$$B \to aaBb \mid aBb \mid ab \mid b$$

■

مثال

آیا زبان $\{a^n b^n c^m : n \ge 0, m \ge 0\}$ خطی است؟

حل: بله- چون می توان یک گرامر خطی برای آن نوشت:

$$S \to Sc \mid aAb \mid \lambda$$
$$A \to aAb \mid \lambda$$

■

مثال

زبان $\{a^n b^n : n \geq 0\}$ خطی است، چون می توان یک گرامر خطی برای آن نوشت. ■

مثال

نشان دهید زبان های خطی تحت اشتراک بسته نمی باشند.

حل: دو زبان زیر خطی هستند، چون می توان برای آنها یک گرامر خطی نوشت:

$$L_1 = \{a^n b^n c^m : n \geq 0, m \geq 0\}$$

$$L_2 = \{a^n b^m c^m : n \geq 0, m \geq 0\}$$

اشتراک این دو زبان برابر است با:

$$L_1 \cap L_2 = \{a^n b^n c^n : n \geq 0\}$$

این زبان خطی نیست. (حتی مستقل از متن هم نیست)

■

چند نکته:

۱- تمامی زبان های خطی، مستقل از متن هستند. اما راجع به حالت عکس، نمی توان با قاطعیت حرفی زد.

۲- خانواده زبان های خطی یکی از زیر مجموعه های مناسب خانواده زبان های مستقل از متن هستند.

۳- خانواده زبان های خطی، تحت اجتماع و هم ریختی و معکوس بسته است اما تحت اشتراک و الحاق بسته نیست.

۴- اگر L_1 خطی و L_2 منظم باشد، آنگاه L_1L_2 یک زبان خطی است.

لم تزریق برای زبان‌های خطی:

فرض کنید L یک زبان خطی نامتناهی باشد. آنگاه عدد صحیح و مثبت m وجود دارد، بطوریکه هر w متعلق به L با فرض $|w| \geq m$ را می توان به صورت $w = uvxyz$ با شرایط $|uvyz| \leq m$ و $|vy| \geq 1$ چنان تجزیه کرد که به ازای هر $i = 0,1,2,...$ داشته باشیم: $uv^i xy^i z \in L$

مثال

به کمک لم تزریق می توان نشان داد که زبان $\{w \in \{a,b\}^* : n_a(w) = n_b(w)\}$ خطی نمی باشد.■

تمرین فصل ۴

۱- گرامر G و زبان L_1 به صورت زیر مفروض است. آیا $L_1 = L(G)$ ؟

$$G: S \to aSb \mid SS \mid \lambda$$

$$L_1 = \{w \in \{a,b\}^* : n_a(w) = n_b(w)\}$$

۲- آیا گرامرهای زیر هم ارز می باشند؟

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

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

۳- گرامرهای زیر چه زبانی را تولید می کنند؟

الف	ب	پ	ت
$S \to AB$ $A \to xAy \mid \lambda$ $B \to zB \mid \lambda$	$S \to AB$ $A \to xAy \mid \lambda$ $B \to yBz \mid \lambda$	$S \to xSz \mid A$ $A \to yAz \mid \lambda$	$S \to Sz \mid Az$ $A \to xAy \mid xy$

۴- گرامرهای زیر چه زبانی را تولید می کنند؟

الف	ب	پ	ت
$S \to aSb \mid a \mid b$	$S \to AB$ $A \to xA \mid \lambda$ $B \to yBz \mid \lambda$	$S \to AM \mid MB$ $M \to aMb \mid \lambda$ $A \to aA \mid a$ $B \to bB \mid b$	$S \to aAb \mid bBa$ $A \to aaAb \mid ab$ $B \to bBa \mid a$

۵- گرامرهای زیر چه زبانی را تولید می کنند؟

الف	ب	پ	ت
$S \to aA$ $A \to bBaa$ $B \to ccAd \mid \lambda$	$S \to abA$ $A \to cdBb$ $B \to aAba \mid \lambda$	$S \to 0S1 \mid 1S0 \mid AA$ $A \to 0A \mid A1 \mid \lambda$	$S \to aSb \mid bY \mid Ya$ $Y \to aY \mid bY$

۶- گرامر زیر، چه زبانی را توصیف می‌کند؟

$$S \to ABC$$
$$A \to 0A1 \mid \lambda$$
$$B \to 1B \mid 1$$
$$C \to 1C0 \mid \lambda$$

۷- دو گرامر معادل با گرامر $S \to SS \mid (S) \mid \lambda$ بنویسید.

۸- یک گرامر مستقل از متن برای زبان زیر بنویسید.

$L_1 = \{w \in \{a,b\}^* : n_a(v) \ge n_b(v)$: برای هر v پیشوند w $\}$

$L_2 = \{w \in \{a,b\}^* : n_a(w) = n_b(w),\ n_a(v) \ge n_b(v)$: برای هر v پیشوند w $\}$

۹- آیا زبان زیر مستقل از متن است؟

$L = \{a^n b^n : n \ge 0,$ n مضربی از ۵ نیست. $\}$

۱۰- گرامری بنویسید که همه رشته های دارای حداقل سه a را تولید کند. $\Sigma = \{a,b\}$

۱۱- گرامر مستقل از متنی برای مجموعه تمام عبارات منظم روی الفبای {a,b} بنویسید.

۱۲- گرامری برای مجموعه اعداد صحیح در C بنویسید. (بدون محدویت برای تعداد ارقام)

۱۳- کدام یک از زبانهای زیر، مستقل از متن است؟

الف- $L=\{a^n ww^R a^n : w\in(a+b)^*, n\ge 0\}$

ب- $L=\{a^m cb^n : m\ne n\}\cup\{a^m db^{2m} : m\ge 0\}$

پ- $L=\{a^n b^m c^n : n\ge m\ge 0\}\cup\{a^i b^j c^i : j>i\ge 0\}$

ت- $L=\{a^n b^n c^m : n\ge 0, m\ge 0\}\cap\{a^{2n} b^{2n} c^{2m} : n\ge 0, m\ge 0\}$

ث- $L=\{a^* b^* c^*\}\cap\{w : w\in(a+b+c)^*,\ n_a(w)=n_b(w)=n_c(w)\}$

۱۴- گرامر مستقل از متنی برای زبان $L=\{(abc)^n(defg)^n : n\ge 1\}$ بنویسید.

۱۵- گرامر مستقل از متنی برای زبان $L=\{a(bcd)^n bce(fe)^n : n\ge 0\}$ بنویسید.

۱۶- گرامر تولید کننده هر زبان را بنویسید.

الف- $L=\{w : |w| \bmod 3 \ne |w| \bmod 2\}$ ($\Sigma=\{a\}$)

ب- $L=\{w : n_a(w)=2n_b(w)\}$ $\Sigma=\{a,b\}$

پ- $L=\{w : n_a(w)=n_b(w)+1\}$ $\Sigma=\{a,b,c\}$

ت- $L=\{w\in\{a,b,c\}^* : n_a(w)+n_b(w)<n_C(w)\}$

ث- $L=\{w\in\{a,b,c\}^* : n_a(w)+n_b(w)>n_C(w)\}$

ج- $L=\{a^n b^m c^k : n+2m=k\}$

چ- $L=\{a^n b^m c^k : n=m \text{ or } m\le k\}$

ح- $L=\{w_1 c w_2 : w_1, w_2\in\{a,b\}^+, w_1\ne w_2{}^R\}$

۱۷- با توجه به زبان $L=\{a^m b^m : m\ge 0\}$، آیا $L^3\cap L^4$ مستقل از متن است؟

۱۸- آیا $L_1\cap L_2\cap L_3$، مستقل از متن است؟

$L_1=\{a^p b^q a^p b^s : p,q,s\ge 0\}$

$L_2=\{a^pb^pa^rb^s : p,r,s\geq 0\}$

$L_3=\{a^pb^qa^rb^p : p,q,r\geq 0\}$

۱۹- آیا زبان L ، مستقل از متن است؟

$L=\{a^ib^jc^k : i,j,k\geq 0\}\bigcap\{w\in\{a,b,c\}^* : |w|_a=|w|_b\}$

۲۰- آیا متمم زبان $L=\{0^n1^n0^n : n\in N\}$ مستقل از متن است؟

۲۱- با توجه به زبان $L=\{(ab)^n(cd)^n : n\geq 0\}$ ، آیا L^* مستقل از متن است؟

۲۲- با فرض اینکه $L\subseteq\{1\}^*$ یک زبان باشد، در مورد نوع زبان L چه می توان گفت؟

۲۳- فرض کنید L_1زبانی منظم و L_3,L_2 زبان های مستقل از متن باشند، آیا زبان $(L_2\bigcup L_3)-L_1$ مستقل از متن است؟

۲۴- آیا زبان $L=\{a^ib^jc^k : i,j,k\geq 1,\ k\leq\max(i,j)\}$ مستقل از متن است؟

۲۵- زبان‌های منظم A و B را روی حروف الفبای Σ در نظر بگیرید. اگر تعریف کنیم $L=\bigcup_{n=1}^{\infty}(A^n\bigcap B^n)$ ، آیا در رابطه با L چه می توان گفت؟

۲۶- اگر L_1 مستقل از متن و L_2 چنین نباشد، آیا $L_1\bigcap L_2$ مستقل از متن است؟

۲۷- آیا زبان $L=\{a^nb^mc^m : n\geq 0, m\geq 0\}$ خطی است؟

۲۸- آیا زبان $L=\{a^mb^nc^k : m>5,\ n+k=3t\}$ ، مستقل از متن است؟ $(m,n,k,t\in N)$

۲۹- اگر زبان تولید شده توسط گرامر زیر را L بنامیم، آیا رابطه $L=L^*$ برقرار است؟

$$S\rightarrow Ab$$
$$A\rightarrow aAb\,|\,\lambda$$

۳۰- اگر زبان تولید شده توسط گرامر زیر را L بنامیم، آیا رابطه $L=L^*$ برقرار است؟

$$S\rightarrow SS\,|\,aSb\,|\,bSa\,|\,\lambda$$

۳۱- اگر $L=\{a^n b^m : n\geq 0, n<m\}$ باشد، آنگاه گرامری برای L^2 بنویسید.

۳۲- اگر $L=\{a^n b^m : n\geq 0, n<m\}$ باشد، آنگاه گرامری برای L^* بنویسید.

۳۳- یک گرامر که مجموعه اعداد صحیح در زبان C را تولید می کند را بنویسید.

۳۴- گرامرزیر چه زبانی را تولید می کند.

$$S\to bN \mid BN \mid AM$$
$$N\to aN \mid bN \mid \lambda$$
$$B\to aBb \mid b$$
$$A\to aAb \mid a$$
$$M\to aN \mid \lambda$$

۳۵- کدام یک از زبانهای زیر، مستقل از متن است؟ (r و q دو عدد صحیح و ثابت و مثبت می باشند.)

الف- $L=\{1^{r+mq}0^{rq} : m\geq 0\}$

ب- $L=\{1^{r+mq}0^{rm} : m\geq 0\}$

۳۶- آیا زبان $L=\{a^n b^m a^p b^q : n+m\leq p+q\}$ مستقل از متن است؟

۳۷- کدام یک از زبان‌های زیر خطی و کدام یک معین می باشند؟

$$L_1=\{w\in\{a,b\}^* : n_a(w)=n_b(w)\}$$

$$L_2=\{a^n b^n : n>0\}\cup\{a^n b^{2n} : n>0\}$$

۳۸- مثالی از ترکیبات زبان C بزنید،که توسط گرامر مستقل از متن،قابل توصیف نباشد.

۳۹- نشان دهید که برای هر گرامر مستقل از متن، یک گرامر معادل وجود دارد که قوانین آن بـه صـورت $A\to aBC$ یـا $A\to\lambda$ باشند.

$$A,B,C\in V$$
$$a\in\Sigma\cup\{\lambda\}$$

۴۰- گرامر تولید کننده زبان را بنویسید.

$L=\{w\in\{a,b\}^* : n_a(w)=n_b(w)$ و w شامل زیر رشته aab نیست$\}$

پاسخ تمرین فصل ۴

۱- خیر، گرامر G رشته‌هایی را تولید می‌کند که تعداد a و b در آنها برابر است و حتماً باید با a شروع شوند که این موضوع در زبان L_1 قید نشده است، بنابراین L_1 با $L(G)$ برابر نیست.

۲- خیر، چون رشته aa توسط گرامر اول تولید می شود، اما توسط گرامر دوم تولید نمی شود.

۳- زبان هر گرامر به صورت زیر است:

الف- $L=\{x^n y^n z^m : n\geq 0, m\geq 0\}$

قانون $A\rightarrow xAy$ رشته هایی به فرم $x^n y^n$ و قانون $B\rightarrow zB$ رشته هایی به فرم z^m تولید می‌کنند. در نهایت به علت وجود قانون $S\rightarrow AB$، رشته هایی به فرم $x^n y^n z^m$ تولید می شود.

ب- $L=\{x^n y^k z^m : k=n+m, n\geq 0, m\geq 0\}$

قانون $A\rightarrow xAy$ رشته هایی به فرم $x^n y^n$ و قانون $B\rightarrow yBz$ رشته هایی به فرم $y^m z^m$ تولید می‌کنند. در نهایت به علت وجود قانون $S\rightarrow AB$،رشته هایی به فرم $x^n y^n y^m z^m$ تولید می‌شود.

پ- $L=\{x^n y^m z^t : t=n+m, n\geq 0, m\geq 0\}$

ت- $L=\{x^n y^n z^k : n\geq 1, k\geq 1\}$

۴- زبان هر گرامر برابر است با:

الف- $L=\{a^n b^{n+1} : n\geq 1\}\cup\{a^{n+1} b^n : n\geq 1\}$

ب- $L=\{x^m y^n z^n : n\geq 0, m\geq 0\}$

قانون $A\rightarrow xA$ رشته هایی به فرم x^m و قانون $B\rightarrow yBz$ رشته هایی به فرم $y^n z^n$ تولید می‌کنند. در نهایت به علت وجود قانون $S\rightarrow AB$،رشته هایی به فرم $x^m y^n z^n$ تولید می شود.

پ- $L=\{a^n b^m : n\neq m\}$ (یا $a^+ a^n b^n + a^n b^n b^+$)

ت- $L=\{a^2a^{2k}b^kb^2 : k\geq 0\}\bigcup\{b^ta^{t+1} : t\geq 1\}$ رشته هایی که به کمک قوانین $S\to aAb, A\to aaAb\,|\,ab$ تولید می شود به فرم $a^2a^{2k}b^kb^2$ می‌باشد $(k\geq 0)$ و رشته هایی که به کمک قوانین $S\to bBa, B\to bBa\,|\,a$ تولید می‌شود به فرم b^ta^{t+1} می‌باشد. $(t\geq 1)$.

۵- زبان هر گرامر برابر است با:

الف- $L=\{a(bcc)^n baa(daa)^n : n\geq 0\}$

ب- $L=\{ab(cda)^n cdb(bab)^n : n\geq 0\}$

پ- $0^n AA1^n + 1^n AA0^n = 0^n(0+1)^*1^n + 1^n(0+1)^*0^n = (0+1)^*$

ت- $\{a,b\}^* - \{a^nb^n : n\geq 0\}$. یعنی تمامی رشته‌ها در $\{a,b\}^*$ به غیر از $\{a^nb^n : n\geq 0\}$.

۶- قانون $A\to 0A1\,|\,\lambda$ زبان 0^n1^n، قانون $B\to 1B\,|\,1$ زبان 1^+ و قانون $C\to 1C0\,|\,\lambda$، زبان 1^m0^m را تولید می کنند. در نهایت به علت وجود $S\to ABC$ ، این گرامر زبان زیر را تولید می‌کند:

$$\{0^n1^n1^+1^m0^m\}=\{0^n1^{n+m+t}0^m :\ t>0, n\geq 0, m\geq 0\}$$

که می توان آن را به صورت $L=\{0^i1^j0^k : j>i+k\}$ نشان داد.

۷- دو گرامر زیر با گرامر $S\to SS\,|\,(S)\,|\,\lambda$ معادل هستند:

$G1: S\to S(S)\,|\,\lambda$
$G2: S\to S(SS)\,|\,\lambda$

۸- گرامر برای L_1 :

$G1: S\to aS\,|\,aSbS\,|\,\lambda$

گرامر برای L_2 :

$G2: S\to aSb\,|\,SS\,|\,\lambda$

۹- بله- چون می توان برای آن یک گرامر مستقل از متن به صورت زیر نوشت:

$$S \rightarrow a^5Sb^5 \mid a^4b^4 \mid a^3b^3 \mid a^2b^2 \mid ab$$

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

$$S \rightarrow XaXaXaX$$
$$X \rightarrow aX \mid bX \mid \lambda$$

۱۱- گرامر مورد نظر عبارت است از:

$$E \rightarrow E+E \mid E.E \mid E^* \mid (E) \mid \lambda \mid \phi \mid a \mid b$$

۱۲- گرامر مورد نظر عبارت است از:

$$S \rightarrow AB$$
$$A \rightarrow + \mid - \mid \lambda$$
$$B \rightarrow D \mid DB$$
$$D \rightarrow 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$$

۱۳- بررسی گزاره ها:

الف- بله، چون می توان یک گرامر مستقل از متن برای آن ارائه داد:

$$S \rightarrow aSa \mid bMb$$
$$M \rightarrow aMa \mid bMb \mid \lambda$$

ب- بله، چون می توان یک گرامر مستقل از متن برای آن ارائه داد:

$$S \rightarrow A \mid B$$
$$A \rightarrow aAb \mid aA \mid Ab \mid ac \mid cb$$
$$B \rightarrow aBbb \mid d$$

پ- زبان L به صورت $\{a^nb^kc^n : n,k \geq 0\}$ است که مستقل از متن می‌باشد.

ت- این زبان به صورت $L=\{a^{2n}b^{2n}c^{2m} : n \geq 0, m \geq 0\}$ است، که مستقل از متن می باشد.

ث- زبان L برابر $\{a^nb^nc^n : n \geq 0\}$ است، که مستقل از متن نمی باشد.

۱۴- این گرامر به صورت زیر است:

$$S \to abXefg$$
$$X \to cSd \mid cd$$

۱۵- این گرامر به صورت زیر است:

$$S \to aX$$
$$X \to bcYe$$
$$Y \to dXf \mid \lambda$$

۱۶- گرامر تولید کننده هر زبان را بنویسید.

الف- طول رشته نباید 6k، 6k+1 باشد.

$$S \to aaaaaaS \mid aa \mid aaa \mid aaaa \mid aaaaa$$

ب- گرامر به صورت زیر است:

$$S \to SS \mid \lambda$$
$$S \to aabS \mid aaSb \mid aSab \mid Saab$$
$$S \to abaS \mid abSa \mid aSba \mid Saba$$
$$S \to baaS \mid baSa \mid bSaa \mid Sbaa$$

پ-

$$S \to NXNaNXN$$
$$X \to aXb \mid bXa \mid XN \mid NX \mid \lambda$$
$$N \to cN \mid \lambda$$

ت-

$$S \to aSc \mid bSc \mid cSa \mid cSb \mid SS \mid XS \mid c$$
$$X \to cX \mid \lambda$$

ث-

$$S \to aSc \mid bSc \mid cSa \mid cSb \mid SS \mid AS \mid BS \mid a \mid b$$
$$A \to aA \mid \lambda$$
$$B \to bB \mid \lambda$$

ج-

$$S \to aSc \mid A$$
$$A \to bAcc \mid \lambda$$

چ-

$$S \to ND \mid AMD$$
$$N \to aNb \mid \lambda$$
$$M \to bMc \mid \lambda$$
$$D \to cD \mid \lambda$$
$$A \to aA \mid \lambda$$

ح-

$$S \to A \mid c$$
$$A \to aAa \mid bAb \mid aMb \mid bMa \mid aNc \mid bNc \mid cNa \mid cNb$$
$$M \to aM \mid bM \mid cN$$
$$N \to aN \mid bN \mid \lambda$$

۱۷- بله- زبان L^3 و L^4 به صورت زیر می باشند:

$$L^3 = \{a^m b^m a^n b^n a^p b^p : m,n,p \ge 0\}$$

$$L^4 = \{a^m b^m a^n b^n a^p b^p a^k b^k : m,n,p,k \ge 0\}$$

اشتراک آنها برابر L^3 است که مستقل از متن است.

۱۸- هر ۳ زبان مستقل از متن است، اما اشتراک آنها $\{a^n b^n a^n b^n : n \ge 0\}$ مستقل از متن نمی‌باشد.

۱۹- بله- زبان L حاصل اشتراک یک زبان مستقل از متن و یک زبان منظم است، بنابراین مستقل از متن است.

۲۰- متمم زبان داده شده یعنی $\{0^n 1^m 0^k : n \ne m \ne k\}$ مستقل از متن است.(خود زبان L مستقل از متن نیست.)

۲۱- بله- زبان L مستقل از متن می باشد و چون زبان مستقل از متن تحت بستار بسته است، بنابراین L^* نیز مستقل از متن است.

۲۲- این زبان حتما مستقل از متن نیست. مانند $L=\{1^{n^2}: n>0\}$. البته اگر بدانیم L مستقل از متن است، آنگاه L حتما منظم خواهد بود.

۲۳- بله- زبان $(L_2 \cup L_3)-L_1$ را به صورت $(L_2 \cup L_3)\cap \overline{L_1}$ می نویسیم. می دانیم که:

الف- زبانهای مستقل از متن نسبت به اجتماع بسته هستند،پس $L_2 \cup L_3$ مستقل از متن است.

ب- زبان منظم نسبت به عمل متمم بسته است، پس $\overline{L_1}$ منظم است.

ج- اشتراک یک زبان مستقل از متن با یک زبان منظم، زبانی است مستقل از متن.

در نتیجه زبان $(L_2 \cup L_3)\cap \overline{L_1}$ مستقل از متن است.

۲۴- بله- زبان L با شرط $k \le \max(i,j)$، از اجتماع دو زبان با شرطهای $k \le i$ و $k \le j$ تشکیل شده:

$L=\{a^i b^j c^k : i,j,k \ge 1, k \le i\} \cup \{a^i b^j c^k : i,j,k \ge 1, k \le j\}$

بنابراین چون این دو زبان مستقل از متن بوده و زبان های مستقل از متن تحت اجتماع بسته هستند، زبان L نیز مستقل از متن می باشد.

۲۵- بله- چون A و B منظم هستند، آنگاه برای هر n عبارت $(A^n \cap B^n)$ منظم بوده ولی چون زبان‌های منظم تحت اجتماع نامتناهی یعنی $L=\bigcup_{n=1}^{\infty}(A^n \cap B^n)$، بسته نمی باشند، زبان L لزوما منظم نیست ولی حتما مستقل از متن می باشد.

۲۶- اگر L_1 مستقل از متن و L_2 منظم باشد، آنگاه $L_1 \cap L_2$ مستقل از متن است. اما در این گزینه در مورد زبان L_2 اطلاعی نداریم و در نتیجه $L_1 \cap L_2$ ممکن است مستقل از متن باشد و ممکن است نباشد.

۲۷- بله- چون می توان یک گرامر خطی برای آن نوشت:

$S \to aS \mid bAc \mid \lambda$
$A \to bAc \mid \lambda$

۲۸- با توجه به محدودیت های داده شده(تعداد a بیشتر از 5 و مجموع تعداد b ها و cها مضرب 3 باشد)، زبان داده شده منظم است، و در نتیجه مستقل از متن نیز هست.

۲۹- خیر- زبان گرامر برابر $L=\{a^nb^{n+1}: n\ge 0\}$ است که شامل λ نیست. پس $L \neq L^*$.

۳۰- بله- زبان گرامر برابر $L=\{w: n_a(w)=n_b(w)\}$ است. اگر L^* را به واحدهای شامل یک L تقسیم کنیم، در هر کدام از آنها تعداد a با b برابر است. پس $L=L^*$.

۳۱- زبان L^2 برابر $L^2=\{a^nb^ma^pb^q: n\ge 0, n<m, p\ge 0, p<q\}$ است. گرامر آن برابر است با:

$S \to XX$
$X \to aXb \mid bY$
$Y \to bY \mid \lambda$

۳۲- گرامر برابر است با:

$S \to SS \mid X \mid \lambda$
$X \to aXb \mid bY$
$Y \to bY \mid \lambda$

۳۳- گرامر برابر است با:

$S \to IDA$
$I \to + \mid - \mid \lambda$
$D \to 0 \mid 1 \mid 2 \mid ... \mid 9$
$A \to DA \mid \lambda$

۳۴- متمم زبان $L=\{a^nb^m : n\ge 0\}$.

۳۵- بررسی زبان ها:

الف-خیر، با فرض اینکه r=2,q=3 داریم: $L=\{1^{2+3m}0^6 : m\ge 0\}$. در این صورت، تعداد صفرها ثابت شده و وابستگی به تعداد 1 ها ندارد. پس زبان منظم می باشد.

ب- بله، با فرض اینکه r=2,q=3 داریم: $L=\{1^{2+3m}0^{2m}:m\geq 0\}$. در این صورت،0 و 1 به یکدیگر وابسته هستند و در نتیجه زبان مستقل از متن است.

زبان را می توان به صورت $(1^q)^m 1^r (0^r)^m$ نشان داد. به علت وابستگی توانی و محدود نبودن m ، نمی‌توان یک گرامر منظم طراحی کرد. اما می‌توان یک گرامر مستقل از متن برای آن طراحی کرد.

۳۶- بله- چون می توان برای آن یک گرامر مستقل از متن نوشت:

$$S \to aSb \mid Sb \mid bAa \mid aBa \mid \lambda$$
$$A \to bAa \mid Aa \mid \lambda$$
$$B \to aBa \mid bAa \mid Ba \mid \lambda$$

۳۷- زبان مستقل از متن L_1 معین است ولی خطی نیست. زبان L_2 خطی است ولی معین نیست.

۳۸- تعریف توابع در C ، توسط گرامر مستقل از متن، قابل توصیف نیست، چون ترتیب توابع در این زبان مهم نیست.

۳۹- هر گرامر مستقل از متن را می توان به فرم نرمال چامسکی در آورد. پس ابتدا گرامر را به فرم نرمال چامسکی تبدیل کرده و سپس قواعد به فرم $A \to a$ را حذف کرده و به جای آن قواعد زیر را قرار می‌دهیم:

$$A \to aMN$$
$$M \to \lambda$$
$$N \to \lambda$$

قواعد دیگر نیازی به تغییر ندارند، چون قواعد به فرم $A \to BC$ در فرم مطلوب هستند. a در قوائد $A \to aBC$ می تواند λ باشد.

فصل ۵

ابهام - ساده سازی گرامر – فرم‌های نرمال

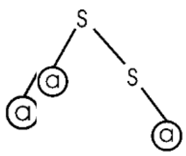
ابهام در گرامر و زبان

گرامر مستقل از متن G در صورتی مبهم خوانده می شود که یک رشته $w \in L(G)$ وجود داشته باشد که حداقل دو درخت اشتقاق مجزا داشته باشد. به بیان دیگر، ابهام به طور ضمنی به معنای وجود دو یا چند اشتقاق چپ ترین یا راست ترین، نیز می باشد.

مثال

آیا گرامر $S \to aS \mid aa \mid a$ مبهم است؟

حل: بله- چون به طور نمونه برای تولید رشته aa ، دو درخت اشتقاق وجود دارد:



مثال

آیا گرامر زیر مبهم است؟

$$S \to AB \mid aaB$$
$$A \to a \mid Aa$$
$$B \to b$$

حل: بله- چون به طور نمونه برای تولید رشته abb ، دو اشتقاق چپ وجود دارد:

$1: S \Rightarrow aaB \Rightarrow aab$

$2: S \Rightarrow AB \Rightarrow AaB \Rightarrow aaB \Rightarrow aab$

■

مثال

آیا گرامر $S \to aSbS \mid bSaS \mid \lambda$ مبهم است؟

حل: بله- چون به طور نمونه برای تولید رشته abab ، دو اشتقاق چپ وجود دارد:

$1: S \Rightarrow aSbS \Rightarrow abSaSbS \Rightarrow abaSbS \Rightarrow ababS \Rightarrow abab$
$2: S \Rightarrow aSbS \Rightarrow abS \Rightarrow abaSbS \Rightarrow ababS \Rightarrow abab$

■

مثال

آیا گرامر $S \to aSb \mid SS \mid \lambda$ مبهم است؟

حل: بله- چون به طور نمونه برای تولید رشته ab ، دو اشتقاق چپ وجود دارد:

$1: S \Rightarrow aSb \Rightarrow ab$
$2: S \Rightarrow SS \Rightarrow aSbS \Rightarrow abS \Rightarrow ab$

مثال

آیا زبان تولید شده توسط گرامر مبهم $S \to aSb \mid SS \mid \lambda$، مبهم است؟

حل: خیر- چون می توان یک گرامر غیر مبهم نظیر این گرامر نوشت:

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

■

مثال

آیا گرامر زیر مبهم است؟

$S \to aB \mid A$
$A \to aA \mid \lambda$
$B \to bB \mid a$

حل: بله- چون به طور نمونه برای تولید رشته aa ، دو اشتقاق چپ وجود دارد:

$1: S \Rightarrow A \Rightarrow aA \Rightarrow aaA \Rightarrow aa$
$2: S \Rightarrow aB \Rightarrow aa$

بنابراین گرامر منظم نیز می تواند مبهم باشد.

■

خانواده زبان های مستقل از متن غیر مبهم،تحت اجتماع بسته نمی باشند.

مثال

نشان دهید زبان $L = \{a^n b^n c^m : n,m \geq 0\} \cup \{a^n b^m c^m : n,m \geq 0\}$، یک زبان مستقل از متن ذاتا مبهم است. (این زبان را می توان به صورت $L = \{0^i 1^j 2^k : i = j \ or \ j = k\}$ نیز نشان داد.)

حل: گرامر این زبان به صورت زیر است. زبان $\{a^n b^n c^m : n,m \geq 0\}$ توسط $S \to S_1$ و زبان $\{a^n b^m c^m : n,m \geq 0\}$ توسط $S \to S_2$ تولید می شود.

$S \to S_1 \mid S_2$

$S_1 \to S_1 c \mid A$
$A \to aAb \mid \lambda$

$S_2 \to aS_2 \mid B$
$B \to bBc \mid \lambda$

این گرامر مبهم است، چون برای رشته $a^n b^n c^n$ دو اشتقاق مجزا وجود دارد.

تذکر:هر دو زبان $\{a^n b^n c^m : n,m \geq 0\}$ و $\{a^n b^m c^m : n,m \geq 0\}$ مستقل از متن غیر مبهم هستند. ■

- اگر L مستقل از متن بوده و هر گرامر تولید کننده L ، مبهم باشد، آنگاه این زبان ذاتا مبهم است.
- زبان‌های منظم نمی توانند ذاتا مبهم باشند.
- تمام S-گرامرها، فاقد ابهام هستند.

ساده سازی گرامرهای مستقل از متن

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

حذف متغیرها و قوانین بی فایده

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

۱- قابل دسترس نبودن از طریق متغیر شروع گرامر

۲- ناتوانی در اشتقاق یک رشته پایانی

- قانونی که شامل یک متغیر بی فایده باشد، قانون بی فایده نامیده می شود.
- گرامر $G=(V,T,S,P)$ را یک گرامر مستقل از متن فرض کنید. آنگاه گرامر هم ارز $\hat{G}=(\hat{V},\hat{T},S,\hat{P})$ وجود دارد که شامل هیچ متغیر یا قانون بی فایده نمی باشد.

مثال

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

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

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

مثال

در گرامر زیر، کدام متغیر بی فایده است؟

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

حل: متغیر A بی فایده است، چون نمی تواند رشته ای از پایانی ها را تولید کند. بنابراین قانون $S \to A$ و $A \to aA$ را می توان حذف کرد، بدون اینکه تغییری در زبان ایجاد شود. ■

مثال

در گرامر زیر کدام متغیرها بی فایده هستند.

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

حل: متغیر C بی فایده است، چون یک رشته پایانی را تولید نمی کند. همچنین متغیر B بی فایده است، چون از متغیر شروع قابل دستیابی نمی باشد. بنابراین می توان متغیرهای C و B و قوانین مربوط به آنها را حذف کرد. گرامر نهایی به صورت زیر می باشد:

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

■

مثال

گرامر زیر را ساده کنید.

$$S \rightarrow aAb \mid bBa \mid bCa$$
$$A \rightarrow aaAb \mid ab$$
$$B \rightarrow bBa \mid a$$
$$C \rightarrow aC \mid bC$$

حل: در گرامر داده شده، متغیرC، غیر مفید است و قاعده هایی که از C استفاده می‌کننـد را می‌تـوان از گرامـر حـذف کـرد. بنابراین گرامر به صورت زیر می باشد:

$$S \rightarrow aAb \mid bBa$$
$$A \rightarrow aaAb \mid ab$$
$$B \rightarrow bBa \mid a$$

■

مثال

در گرامر زیر متغیرهای بی فایده را حذف کنید.

$$S \rightarrow AC \mid BS \mid B$$
$$A \rightarrow aA \mid aF$$
$$B \rightarrow CF \mid b$$
$$C \rightarrow cC \mid D$$
$$D \rightarrow aD \mid BD \mid C$$
$$E \rightarrow aA \mid BSA$$
$$F \rightarrow bB \mid b$$

حل: ابتدا متغیرهایی را که به رشته‌ای از الفبا نمی‌رسند یعنی C و D را حذف می‌کنیم:

$$S \rightarrow BS \mid B$$
$$A \rightarrow aA \mid aF$$
$$B \rightarrow b$$
$$E \rightarrow aA \mid BSA$$

$F \to bB \mid b$

سپس متغیرهایی را که نمی‌توان از S به آنها رسید، یعنی A و E و F را حذف می‌کنیم:

$S \to BS \mid B$

$B \to b$

■

حذف قوانین λ

هر قانونی از یک گرامر مستقل از متن به فرم $A \to \lambda$ را قانون λ می گویند. این قوانین در بعضی مواقع نامطلوب می باشند. هر متغیر A که اشتقاق $A \overset{*}{\Longrightarrow} \lambda$ برای آن امکان پذیر باشد را متغیر میرا می نامند.

برخی گرامرها زبانهایی را تولید می کنند که هر چند فاقد λ هستند، تعدادی متغیر میرا یا قانون λ در آنها وجود دارند. در این موارد، می توان قوانین λ را حذف کرد.

به ازای هر گرامر مستقل از متن، یک گرامر هم ارز فاقد قانون λ ، وجود دارد.

مثال

گرامر زیر، زبان $\{a^n b^n : n \geq 1\}$ را تولید می کند. این زبان فاقد λ می باشد.

$S \to aAb$
$A \to aAb \mid \lambda$

برای حذف قانون $A \to \lambda$، دو قانون جدید که با جایگزینی λ در A های سمت راست بدست آمده اند را بـه گرامـر اضـافه می کنیم:

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

■

مثال

قوانین λ را در گرامر زیر حذف کنید.

$S \to AB$

$A \to aA \mid \lambda$

$B \to bB \mid \lambda$

حل: ابتدا $A \to \lambda$ را حذف می کنیم:

$S \to AB \mid B$
$A \to aA \mid a$
$B \to bB \mid b$

سپس $B \to \lambda$ را حذف می کنیم:

$S \to AB \mid A \mid B \mid \lambda$
$A \to aA \mid a$
$B \to bB \mid b$

تذکر مهم: نیازی به حذف قانون $S \to \lambda$ نمی باشد، چون S ، متغیر شروع است.

مثال

در گرامر زیر، قوانین λ را حذف کنید؟

$S \to ABaC$
$A \to BC$
$B \to b \mid \lambda$
$C \to D \mid \lambda$
$D \to d$

حل: ابتدا $B \to \lambda$ را حذف می کنیم:

$S \to ABaC \mid AaC$
$A \to BC \mid C$
$B \to b$
$C \to D \mid \lambda$
$D \to d$

سپس $C \to \lambda$ را حذف می کنیم:

$$S \to ABaC \mid AaC \mid ABa \mid Aa$$
$$A \to BC \mid C \mid B \mid \lambda$$
$$B \to b$$
$$C \to D$$
$$D \to d$$

در نهایت $A \to \lambda$ را حذف می کنیم:

$$S \to ABaC \mid AaC \mid ABa \mid Aa \mid BaC \mid aC \mid Ba \mid a$$
$$A \to BC \mid B \mid C$$
$$B \to b$$
$$C \to D$$
$$D \to d$$

■

حذف قوانین واحد

هر قانونی از یک گرامر مستقل از متن به فرم $A \to B$ که در آن $A, B \in V$، قانون واحد یا یکه نامیده می‌شود. این قوانین گاهی اوقات نامطلوب هستند و باید حذف شوند.

مثال

از گرامر زیر قوانین واحد را حذف کنید.

$$S \to Aa \mid B$$

$$B \to A \mid bb$$

$$A \to bc \mid a \mid B$$

حل:

$$S \to Aa \mid bc \mid a \mid bb$$

$$B \to bc \mid a \mid bb$$
$$A \to bc \mid a \mid bb$$

توجه کنید که در اثر حذف قوانین واحد، متغیر B و قوانین مربوط به آن بی فایده شده اند. ■

ممکن است حذف قوانین λ، باعث تولید قوانین واحد شود که قبلا وجود نداشته اند.

زبان L را یک زبان مستقل از متن فاقد λ فرض کنید. آنگاه یک گرامر مستقل از متن وجود خواهد داشت که L را تولید کرده و فاقد هرگونه قانون بی فایده، قانون λ و قانون واحد باشد.

مثال

در گرامر زیر قوانین λ را حذف کنید.

$$S \to aA$$
$$A \to BB$$
$$B \to aBb \mid \lambda$$

حل: با حذف $B \to \lambda$ گرامر زیر بدست می آید:

$$S \to aA \mid a$$
$$A \to BB \mid B$$
$$B \to aBb \mid ab$$

در این گرامر قانون واحد $A \to B$ تولید شده که قبلا وجود نداشت. ■

مثال

در گرامر زیر تمامی قوانین λ، واحد و بی فایده را حذف کنید.

$$S \to aA \mid aBB$$
$$A \to aaA \mid \lambda$$
$$B \to bC \mid bbC$$
$$C \to B$$

حل: ابتدا قانون $A \to \lambda$ را حذف می کنیم:

$$S \to aA \mid aBB \mid a$$
$$A \to aaA \mid aa$$
$$B \to bC \mid bbC$$
$$C \to B$$

سپس قانون واحد $C \to B$ را حذف می کنیم:

$$S \to aA \mid aBB \mid a$$
$$A \to aaA \mid aa$$
$$B \to bC \mid bbC$$
$$C \to bC \mid bbC$$

نهایتا اینکه B و C بی فایده هستند، بنابراین داریم:

$$S \to aA \mid a$$
$$A \to aaA \mid aa$$

زبان تولید شده بوسیله این گرامر $L((aa)^* a)$ می باشد. ■

فرم‌های نرمال گرامر مستقل از متن

برای گرامر مستقل از متن، دو فرم نرمال چامسکی و گریباخ وجود دارد.

فرم نرمال چامسکی

یک گرامر مستقل از متن در صورتی در فرم نرمال چامسکی قرار دارد که تمام قوانین آن به یکی از دو فرم $A \to BC$ و یا $A \to a$ باشد. که در آن A,B,C عضو V بوده و a عضو T است.

مثال

گرامر زیر در فرم نرمال چامسکی قرار دارد:

$S \to AS \mid BS \mid a$

$A \to SA \mid a$

$B \to SB \mid b$

■

مثال

گرامر زیر را به فرم نرمال چامسکی تبدیل کنید.

$S \to ABa$

$A \to aab$

$B \to Ac$

حل:

قدم دوم : نرمال کردن قانون اول و سوم	قدم اول : معرفی متغیرهای جدید X,Y,Z
$S \to AT$	$S \to ABX$

$X \to a$ $A \to XXY$ $Y \to b$ $B \to AZ$ $Z \to c$	$T \to BX$ $A \to XF$ $F \to XY$ $B \to AZ$ $X \to a, Y \to b, Z \to c$

مثال

گرامر زیر را به فرم نرمال چامسکی تبدیل کنید.

$S \to aAB$

$A \to aA \mid b$

$B \to bB \mid b$

حل:

قدم اول	قدم دوم
$S \to TAB$ $T \to a$ $A \to TA \mid b$ $B \to FB \mid b$ $F \to b$	$S \to TK$ $K \to AB$ $T \to a$ $A \to TA \mid b$ $B \to FB \mid b$ $F \to b$

مثال

گرامر زیر را به فرم نرمال چامسکی تبدیل کنید.

$S \to aAbB \mid ab$

$A \to ABS \mid a$

$B \to bb$

حل:

قدم سوم	قدم دوم	قدم اول
$S \to TK \mid TF$ $K \to AX$ $X \to FB$ $A \to AU \mid a$ $U \to BS$ $B \to FF$ $T \to a$ $F \to b$	$S \to TK \mid TF$ $K \to AFB$ $A \to AU \mid a$ $U \to BS$ $B \to FF$ $T \to a$ $F \to b$	$S \to TAFB \mid TF$ $A \to ABS \mid a$ $B \to FF$ $T \to a$ $F \to b$

مثال

گرامر زیر را به فرم نرمال چامسکی تبدیل کنید.

$S \to abAB$
$A \to bAB \mid \lambda$
$B \to BAa \mid A \mid \lambda$

حل: ابتدا قواعد λ را حذف می کنیم:

حذف $A \to \lambda$	حذف $B \to \lambda$
$S \to abAB \mid abB$ $A \to bAB \mid bB$ $B \to BAa \mid A \mid Ba \mid \lambda$	$S \to abAB \mid abB \mid abA \mid ab$ $A \to bAB \mid bB \mid bA \mid b$ $B \to BAa \mid A \mid Ba \mid Aa \mid a$

سپس قاعده یکه $B \to A$ را حذف می کنیم:

$S \to abAB \mid abB \mid abA \mid ab$
$A \to bAB \mid bB \mid bA \mid b$
$B \to BAa \mid Ba \mid Aa \mid a \mid bAB \mid bB \mid bA \mid b$

در نهایت گرامر را به فرم نرمال چامسکی در می آوریم:

قدم اول : معرفی متغیرهای جدید Y, X	قدم دوم : معرفی متغیرهای جدید P,T,K,N, M
$S \to XYAB \mid XYB \mid XYA \mid XY$ $A \to YAB \mid YB \mid YA \mid b$ $B \to BAX \mid BX \mid AX \mid a \mid YAB \mid YB \mid YA \mid b$ $X \to a$ $Y \to b$	$S \to XM \mid XN \mid XK \mid XY$ $M \to YT$ $T \to AB$ $N \to YB$ $A \to YT \mid YB \mid YA \mid b$ $B \to BP \mid BX \mid AX \mid a \mid YT \mid YB \mid YA \mid b$ $P \to AX$ $X \to a$ $Y \to b$

در فرم نرمال چامسکی، تولید رشته‌ای به طول n دارای اشتقاقی با طول $2n-1$ می‌باشد. $(n \ge 1)$ (n اشتقاق از $A \to a$ و $n-1$ اشتقاق از $A \to BC$)

فرض کنید G یک گرامر مستقل از متن(CFG) در فرم نرمال چامسکی بـا b متغیـر باشـد. در ایـن صورت اگر G بتواند رشته ای را با تعداد گام هـای اشـتقاق بیشـتر از b تولیـد کنـد،آنگاه L(G) نامحـدود است.

فرم نرمال گریباخ

یک گرامر مستقل از متن در فرم نرمال گریباخ است هرگاه تمام قوانین آن به فرم $A \to aX$ باشند، که در آن، $x \in V^*, a \in T$.

مثال

گرامر $S \to abSb \mid aa$ را به فرم نرمال گریباخ تبدیل کنید.

حل: متغیرهای جدید A و B را معرفی می کنیم که مترادف با a و b هستند:

$S \to aBSB \mid aA$

$A \to a$

$B \to b$

■

مثال

گرامر $S \to ab \mid aS \mid aaS$ را به فرم نرمال گریباخ تبدیل کنید.

$S \to aX \mid aS \mid aYS$
$X \to b$
$Y \to a$

■

مثال

گرامر زیر را به فرم نرمال گریباخ تبدیل کنید.

$S \to AB$

$A \to aA \mid bB \mid b$

$B \to b$

حل: قاعده $S \to AB$ با تعریف فرم نرمال گریباخ مغایر است. بنابراین به جای A در این قانون از قوانین جایگزین آن استفاده می کنیم:

$S \to aAB \mid bBB \mid bB$

$A \to aA \mid bB \mid b$

$B \to b$

■

به ازای هرگرامر مستقل از متن در صورتیکه شامل λ نباشد، یک گرامر معادل به فرم نرمال گریباخ وجود دارد.

در فرم نرمال گریباخ، برای تولید رشته‌ای به طول n ، به اشتقاقی با طول n نیـاز اسـت. چـون در هـر مرحلـه یکـی از نمادهای رشته ایجاد می‌شود.

مثال

گرامر زیر را به فرم نرمال گریباخ تبدیل کنید.

$S \to ABb \mid a$
$A \to aaA \mid B$
$B \to bAb$

حل: ابتدا قاعده یکه $A \to B$ را حذف می کنیم:

$S \to ABb \mid a$
$A \to aaA \mid bAb$
$B \to bAb$

سپس متغیر A در $B \to ABb$ را جایگزین می کنیم:

$S \to aaABb \mid bAbBb \mid a$
$A \to aaA \mid bAb$
$B \to bAb$

سپس به فرم نرمال گریباخ تبدیل می کنیم:

$S \to aXABY \mid bAYBY \mid a$
$A \to aXA \mid bAY$
$B \to bAY$
$X \to a$
$Y \to b$

تمرین فصل ۵

۱- درخت اشتقاق را تعریف کنید.

۲- با توجه به گرامر زیر، درخت اشتقاق برای تولید رشته abbbb ، را رسم کنید.

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

۳- اگر G یک گرامر مستقل از متن باشد، آنگاه هر $w \in L(G)$ اشتقاق چپ دارد یا یک اشتقاق راست؟

۴- ترتیب حذف قواعد تولید λ، بی فایده و واحد را از یک گرامر مستقل از متن که زبان آن فاقد λاست را مشخص کنید.

۵- از گرامر زیر، قوانین نامطلوب (λ، واحد و بی فایده) را حذف کنید.

$$S \rightarrow aA \mid aBB$$
$$A \rightarrow aaA \mid \lambda$$
$$B \rightarrow bB \mid bbC$$
$$C \rightarrow B$$

۶- آیا حذف قوانین بی فایده لزوما موجب تولید گرامرهای کمینه می‌شود؟

۷- قوانین بی فایده را از گرامر زیر حذف کنید.

$S \to aA \mid a \mid B \mid C$
$A \to aB \mid \lambda$
$B \to Aa$
$C \to cCD$
$D \to ddd$

۸- قوانین λ و بی فایده را از گرامر زیر حذف کنید.

$S \to AaB \mid aaB$
$A \to \lambda$
$B \to aaA \mid \lambda$

۹- گرامر زیر را به فرم نرمال چامسکی تبدیل کنید.

$S \to aSb \mid ab$

۱۰- گرامر زیر را به فرم نرمال چامسکی تبدیل کنید.

$S \to aSaA \mid A$
$A \to abA \mid b$

۱۱- گرامر $S \to aSb \mid ab$ را به فرم نرمال گریباخ تبدیل کنید.

۱۲- گرامر $S \to aSb \mid aS \mid aaS$ را به فرم نرمال گریباخ تبدیل کنید.

۱۳- توسط الگوریتم CYK که توسط کوک، یانگر و کسامی ابداع شد، چه موضوعی را می توان تصمیم گیری کرد؟پیچیدگی این الگوریتم را مشخص کنید.

۱۴- اگر G گرامر مستقل از متنی باشد که هر یک از قوانین آن به فرم $A \to v$ بوده و $|v|>1$ ، آنگاه محدوده ارتفاع درخت اشتقاق به ازای تمامی $w \in L(G)$ را مشخص کنید.

۱۵- فرض کنید G=(V,T,S,P) یک گرامر مستقل از متن(CFG) بدون هیچ قانون λ یا قانون واحدی باشد. بعلاوه، فرض کنید K حداکثر تعداد سمبل های موجود در طرف راست قوانین P باشد. گرامر معادل در فرم نرمال چامسکی، حداکثر دارای چه تعداد قانون تولید است؟

۱۶- گرامر زیر را با BNF نمایش دهید.

$E \to T \mid E + T$
$T \to F \mid T * E$

حل تمرین فصل ۵

۱- فرض کنید $G=(V,T,S,P)$ یک گرامر مستقل از متن باشد. یک درخت مرتب، درخت اشتقاقی برای G خواهد بود اگر و تنها اگر خواص زیر را داشته باشد:

الف- ریشه دارای نام S است.

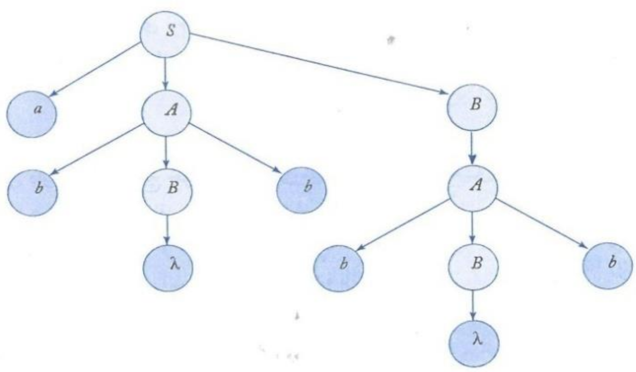
ب- هر یک از برگها دارای نامی از $T\cup\{\lambda\}$ باشد.

پ- هر یک از گره های میانی دارای نامی از V است.

ت- اگر یکی از گره ها دارای نام $A\in V$ بوده و فرزندان آن هم از چپ به راست به صورت $a_1,a_2,...,a_n$ نامگذاری شوند، آنگاه P باید قانونی به فرم $A\rightarrow a_1a_2...a_n$ داشته باشد.

ث- برگ های دارای نام λ، هیچ خواهر و برادری ندارند.

۲- درخت اشتقاق تولید رشته abbbb :



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

۳- اگر G یک گرامر مستقل از متن باشد، آنگاه هر $w\in L(G)$ یک اشتقاق چپ و یک اشتقاق راست دارد.

۴- ترتیب حذف برابر است با:

الف- قواعد تولید λ ب- قواعد واحد ج- قواعد بی فایده

۵- مراحل کار به صورت زیر است:

۱- حذف قوانین λ	۲- حذف قوانین واحد	۳- حذف قوائد بی فایده
$S \to aA \mid a \mid aBB$ $A \to aaA \mid aa$ $B \to bB \mid bbC$ $C \to B$	$S \to aA \mid a \mid aBB$ $A \to aaA \mid aa$ $B \to bB \mid bbC$ $C \to bB \mid bbC$	$S \to aA \mid a$ $A \to aaA \mid aa$

۶- خیر- به طور نمونه در گرامر $S \to aA; A \to a$ هیچ قانون بی فایده، قانون واحد یا قانون λ به چشم نمی خورد. اما چون $S \to aa$ یکی از گرامرهای متناظر است، گرامر فوق کمینه نمی باشد.

مثالی دیگر: گرامر زیر را در نظر بگیرید:

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

این گرامر زبان $L = \{a^n b^n : n \geq 1\}$ را تولید می کند. اگر روال حذف قوانین بی فایده را روی این گرامر اجرا کنیم، هیچ یک از قواعد حذف نمی شود. از طرفی گرامر زیر که کمینه است، همین زبان را تولید می کند:

$$S \to aSb \mid ab$$

۷- متغیر D و قوانین متناظر با آن بی فایده است.

$$S \to aA \mid a \mid B$$
$$A \to aB \mid \lambda$$
$$B \to Aa$$

۸- گرامر بعد از حذف قوانین λ و بی فایده به صورت زیر در می آید:

$$S \to aaB \mid aB \mid aa \mid a$$
$$B \to aa$$

۹- به جای a و b از متغیرهای M و N استفاده می کنیم:

$$S \to MAN \mid MN$$
$$M \to a$$
$$N \to b$$

به جای AN از X استفاده می کنیم:

$$S \to MX \mid MN$$
$$X \to AN$$
$$M \to a$$
$$N \to b$$

۱۰- ابتدا قانون یکه $S \to A$ را حذف می کنیم:

$$S \to aSaA \mid abA \mid b$$
$$A \to abA \mid b$$

سپس گرامر را به فرم نرمال چامسکی در می آوریم:

$S \to MX \mid MY \mid b$ $X \to SK$ $K \to MA$ $Y \to NA$ $A \to MY \mid b$ $M \to a$ $N \to b$	$S \to MX \mid MY \mid b$ $X \to SMA$ $Y \to NA$ $A \to MY \mid b$ $M \to a$ $N \to b$	$S \to MSMA \mid MNA \mid b$ $A \to MNA \mid b$ $M \to a$ $N \to b$

۱۱- فرم نرمال گریباخ:

$$S \to aSX \mid aX$$
$$X \to b$$

۱۲- فرم نرمال گریباخ:

$$S \rightarrow aSX \mid aS \mid aYS$$
$$X \rightarrow b$$
$$Y \rightarrow a$$

۱۳- می‌توان عضویت یا عدم عضویت یک رشته در زبانی که بوسیله گرامری که در فرم نرمال چامسکی تولید می شود را تصمیم گیری کرد. پیچیدگی این الگوریتم $|w|^3$ می‌باشد. این الگوریتم مساله را به بخش های کوچکتر تقسیم می کند. روش کار در کتاب لینز آورده شده است.

۱۴- ارتفاع در محدوده $\log_{|v|}^{|w|} \leq h \leq \frac{|w|-1}{|v|-1}$ است.

۱۵- حداکثر تعداد قانون تولید برابر است با: $(k-1)|P|+|T|$

۱۶- از فرمی به نام BNF برای تعریف گرامرها استفاده می شود.

$$<\text{expression}>::=<\text{term}> \mid <\text{expression}> + <\text{term}>$$
$$<\text{term}>::=<\text{factor}> \mid <\text{term}> * <\text{factor}>$$

فصل ۶

اتوماتای پشته‌ای

ماشین پشته‌ای PDA (Push Down Automaton) علاوه بر اجزاء ماشین متناهی(FA) ، دارای یک پشته نیز می‌باشد. این ماشین، علاوه بر خواندن نوار ورودی از چپ به راست، قادر به نوشتن هر تعداد نماد درون یک پشته (push) و بازیابی آنها (pop) می‌باشد. ماشین‌های پشته‌ای پذیرنده زبانهای مستقل از متن می‌باشند. ماشین پشته ای از ماشین متناهی قویتر است، یعنی زبان های بیشتری را می پذیرد. اتوماتای پشته ای بر دو نوع معین (DPDA) و نامعین(NPDA) است.

اتوماتای پشته ای نامعین

پذیرنده پشته‌ای نامعین بوسیله هفت تایی $M = (Q, \Sigma, \Gamma, \delta, q_0, z, F)$ تعریف می‌شود:

Q : مجموعه متناهی از حالت های داخلی واحد کنترل

Σ : الفبای ورودی

Γ : الفبای پشته

δ : تابع انتقال

q_0 : حالت شروع واحد کنترل

z : سمبل ته پشته

F : مجموعه حالات پایانی

تعریف : زبان مورد پذیرش توسط یک npda ، مجموعه تمام رشته هایی است که npda با رسیدن به انتهای آن رشته ها، مـی تواند در حالت پایانی قرار گیرد.

تابع انتقال

قانون $\delta(q_i, a, A) = (q_j, B)$ ، یعنی اگر ماشین در وضعیت q_i باشد و a مقابل هد نوار خـوان باشـد و A بـالای پشـته باشد، آنگاه ماشین به حالت q_j تغییر وضـعیت داده و A را از پشـته pop کـرده و B را push می‌کنـد. تـابع انتقـال در ماشین پشته‌ای نامعین به صورت $Q\times(\sum\bigcup\{\lambda\})\times\Gamma\rightarrow Q\times\Gamma^*$ بیان می شود.

یعنی این تابع ۳ ورودی و دو خروجی دارد. ورودی‌های آن به ترتیب برابرند با: حالت جاری ماشـین، سـمبل خوانـده شـده از نوار یا λ و سمبل بالای پشته. خروجی های نیز عبارتند از: حالت بعدی ماشین و سمبلی که در بالای پشته push می شود.

تفسیر چند قانون نمونه :

۱- قانون $\delta(q_0, a, 0) = \{(q_1, 1)\}$ یعنی ماشین در وضعیت q_0 با خواندن a روی نوار به وضعیت q_1 رفتـه و 0 از بـالای پشته pop شده و 1 در پشته push می شود. یعنی به جای 0 مقدار 1 قرار می دهیم.

۲- قانون $\delta(q_0, a, 0) = \{(q_1, 10)\}$ یعنی ماشین در وضعیت q_0 با خواندن a روی نوار به وضعیت q_1 رفتـه و 0 بـالای پشته با 10 جایگزین می شود. در واقع 1 در بالای پشته push می شود.

۳- قانون $\delta(q_0, a, 0) = \{(q_1, \lambda)\}$ یعنی ماشین در وضعیت q_0 با خواندن a روی نوار به وضعیت q_1 رفتـه و 0 بـا λ جایگزین می‌شود. یعنی 0 از بالای پشته pop می شود.

مثال

نحوه عملکرد $\delta(q_0, a, z) = \{(q_1, 1z)\}$ را تشریح کنید.

حل: هرگاه واحد کنترل در حالت q_0 قرار گیرد و سمبل ورودی خوانده شده a و سمبل واقع در بالای پشـته z باشـد، آنگـاه واحد کنترل به حالت q_1 رفته و 1 در پشته push می شود.■

مثال

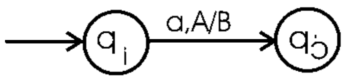
نحوه عملکرد $\delta(q_0, a, 0) = \{(q_1, 1), (q_2, \lambda)\}$ را تشریح کنید.

حل: هرگاه واحد کنترل در حالت q_0 قرار گیرد و سمبل ورودی خوانده شده a و سمبل واقع در بالای پشته 0 باشد، آنگاه یکی از دو حالت زیر اتفاق می افتد:

۱- واحد کنترل به حالت q_1 رفته و 0 از پشته pop شده و 1 به جای آن push می شود.

۲- واحد کنترل به حالت q_2 رفته و سمبل 0 از بالای پشته حذف می شود.

گراف انتقال برای $\delta(q_i, a, A) = (q_j, B)$ به صورت زیر می باشد:



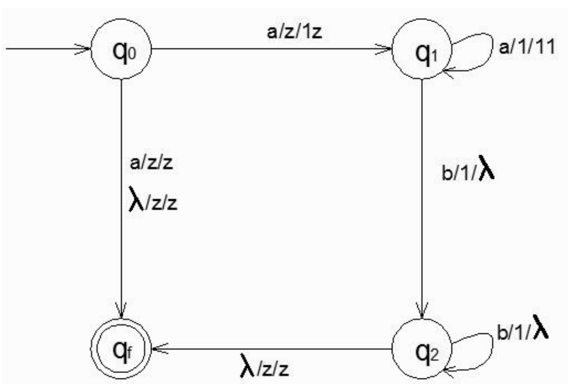
مثال

ماشین npda با انتقالات زیر چه زبانی را می پذیرد؟

$$\sum = \{a, b\}, \Gamma = \{0,1\}$$

$$\delta(q_0, a, z) = \{(q_1, 1z), (q_f, z)\}$$
$$\delta(q_0, \lambda, z) = \{(q_f, z)\}$$
$$\delta(q_1, a, 1) = \{(q_1, 11)\}$$
$$\delta(q_1, b, 1) = \{(q_2, \lambda)\}$$
$$\delta(q_2, b, 1) = \{(q_2, \lambda)\}$$
$$\delta(q_2, \lambda, z) = \{(q_f, z)\}$$

حل: شکل ماشین به صوت زیر است:



در صورتی npda به حالت نهایی می رود که رشته ورودی جزء زبان $L = \{a^n b^n : n \geq 0\} \bigcup \{a\}$ باشد. با ذکر دو مثال این ماشین را بررسی می کنیم:

۱- اگر ورودی aabb باشد، با خواندن اولین a به q_1رفته و عدد 1 در پشته push می کند. سپس با خواندن a بعدی، عدد 1 دیگری در پشته push کرده و در q_1 باقی می‌ماند. با خواندن b ، عدد 1 بالای پشته pop شده و به حالت q_2 می رود. با خواندن b بعدی، عدد 1 دیگری نیز pop شده و در q_2 باقی می‌ماند. در نهایت چون رشته ورودی تمام شده و پشته خالی است، به حالت نهایی q_f می‌رود.

بنابراین توسط انتقال $\delta(q_1,a,1)=\{(q_1,11)\}$ پس از خواندن یک سمبل a از ورودی، یک سمبل 1 به پشته اضافه می شود و در انتقال $\delta(q_2,b,1)=\{(q_2,\lambda)\}$، به محض برخورد با یک b از ورودی، یک سمبل1 از پشته حذف می کند. به کمک این دو انتقال، تعداد a ها شمارش شده و با تعداد b ها تطبیق داده می‌شود.

۲- اگر ورودی a باشد، ماشین از q_0 به حالت نهایی q_f می رود.

علت نامعین بودن این ماشین این است که در حالت q_0، با خواندن a، ماشین هم می تواند به q_1 برود و هم به q_3 .

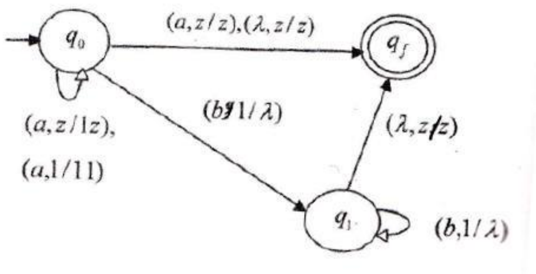
توجه: در یکی از شرایط زیر ورودی پذیرفته نمی‌شود:

۱- اگر زمانی که هنوز تعدادی b در ورودی مانده باشد و پشته خالی باشد.

۲- زمانی که b ها تمام شده و پشته خالی نباشد.

۳- در ورودی بعد از b ها، نماد a ظاهر شود.

این ماشین را با ۳ حالت به صورت زیر نیز می توان ساخت.



مثال

ماشین npda با انتقالات زیر چه زبانی را می پذیرد؟

$$\delta(q_0,a,z)=\{(q_0,0z)\}$$

$\delta(q_0,b,z)=\{(q_0,1z)\}$

$\delta(q_0,a,1)=\{(q_0,\lambda)\}$

$\delta(q_0,b,0)=\{(q_0,\lambda)\}$

$\delta(q_0,a,0)=\{(q_0,00)\}$

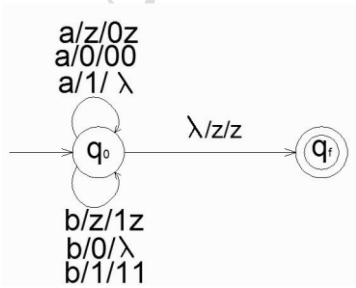
$\delta(q_0,b,1)=\{(q_0,11)\}$

$\delta(q_0,\lambda,z)=\{(q_f,z)\}$

حل:

این ماشین زبان $L=\{w\in\{a,b\}^*:n_a(w)=n_b(w)\}$ را می پذیرد. یعنی رشته هایی که تعداد a ها با تعداد b هـا در آن برابر است ولی فرقی ندارد با a شروع شود یا با b. به طور نمونه اگر رشته ورودی ba باشد، ماشین با خوانـدن b ، 1 در پشته push می کند. با خواندن a ، مقدر 1 بالای پشته را pop می کند. در این حالت چـون ورودی تمـام شـده و بـالای پشته z است، به حالت نهایی می رود.

شکل این ماشین به صورت زیر است:



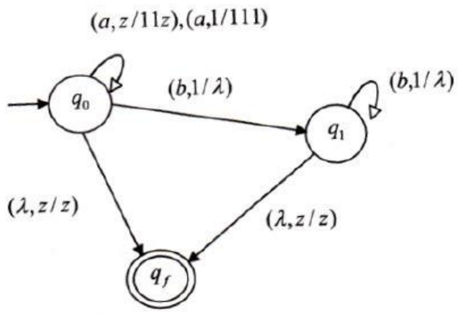
برای جواب دادن به این نوع سئوال ها، نیازی به رسم شکل نیست و برای فهم بهتر شکل رسم شد. ■

مثال

npda ای بسازید که زبان $L=\{a^nb^{2n}:n\geq 0\}$ را روی الفبای $\Sigma=\{a,b\}$ پذیرش کند؟

حل: با دیدن هر a ، دو تا 1 در پشته push کرده و با دیدن هر b ، یک عدد 1 از پشته pop می کنیم.

$\delta(q_0, a, z) = \{(q_1, 11z)\}$
$\delta(q_0, a, 1) = \{(q_1, 111)\}$
$\delta(q_0, \lambda, z) = \{(q_f, z)\}$
$\delta(q_0, b, 1) = \{(q_1, \lambda)\}$
$\delta(q_1, b, 1) = \{(q_1, \lambda)\}$
$\delta(q_1, \lambda, z) = \{(q_f, z)\}$

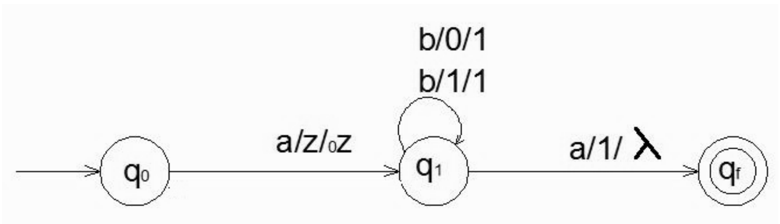


مثال

ماشین npda با انتقالات زیر چه زبانی را می پذیرد؟

$\delta(q_0, a, z) = \{(q_1, 0z)\}$
$\delta(q_1, b, 0) = \{(q_1, 1)\}$
$\delta(q_1, b, 1) = \{(q_1, 1)\}$
$\delta(q_1, a, 1) = \{(q_f, \lambda)\}$

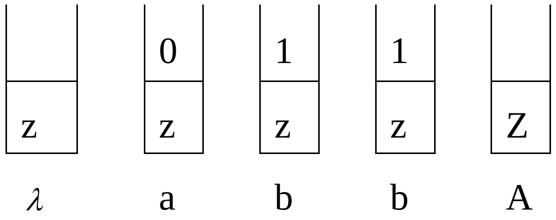
حل: شکل ماشین به صورت زیر است :



زبان مورد نظر ab^+a می باشد.

در این ماشین نیازی به تطبیق دادن تعداد a با b نیست و فقط باید توالی چک شود. یعنی باید ابتدا یک a ، سپس یـک یـا چند b و در نهایت یک a بیآید.

به طور نمونه اگر رشته ورودی abba باشد، با خواندن اولین a، عدد 0 را push کرده و با خواندن b ، عـدد 0 را pop و 1 را Push می‌کند. با خواندن b بعدی تغییری در پشته نمی‌افتد. در نهایت با خواندن a ، عدد 1 بالای پشته pop شـده و به حالت نهایی می رود:



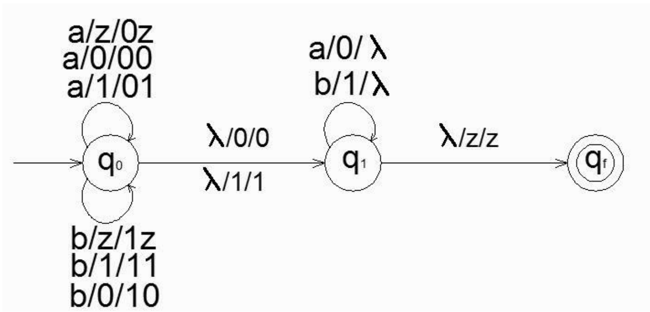
مثال

ماشین npda با انتقالات زیر چه زبانی را می پذیرد؟

$$\delta(q_0,a,z)=\{(q_0,0z)\} \quad \delta(q_0,a,0)=\{(q_0,00)\} \quad \delta(q_0,a,1)=\{(q_0,01)\}$$
$$\delta(q_0,b,z)=\{(q_0,1z)\} \quad \delta(q_0,b,0)=\{(q_0,10)\} \quad \delta(q_0,b,1)=\{(q_0,11)\}$$
$$\delta(q_1,a,0)=\{(q_1,\lambda)\} \quad \delta(q_1,b,1)=\{(q_1,\lambda)\} \quad \delta(q_1,\lambda,z)=\{(q_f,z)\}$$
$$\delta(q_0,\lambda,0)=\{(q_1,0)\} \quad \delta(q_0,\lambda,1)=\{(q_1,1)\}$$

حل:

شکل این ماشین به صورت زیر است:



این ماشین زبان $L=\{ww^R : w\in\{a,b\}^+\}$ را می پذیرد. رشته abba، نمونه‌ای از رشته‌های این زبان است. این رشته از دو بخش w و w^R تشکیل شده است. برای پی بردن به نحوه عملکرد این ماشین تصور کنید که در وسط رشته، λ قرار دارد. (مثلا $ab\lambda ba$).

توسط قوانین $\delta(q_0,\lambda,1)=\{(q_1,1)\}$, $\delta(q_0,\lambda,0)=\{(q_1,0)\}$ می‌توان بدون خواندن ورودی و تغییر در پشته، فقط از حالت q_0 به q_1 رفت. از این ویژگی برای تشخیص وسط رشته استفاده شده است.

ماشین مورد نظر با دیدن a در ورودی، عدد 0 و با دیدن b عدد 1 را در پشته push می‌کند. نکته این ماشین در این است که نباید تا انتهای رشته ورودی این عمل را انجام داد و باید در وسط رشته، از حالت q_0 به q_1 رفت. در حالت q_1 با دیدن a، 0 بالای پشته pop می شود و با دیدن b، 1 بالای پشته pop می شود. بعد از پایان ورودی، در صورتی که پشته خالی باشد، از q_1 به q_f رفته و رشته پذیرفته می شود.■

پیکربندی لحظه ای

پیکربندی لحظه‌ای یک اتومات پشته‌ای، سه تایی (q,w,u) است که در آن، q حالت جاری واحد کنترل، w بخش خوانده نشده رشته ورودی و u محتویات پشته است(چپ ترین سمبل بیانگر بالای پشته است). انتقال از یک پیکربندی به پیکربندی دیگر را بوسیله $\mapsto$ نمایش می دهند.

به طور نمونه دنباله تغییرات پیکربندی برای پذیرش رشته abba در ماشین پذیرنده زبان $L=\{ww^R : w\in\{a,b\}^+\}$ به صورت زیر است:

$$(q_0,abba,z)\mapsto(q_0,bba,0z)\mapsto(q_0,ba,10z)\mapsto(q_1,ba,10z)\mapsto(q_1,a,0z)$$
$$\mapsto(q_1,\lambda,z)\mapsto(q_f,z).$$

توجه شود که در حرکت سوم از ویژگی نامعین بودن اتومات برای تشخیص وسط رشته استفاده شد و بدون خواندن ورودی و تغییر در پشته، فقط از حالت q_1 به حالت q_f رفت.

اتوماتای پشته ای معین

اتومات پشته‌ای $M=(Q,\sum,\Gamma,\delta,q_0,z,F)$ معین گفته می شود اگر هم در تعریف npda صدق کند و همچنین دارای محدودیت هایی به این شرح باشد که به ازای هر $a\in\sum\bigcup\{\lambda\}\ ,\ b\in\Gamma$:

۱- $\delta(q,a,b)$ حداکثر یک عضو داشته باشد.

۲- اگر $\delta(q,\lambda,b)$ تهی نباشد، آنگاه $\delta(q,c,b)$ باید به ازای هر $c\in\sum$ تهی باشد.

اولین شرط فوق صرفا مستلزم آن است که به ازای هر سمبل ورودی مفروض و هر عنصر بالای پشته، حداکثر یک حرکت قابل انجام باشد. بر اساس شرط دوم، چنانچه در یکی از پیکربندی های مفروض به یک انتقال λ برخورد کنیم، آنگاه هیچ حرکتی برای جلو بردن و مصرف ورودی امکان پذیر نمی‌باشد. به طور مثال اگر قانون $\delta(q_0,\lambda,z)=\{(q_f,z)\}$ داشته باشیم، آنگاه نباید قانونی مانند $\delta(q_0,a,z)=\{(q_0,0z)\}$ وجود داشته باشد.

- هیچ هم ارزی بین اتوماتای پشته‌ای معین و نامعین وجود ندارد. (بالعکس اتوماتای متناهی)

- هر PDA قطعی، می تواند $\lambda-$ transition داشته باشد. (بدون دریافت ورودی، بتواند تغییر حالت دهد.)

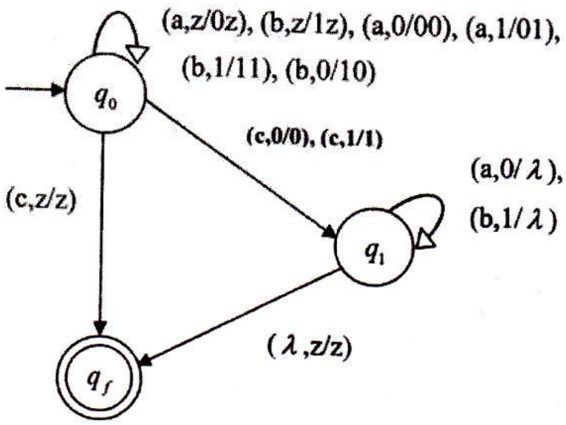
زبان مستقل از متن معین

زبان L یک زبان مستقل از متن معین(قطعی) نامیده می‌شود اگر و تنها اگر یک dpda به نام M وجود داشته باشد که L=L(M).

مثال

آیا زبان مستقل از متن $L=\{wcw^R : w\in\{a,b\}^*\}$ ، معین است؟

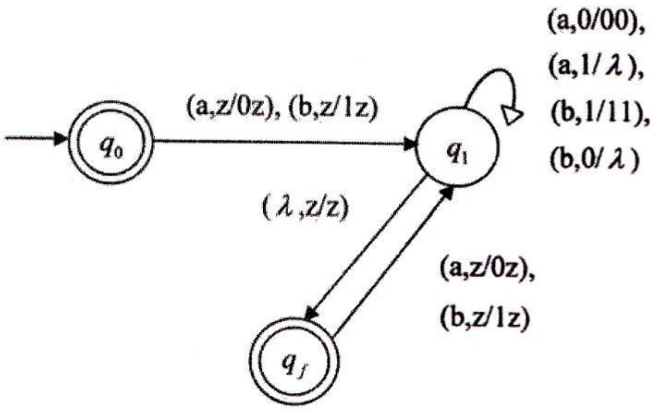
حل: بله- چون می توان یک dpda برای آن ساخت.



مثال

آیا زبان مستقل از متن $L=\{w\in\{a,b\}^* : n_a(w)=n_b(w)\}$ ، معین است؟

حل: بله- چون می توان برای آن یک dpda طراحی کرد:



■

مثال

آیا زبان مستقل از متن $L=\{a^nb^n : n\geq 0\}\cup\{a^nb^{2n} : n\geq 0\}$، معین است؟

حل: خیر- چون باید یک یا دو b را با هر یک از a ها تطابق دهد. اطلاعات موجود در ابتدای رشته چنان نیست که بتوان به کمک آن انتخاب خود را قطعی کرد. ■

مثال

آیا زبان مستقل از متن $L=\{ww^R : w\in\{a,b\}^*\}$ ، معین است؟

حل- خیر- چون وسط رشته باید با آزمایش و خطا حدس زده شود. ■

مثال

آیا زبان مستقل از متن $L=\{a^nb^n : n\geq 1\}\cup\{a\}$ ، معین است؟

حل: بله- چون می توان برای آن یک dpda طراحی کرد:

$$\delta(q_0,a,z)=\{(q_3,1z)\}$$
$$\delta(q_3,a,1)=\{(q_1,11)\}$$
$$\delta(q_1,a,1)=\{(q_1,11)\}$$
$$\delta(q_1,b,1)=\{(q_1,\lambda)\}$$
$$\delta(q_1,\lambda,z)=\{(q_2,z)\}$$

که q_2 و q_3 حالت‌های پایانی هستند.

در نگاه اول ممکن است فکر کنید که این زبان نامعین است. اما چون توانستیم یک dpda برای آن بسازیم، این زبان معین است. اگر اولین سمبل ورودی یک a باشد، ماشین به یکی از حالات پایانی رفته و در صورت وجود سمبل‌های بعدی، از این حالت خارج شده و a^nb^n را می پذیرد.

چند نکته:

۱- خانواده زبان‌های مستقل از متن معین، تحت اجتماع ، اشتراک و معکوس بسته نیستند.

۲- زبان‌های مستقل از متن معین، تحت عمل مکمل گیری بسته می باشند.

۳- خانواده زبان‌های مستقل از متن معین، تحت تفاضل منظم بسته است.

۴- اجتماع یک زبان مستقل از متن معین با یک زبان منظم، مستقل از متن معین می باشد.

چند نکته:

۱- تمامی زبان‌های منظم، مستقل از متن معین می باشند. چون هر زبان منظمی را می توان با یک dfa پذیرفت و چنین dfa یک dpda با پشته استفاده نشده است.

۲- زبان‌های مستقل از متن معین، هیچگاه ذاتا مبهم نیستند.

۳- به ازای هر npda به نام M،یک npda به نام $\widehat{M}$ حداکثر ۳ حالتی وجود دارد، بطوریکه:

$$L(M) = L(\widehat{M})$$

البته می توان تعداد حالت ها را به ۲ حالت کاهش داد. (با قرار دادن یک سمبل در پشته)

ساخت اتوماتای پشته ای با استفاده از گرامر در فرم گریباخ

می توان با داشتن گرامر در فرم گریباخ برای یک زبان مستقل از متن، ماشین پشته ای برای آن طراحی کرد.

مثال

pda ای بسازید که زبان تولید شده بوسیله گرامر $S \to aSbb \mid a$ را پذیرش کند.

حل: ابتدا گرامر را به فرم گریباخ در می آوریم:

$$S \to aSA \mid a$$
$$A \to bB$$
$$B \to b$$

مراحل تبدیل گرامر به ماشین:

الف- قرار دادن متغیر شروع روی z :

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

ب- شبیه سازی قوانین:

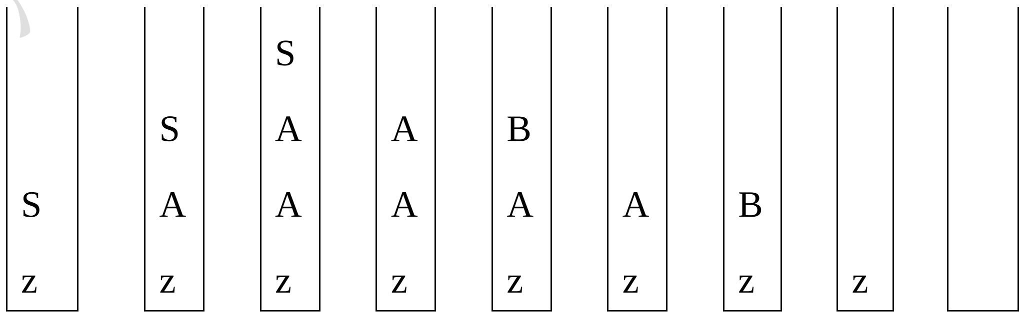
$$S \to aSA \mid a \quad : \quad \delta(q_1, a, S) = \{(q_1, SA), (q_1, \lambda)\}$$
$$A \to bB \quad : \quad \delta(q_1, b, A) = \{(q_1, B)\}$$
$$B \to b \quad : \quad \delta(q_1, b, B) = \{(q_1, \lambda)\}$$

ج- با وقوع z در ته پشته، کار اشتقاق تمام شده و ماشین بوسیله دستور زیر در حالت پایانی قرار می گیرد:

$$\delta(q_1, \lambda, z) = \{(q_f, \lambda)\}$$

این ماشین پشته ای زبان $L = \{a^{n+1}b^{2n} : n \ge 0\}$ را می پذیرد.

نحوه پذیرش رشته a^3b^4 توسط ماشین:



λ	a	a	a	b	b	b	b	λ

مثال

یک npda برای زبان $L=\{a^n b^{n+1} : n \ge 0\}$ پیدا کنید.

حل: گرامر آن به فرم $S \to aSb \mid b$ است. فرم گریباخ این گرامر برابر است با:

$$S \to aSB \mid b$$
$$B \to b$$

پس یک ماشین پشته ای با سه حالت q_0 و q_1 و q_f می سازیم. در صورتی می‌توان حالت q_1 را حذف کرد که از یک سمبل ویژه پشته مانند z_1 برای نشانه گذاری آن استفاده کرد:

$$\delta(q_0,\lambda,z)=\{(q_0,Sz_1)\}$$
$$\delta(q_0,a,S)=\{(q_0,SB)\}$$
$$\delta(q_0,b,S)=\{(q_0,\lambda)\}$$
$$\delta(q_0,b,B)=\{(q_0,\lambda)\}$$
$$\delta(q_0,\lambda,z_1)=\{(q_f,\lambda)\}.$$

■

مثال

یک npda ای طراحی کنید که زبان تولید شده توسط گرامر $S \to aSbb \mid aab$ را بپذیرد.

حل: ابتدا گرامر $S \to aSbb \mid aab$ را به فرم گریباخ در می آوریم:

$$S \to aSA \mid aD$$
$$A \to bB$$
$$B \to b$$
$$D \to aB$$

سپس npda آن را طراحی می‌کنیم:

$\delta(q_0, \lambda, z) = \{(q_1, Sz)\}$
$\delta(q_1, a, S) = \{(q_1, SA)\}$
$\delta(q_1, a, S) = \{(q_1, D)\}$
$\delta(q_1, a, D) = \{(q_1, B)\}$
$\delta(q_1, b, A) = \{(q_1, B)\}$
$\delta(q_1, b, B) = \{(q_1, \lambda)\}$
$\delta(q_1, \lambda, z) = \{(q_f, z)\}$

■

تشخیص مستقل از متن بودن یک زبان

یکی از روش های تشخیص مستقل از متن بودن یک زبان، امکان طراحی یک ماشین پشته ای برای آن زبان است. قبلا دیدیم که روش دیگر این است که بتوان برای آن زبان، یک گرامر مستقل از متن نوشت.

مثال

آیا $L = \{a^n b^m c^n d^m : m,n \geq 0\}$ مستقل از متن است؟

حل: خیر- ترتیب قرار گرفتن نمادها طوری است که نمی‌توان توسط یک پشته، تساوی تعداد a ها با تعداد c ها و همچنین تعداد b ها با تعداد d ها را کنترل کرد.■

مثال

آیا $L = \{a^n b^n c^m d^m : m \geq n \geq 0\}$ مستقل از متن است؟

حل: خیر- می توان تساوی تعداد نمادهای a با b و تعداد c با d را چک کرد. اما با همان پشته دیگر نمی‌توان شرط $m \geq n$ را نیز چک کرد.■

مثال

آیا زبان $\{a^{3n} b^{2n} a^{5n} : n \in N\}$ مستقل از متن است؟

حل: خیر- بعد از تطبیق تعداد a ها با b ها ، مقدار n فراموش شده و نمی توان تعداد a در a^{5n} را چک کرد.■

گرامر LL

گرامرها در تمام جنبه‌های نحو زبان‌های برنامه سازی موفق نمی باشند. با تعمیم این گرامرها بطوری که قابلیت تجزیه را از دست بدهند، به گرامر LL می‌رسیم. L اول به معنای پویش از چپ به راست و L دوم به معنای ساخت اشتقاق های چپ ترین است.در این گرامرها، با نگاه به بخش محدودی از ورودی شامل سمبل خوانده شده بعلاوه تعداد متناهی از سمبل‌های بعد از آن، به راحتی قانون مناسب را می‌توان تعیین کرد.

مثال

گرامر $S \to aSb \mid ab$ یک $LL(2)$ است، چون با در اختیار داشتن سمبل جاری خوانده شده و پیش بینی سمبل بعدی، می توان قانون درست را انتخاب کرد. اگر سمبل خوانده شده a و سمبل بعدی b باشد، از قانون $S \to ab$ و در غیر اینصورت از $S \to aSb$ باید استفاده کرد. ■

مثال

یک LL گرامر برای زبان $L = \{a^n b^m c^{n+m} : n \geq 0, m \geq 0\}$ بسازید.

حل:

$$S \to aSc \mid X \mid \lambda$$
$$X \to bXc \mid \lambda$$

این گرامر $LL(1)$ است. چون اگر سمبل جاری خوانده شده a باشد، از $S \to aSc$، اگر b باشد از $S \to X$ و اگر c باشد از $S \to \lambda$ باید استفاده کرد.■

مثال

یک LL گرامر برای زبان $L(a^*ba) \cup L(abbb^*)$ بسازید.

حل:

$S \to S_1 \mid S_2$
$S_1 \to aS_1 \mid ba$
$S_2 \to abbB$
$B \to bB \mid \lambda$

این گرامر $LL(3)$ است. چون اگر سه سمبل اول، aaa یا aab یا aba باشد، آنگـاه رشـته در a^*ba و اگـر abb باشد، رشته در $abbb^*$ وجود دارد. ■

چند نکته:

۱- گرامری که ساده باشد، حتما LL نیز می باشد.

۲- گرامر LL ، مبهم نمی باشد.

۳- زبان گرامری LL(k)، مستقل از متن معین است.

۴- یک گرامر به فرم نرمال گریباخ، در صورتی که LL باشد، الزاما LL(1) خواهد بود.

تمرین فصل ۶

۱- کدام زبان مستقل از متن نیست؟

$L_1 = \{a^n b^j a^k a^m : n + j \le k + m\}$
$L_2 = \{a^n b^j a^k b^m : n \le k, j \le m\}$

۲- آیا زبان $L = \{a^i b^j c^k d^t a^t b^k c^j d^i : i, j, k, t \ge 0\}$ مستقل از متن است؟

۳- آیا زبان $L = \{a^n b^m c^{m+3} d^p e^{n+p} : n, m, p \ge 0\}$ مستقل از متن است؟

۴- کدام جمله صحیح است؟

الف) اگر یک PDA هیچگاه عمل POP کردن را انجام ندهد آنگاه زبانی که می پذیرد منظم است.

ب) مجموعه زبان‌های پذیرفته شده توسط NPDA از DPDA بیشتر است.

۵- در یک اتومات پوش دان(Push Down) طول پشته حداکثر 6 است. زبانهایی که این اتومات می تواند بپذیرد، در کدام مجموعه زبان قرار می گیرد؟

۶- یک PDA با n حالت برای محاسبه کلیه ورودی ها حداکثر از 2^n سلول از حافظه پشته خود استفاده می کند. آیا برای زبان این ماشین یک گرامر منظم وجود دارد؟

۷- آیا هر زبان مستقل از متن، دارای یک PDA متناظر با پشته متناهی است؟

۸- اتوماتون یک PDA ، فاقد انتقال بلادرنگ ($\lambda - transition$) است. آیا زمان محاسبه هر ورودی، متناهی است؟

۹- آیا یک PDA دلخواه در هر مسیر محاسبه برای ورودی w ، فقط از مقدار متناهی از حافظه پشته خود استفاده می کند؟

۱۰- هر زبان مستقل از متن می‌تواند با یک PDA با حداکثر چند حالت توصیف شود؟

۱۱- آیا زبان $L = \{w \in \{0,1\}^* : n_0(w) = f(n_1(w))\}$ مستقل از متن است؟

$$f(x) = \begin{cases} 3 & x \ \ is \ \ odd \\ x+1 & x \ \ is \ \ even \end{cases}$$

۱۲- آیا زبان $L = \{w \in \{0,1\}^* : |w|_0 = f(|w|_1)\}$ که در آن $f(n) = n \bmod 3$ است، مستقل از متن است؟
($|w|_i$ تعداد حروف i در کلمه w باشد)

۱۳- زبان $L = \{a^n b^n c^n : n \geq 0\}$ مفروض است. آیا زبان $h(L)$ مستقل از متن است؟

$$h(L): \quad h(a) = a, \quad h(b) = bb, \quad h(c) = a$$

۱۴- کدام یک از زبان‌های زیر مستقل از متن می باشند؟

$\{a^n b^n c^m d^m : m \geq 0, n \geq 0\}$

$\{a^m b^k c^m : m, k \geq 0\}$

$\{a^n b^{2m} c^m : n, m \geq 0\}$

$\{a^m b^m c^n d^n : n, m \geq 0\}$

$\{a^n b^m : n = 3m\}$

$\{a^n b^m c^n : m, n \geq 0\}$

$\{a^n b^m c^m d^n : m, n \geq 0\}$

۱۵- برای کدام یک از زبان‌های زیر، می توان یک ماشین پشته ای طراحی کرد؟ $x, y \in \{a, b\}^*$

$L_1 = \{xy : x \neq y\}$

$L_2 = \{xy : |x| \neq |y|\}$

$L_3 = \{xy : |x| = |y|, x \neq y\}$

$L_4 = \{xy : x = y^R\}$

$L_5 = \{xy : x = y\}$

۱۶- نشان دهید که تمامی زبان‌های منظم، مستقل از متن معین هستند.

۱۷- نشان دهید که زبان های مستقل از متن معین، هیچگاه ذاتا مبهم نیستند.

۱۸- یک زبان مستقل از متن معین ارائه دهید که معکوس آن نامعین باشد.

۱۹- اگر در یک npda ،در هر حرکت، طول پشته حداکثر به اندازه یک سمبل افزایش یابد، آنگاه دستور $\delta(q_i, a, b) = \{(q_j, cde)\}$ را چگونه باید پیاده سازی کرد؟

۲۰- برای هر یک از زبان‌های زیر، ماشین پشته ای طراحی کنید.

$L_1 = L(aa^+b)$
$L_2 = L(ab^+aba^*)$

۲۱- برای هر یک از زبان‌های زیر، ماشین پشته ای طراحی کنید.

$L_1 = \{a^n b^m c^{n+m} : n \geq 0, m \geq 0\}$

$L_2 = \{a^n b^{n+m} c^m : n \geq 0, m \geq 0\}$

$L_3 = \{a^3 b^n c^n : n \geq 0\}$

$L_4 = \{a^n b^m : n \geq 0, n \neq m\}$

۲۲- برای هر یک از زبان‌های زیر، ماشین پشته ای طراحی کنید.

$L_1 = \{w \in \{a,b\}^* : n_a(w) = n_b(w) + 1\}$

$L_2 = \{w \in \{a,b,c\}^* : n_a(w) + n_b(w) = n_c(w)\}$

$L_3 = \{w \in \{a,b\}^* : wcw^R\}$

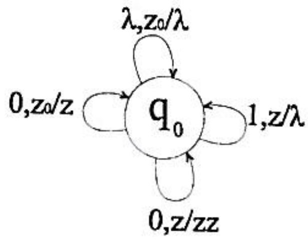
$L_4 = \{w \in \{a,b\}^* : 2n_a(w) \leq n_b(w) \leq 3n_a(w)\}$

۲۳- برای زبان زیر، ماشین پشته ای طراحی کنید.

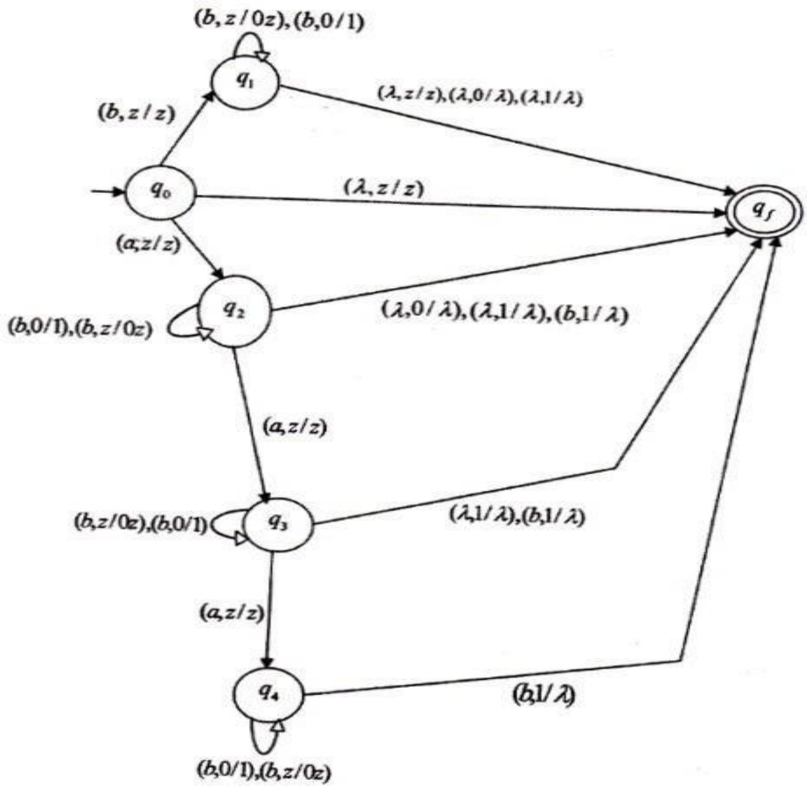
$L = \{ab(ab)^n b(ab)^n : n \geq 0\}$

۲۴- فرض کنید $L = \{a^n b^{2n} : n \geq 0\}$ باشد. یک ماشین پشته ای برای L^* طراحی کنید.

۲۵- اتومات پوش دان $(Q, \Sigma, \Gamma, \delta, q_0, z_0)$ با $Q = \{q_0\}, \Sigma = \{0,1\}, \Gamma = \{z_0, z\}$ مطابق شکل زیر مفروض است. زبان این اتومات را مشخص کنید.



۲۶- ماشین پشته ای زیر، چه زبانی را می پذیرد؟



حل تمرین فصل ۶

۱- زبان L_2 مستقل از متن نمی باشد، چون رشته های آن را نمی توان توسط یک پشته بررسی کرد.

۲- بله- زیرا می توان جملات آن را به کمک یک پشته، بررسی کرد.

۳- زبان داده شده را می توان به صورت زیر نوشت:

$$L = \{a^n b^m c^m c^3 d^p e^p e^n : n, m, p \geq 0\}$$

می توان جملات این زبان را به کمک یک پشته، بررسی کرد. پس مشتقل از متن است.

۴- هر دو گزاره صحیح هستند.

۵- چون طول پشته محدود است، ماشین پشته‌ای مانند ماشین متناهی عمل کرده و فقط زبان های مـنظم را می‌پذیرد.

۶- بله- ماشین PDA ای که برای محاسبه ورودی ها حداکثر از تعدادی مشخص از خانه هـای پشـته(مثلا 2^n)، استفاده کند، معادل DFA است. بنابراین برای زبان این ماشین، یک گرامر منظم وجود دارد.

۷- خیر- هر زبان منظم دارای یک PDA متناظر با پشته متناهی است.

۸- بله- چون اگر انتقال λ نداشته باشد، به ازای هر الفبای رشته، یک انتقال انجام شده و چون طول رشـته متنـاهی اسـت، بنابراین حتما در زمان متناهی پیمایش رشته به پایان می رسد.

۹- خیر- چون ممکن است در حلقه بی افتد و از مقدار نامتناهی از حافظه پشته خود استفاده کند.

۱۰- بله- برای هر زبان مستقل از متن، یک PDA با حداکثر دو حالت وجود دارد.

۱۱- بله- زبان L را می توان به صورت زیر نوشت:

$L = \{w \in \{0,1\}^* : n_0(w) = 3 \ or \ n_0(w) = n_1(w) + 1\}$

تعداد یکها، هر عددی می تواند باشد ولی تعداد صفرها برابر 3 یا یک واحد بیشتر از تعداد یکها است. بنابراین چون می تـوان رشته های این زبان را به کمک یک پشته پذیرفت، زبان مستقل از متن است.

۱۲- از آنجا که باقیمانده تقسیم هر عددی بر 3 ، برابر یکی از اعداد 0 تا 2 می باشد، داریم:

$L = \{w \in \{0,1\}^* : \ |w|_0 = 0 \ or \ |w|_0 = 1 \ or \ |w|_0 = 2\}$

در این زبان تعداد یکها، هر عددی می تواند باشد ولی تعداد صفرها باید یکی از اعداد 0 ،1 و یا 2 باشد. بنـابراین چـون مـی توان رشته های این زبان را به کمک یک پشته پذیرفت، زبان مستقل از متن است.

۱۳- خیر- برای مشخص کردن زبان مربوط به هومورفیسم،کافی است در $\{a^n b^n c^n\}$ بـه جـای a همـان a ، بـه جـای b مقدار bb و به جای c مقدار a را قرار داد. در این صورت خواهیم داشت:

$a^n (bb)^n a^n = a^n b^{2n} a^n$

این زبان مستقل از متن نیست، چون علاوه بر کنترل تعداد نمادهـای a , b در $a^n b^{2n}$ ، بایـد تعـداد a , b در $b^{2n} a^n$ نیـز چک شود که با یک پشته ممکن نیست.

۱۴- همه زبان های داده شده مستقل از متن هستند، چون می توان برای آنها یک ماشین پشته ای رسم کرد.

۱۵- برای همه به غیر از L_5 .

۱۶- هر زبان منظمی را می توان با یک dfa پذیرفت. چنین dfa ای یک dpda با پشته استفاده نشده است.

۱۷- به ازای هر زبان مستقل از متن معین، یک dpda وجود دارد. اگر این dpda به گرامر تبدیل شود،گرامر به دست آمـده، غیر مبهم خواهد بود.

۱۸- معکوس زبان مستقل از متن معین زیر، نامعین است:

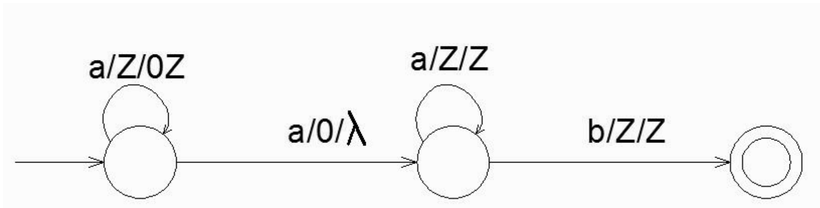
$$L = \{w_1 c w_2, w_1 = w_2^R, w_1, w_2 \in \{a,b,c\}^*\}$$

۱۹- از دو دستور زیر استفاده می کنیم:

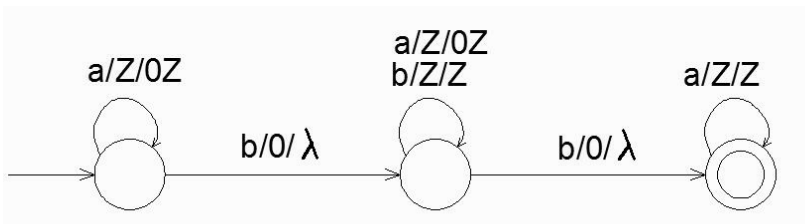
$$\delta(q_i, a, b) = \{(q_k, de)\}$$
$$\delta(q_k, \lambda, d) = \{(q_j, cd)\}$$

۲۰- هر یک از ماشین ها در زیر نشان داده شده است:

ماشین زبان $L_1 = L(aa^+b)$:

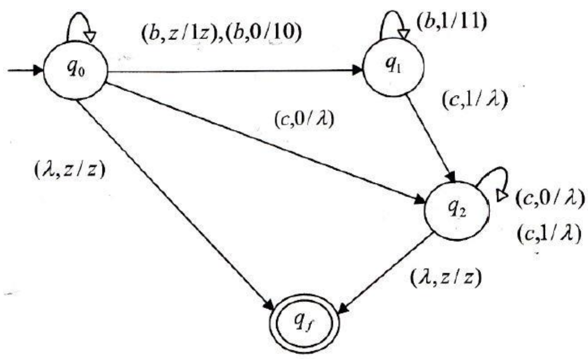


ماشین زبان $L_2 = L(ab^+aba^*)$:

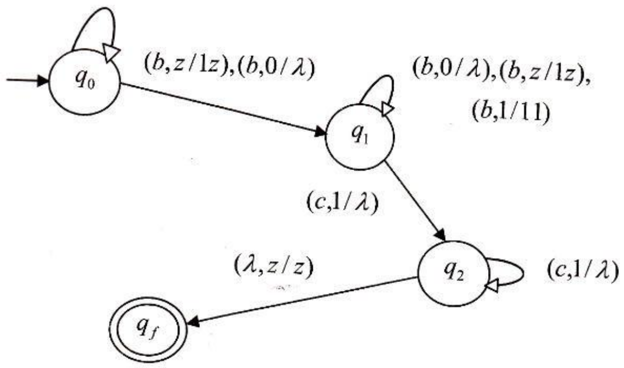


۲۱- هر یک از ماشین ها در زیر نشان داده شده است:

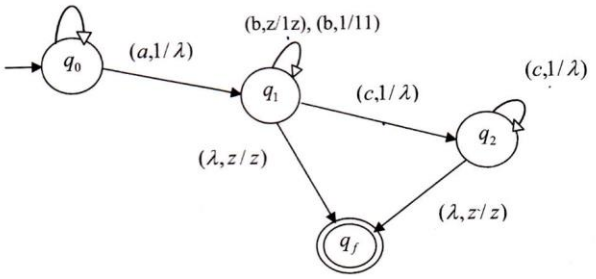
ماشین زبان $L_1 = \{a^n b^m c^{n+m} : n \geq 0, m \geq 0\}$:



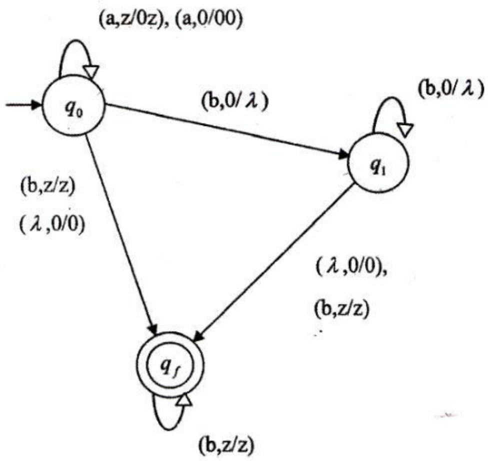
ماشین زبان $L_2 = \{a^n b^{n+m} c^m : n \geq 0, m \geq 0\}$:



ماشین زبان $L_3=\{a^3b^nc^n : n\geq 0\}$:

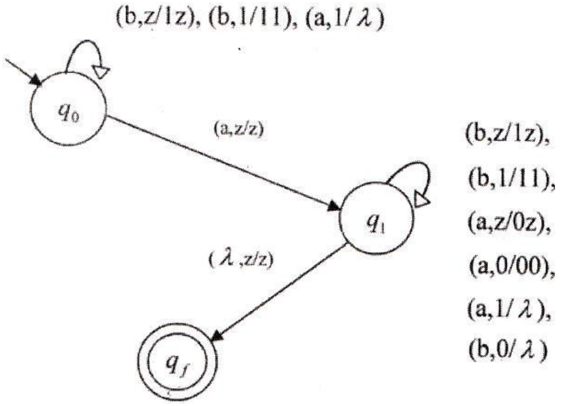


ماشین زبان $L_4=\{a^nb^m : n\geq 0, n\neq m\}$:

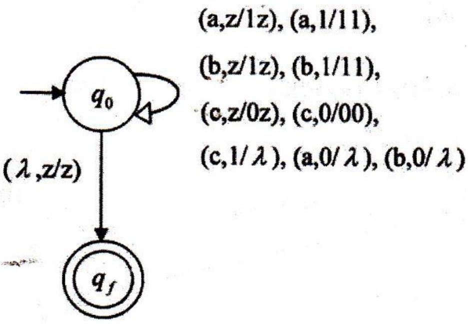


۲۲- هر یک از ماشین ها در زیر نشان داده شده است:

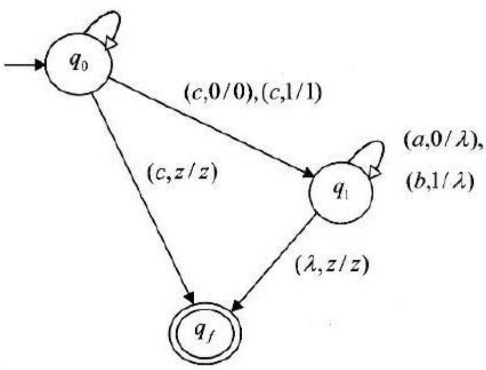
ماشین زبان $L_1 = \{w \in \{a,b\}^* : n_a(w) = n_b(w) + 1\}$:



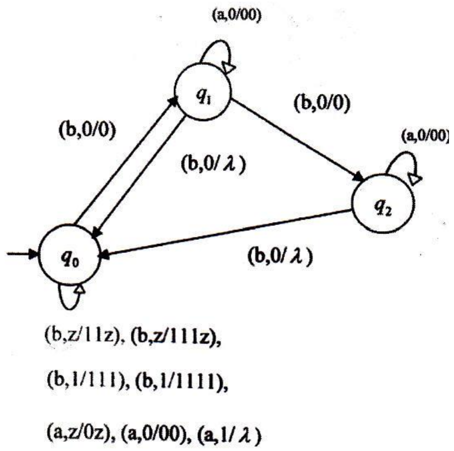
ماشین زبان $L_2 = \{w \in \{a,b,c\}^* : n_a(w) + n_b(w) = n_c(w)\}$:



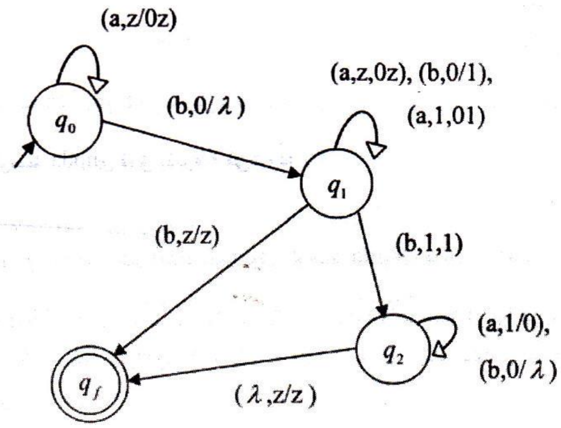
ماشین زبان $L_3 = \{w \in \{a,b\}^* : wcw^R\}$:



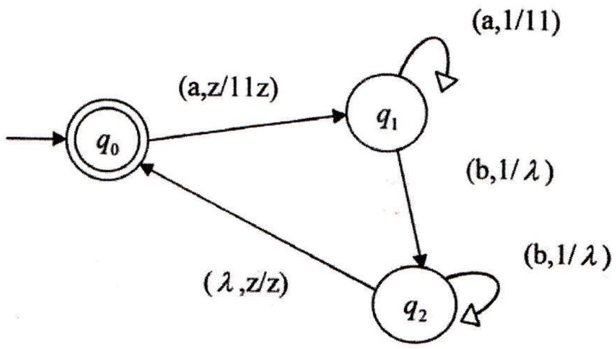
ماشین زبان $L_4 = \{w \in \{a,b\}^* : 2n_a(w) \le n_b(w) \le 3n_a(w)\}$:



۲۳- ماشین به صورت زیر است:



۲۴- ماشین پشته ای به صورت زیر است:



۲۵- زبان ماشین داده شده برابر است با:

$\{w \in (0+1)^* : n_0(w) = n_1(w)\, and\, n_0(v) > n_1(v)\, for\, any\, prefix\, v,\ w = vr, r \in \Sigma^+\}$

۲۶- ماشین داده شده زبان $L = \{a^n b^m : n \leq m \leq 3\}$ را می پذیرد. حالت های ممکن :

الف- اگر n=0 باشد،آنگاه m برابر 0 تا 3 می‌تواند باشد.

ب- اگر n=1 باشد،آنگاه m برابر 1 تا 3 می‌تواند باشد.

پ- اگر n=2 باشد،آنگاه m برابر 2 تا 3 می‌تواند باشد.

ت- اگر n=3 باشد،آنگاه m برابر 3 می‌تواند باشد.

فصل ۷

ماشین تورینگ

ماشین تورینگ یک کامپیوتر ساده دارای واحد پردازشی با حافظه محدود و نواری با ظرفیت نامحدود است. دستوراتی که این ماشین می‌تواند انجام دهد بسیار محدود است. ماشین تورینگ کاملترین مدل برای یک ماشین محاسبه‌گر می باشد و از ماشین‌های متناهی و پشته‌ای کاملتر است. ماشین تورینگ پذیرنده زبانهای منظم، مستقل از متن، وابسته به متن و بدون محدودیت است.

ماشین تورینگ استاندارد

ماشین تورینگ M یک هفت‌تایی است که به صورت $M=(Q,\sum,\Gamma,\delta,q_0,B,F)$ تعریف می‌شود، که در آن:

Q : مجموعه حالات داخلی

$\sum$: الفبای ورودی

Γ : الفبای نوار

δ : تابع انتقال

B : سمبل خالی (متعلق به Γ است و در $\sum$ نیست.)

q_0 : حالت شروع

F : مجموعه حالت های پایانی

نوار ماشین تورینگ به سلول‌هایی تقسیم شده و هر سلول قادر به نگهداری فقط یک سمبل است. به این نوار یک هد خواندن-نوشتن متصل است که می تواند به سمت راست یا چپ نوار حرکت کرده و در هر حرکت فقط یک سمبل را بخواند و بنویسد.

تابع انتقال δ در تورینگ، بـه صـورت $Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$ تعریـف مـی شـود. یعنـی دو ورودی و ۳ خروجی دارد. به عنوان نمونه، وضعیت $\delta(q_0, a) = (q_1, b, R)$ یعنی ماشین در وضعیت q_0 بوده و هد a را می بیند. ماشین به وضعیت q_1 رفته و a با b جایگزین می‌شود و هد به سمت راست می‌رود.

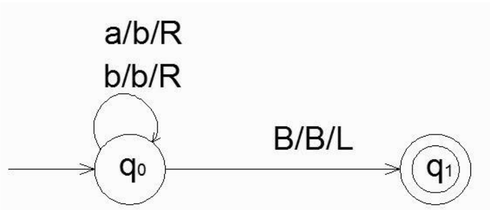
مثال

عملکرد ماشین تورینگ زیر چیست؟

$$Q = \{q_0, q_1\}, \quad \Sigma = \{a, b\}, \quad \Gamma = \{a, b, B\}, \quad F = \{q_1\}$$

$$\delta(q_0, a) = (q_0, b, R)$$
$$\delta(q_0, b) = (q_0, b, R)$$
$$\delta(q_0, B) = (q_1, B, L)$$

حل: نمایش این ماشین به صورت زیر است:



در این ماشین اگر سمبل a، زیر هد قرار بگیرد، یک b جایگزین آن شده و هد به سمت راست حرکت می کند و ماشـین در حالت q_0 باقی می ماند. اگر b زیر هد قرار بگیرد، محتوای نوار هیچ تغییری نکرده و هد به راست می‌رود. ماشین با رسـیدن به اولین سلول خالی(B)، یک سلول به عقب بر می‌گردد و در حالت نهایی q_1 متوقف می‌شود. با فرض اینکه رشته aa بر روی نوار باشد، عملکرد این ماشین را می توان به صورت زیر نمایش داد: $q_0aa \mapsto bq_0a \mapsto bbq_0B \mapsto bq_1b$ ■

ماشین تورینگ در نقش پذیرنده زبان

زبانی که ماشین تورینگ M می‌پذیرد به صورت زیر تعریف می‌شود:

$$L(M) = \{w \in \Sigma^+ : q_f \in F,\ x_1, x_2 \in \Gamma^*,\ q_0 w \overset{*}{\longmapsto} x_1 q_f x_2\}$$

بر اساس این تعریف ورودی w روی نوار نوشته شده و در هر یک از طرفین آن از سمبل فضای خالی استفاده می شود.

اگر w عضو $L(M)$ نباشد، یکی از دو حالت زیر اتفاق می افتد:

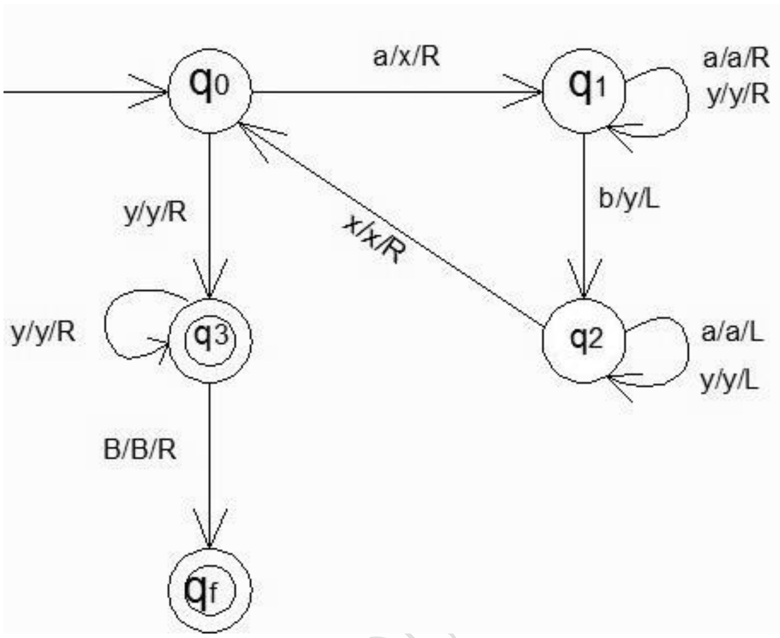
۱- ماشین در یک حالت غیر پایانی متوقف می شود.

۲- ماشین به یک حلقه بی نهایت وارد شده و هرگز متوقف نمی شود.

بنابراین، هر رشته ای که باعث توقف M نشود، عضو $L(M)$ نمی باشد.

مثال

ماشین تورینگ زیر چه زبانی را می پذیرد؟



حل:

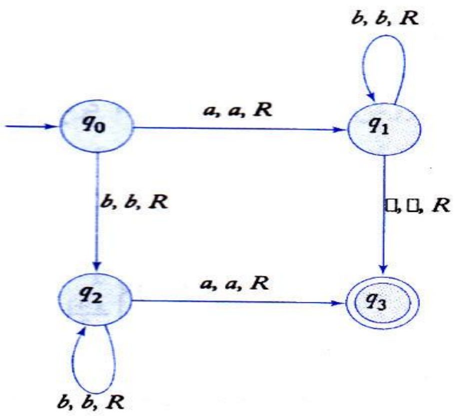
این ماشین زبان $\{a^n b^n : n \geq 1\}$ ، را می پذیرد. در این ماشین با خواندن یک a به جای آن x قرار داده می‌شود و a هـای بعدی را رد کرده تا به یک b برسد، سپس آن را با y جایگزین کرده و دوباره به سمت چپ‌ترین a رفته و آنرا با x جایگزین می‌کند و مجددا چپ‌ترین b را با y جایگزین می‌کند و این حرکت پاندولی را ادامه می دهد و تک‌تک a ها را بـا b نظیـر آن تطبیق می دهد. اگر a یا b ای باقی نمانده بود، آنگاه رشته حتما عضو زبان خواهد بود.

بعنوان مثال، ورودی aabb پیکربندی های متوالی زیر را ایجاد می کند:

$$q_0aabb \mapsto xq_1abb \mapsto xaq_1bb \mapsto xq_2ayb \mapsto q_2xayb \mapsto xq_0ayb \mapsto xxq_1yb$$
$$\mapsto xxyq_1b \mapsto xxq_2yy \mapsto xq_2xyy \mapsto xxq_0yy \mapsto xxyq_3y \mapsto xxyyq_3B \mapsto xxyyBq_fB.$$

مثال

یک ماشین تورینگ با گراف انتقال زیر، چه زبانی را می پذیرد؟ (فاصله خالی با نماد مربع نشان داده شده است.)



حل: ماشین داده شده زبان زیر را می پذیرد:

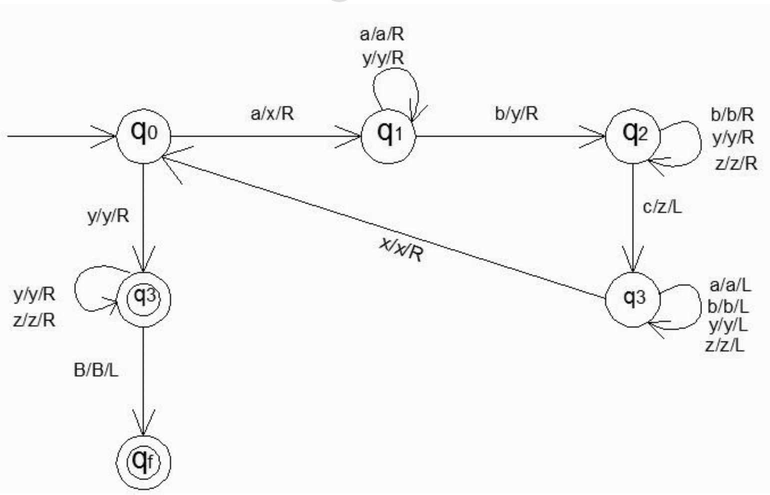
$$L = \{ab^n : n \geq 0\} \cup \{b^n a : n \geq 1\}$$

■

مثال

ماشین تورینگی برای پذیرش زبان $L = \{a^n b^n c^n : n \geq 1\}$، طراحی کنید.

حل: حرف a را با x ، b را با y و c را با z علامت‌گذاری می‌کنیم.



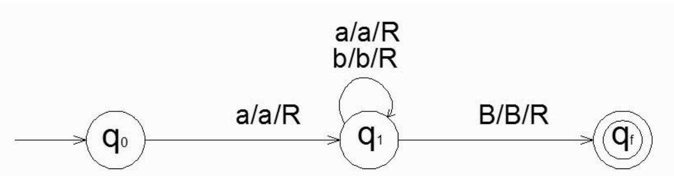
قبلا دیدیم که زبان $L = \{a^n b^n c^n : n \geq 1\}$ مستقل از متن نیست و توسط ماشین پشته ای پذیرش نمی‌شود. اما این زبان توسط تورینگ پذیرش شد. پس ماشین تورینگ از ماشین پشته ای قویتر می باشد.■

مثال

ماشین تورینگ زیر چه زبانی را می پذیرد؟

$$\delta(q_0,a)=(q_1,a,R)$$
$$\delta(q_1,a)=(q_1,a,R)$$
$$\delta(q_1,b)=(q_1,b,R)$$
$$\delta(q_1,B)=(q_f,B,R)$$

حل: شکل ماشین به صورت زیر است:



این ماشین زبان $L(a(a+b)^*)$ روی الفبای {a,b} را می پذیرد. پس ماشین تورینگ می‌تواند زبان مـنظم را بپذیرد.■

مثال

ماشین تورینگ زیر چه زبانی را می پذیرد؟

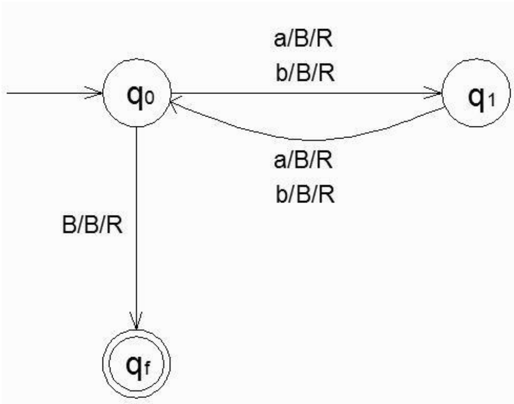
$$\delta(q_0,a)=(q_1,a,R)$$
$$\delta(q_1,b)=(q_2,b,R)$$
$$\delta(q_2,a)=(q_2,a,R)$$
$$\delta(q_2,b)=(q_f,b,R)$$

حل: این ماشین زبان $L=L(aba^*b)$ روی الفبای {a,b} را می پذیرد. ■

مثال

یک ماشین تورینگ طراحی کنید که زبان $L=\{w:$ زوج است $|w|\}$ را روی الفبای {a,b} پذیرش کند.

حل:



■

مثال

یک ماشین تورینگ طراحی کنید که زبان $L=\{a^ib^jc^k : k=i\times j\ ,\ i,j,k\geq 1\}$ را تصمیم گیری کند.

حل: کارهای انجام شده روی رشته ورودی توسط ماشین:

۱- ابتدا رشته ورودی را از چپ به راست پویش کرده تا مطمئن شود که رشته ورودی عضو $a^*b^*c^*$ است و اگر اینطور نبود به حالت عدم پذیرش می رود.

۲- هد به انتهای سمت چپ بر گردانده می شود.

۳- یکی از a ها را علامت زده و آنقدر به راست می رود تا به یک نماد b برسد. سپس b ها را با c ها تطبیق می‌دهد، یعنی با علامت زدن یک b، یک c را نیز علامت می زند. این کار را تا تمام شدن bها ادامه می‌دهد.

۴- تمام b های علامت گذاری شده دوباره به نماد b برگردانده می شوند. تا زمانی که a باقی مانده باشد، مرحله سوم تکرار می شود. اگر تمام a ها علامت گذاری شده باشند، بررسی می شود که تمام c ها نیز علامت گذاری شده باشند. در این صورت رشته پذیرفته شده و در غیر این صورت رشته رد می شود.

این زبان را نمی توان با ماشین پشته ای پذیرفت. ■

برای همه ماشین های تورینگ، یک ماشین دیگر با تنها یک حالت پایانی وجود دارد که همـان زبـان را می پذیرد. در واقع اگر چند حالت پایانی داشته باشیم، یک حالت پایانی جدید معرفی می کنـیم و بـرای همه $q \in F$ و $a \in \Gamma$ انتقالات $\delta(q,a) = (q_f, a, R)$ را انجام می دهیم.

ماشین تورینگ به عنوان مترجم

ماشین تورینگ علاوه بر دارا بودن خاصیت پذیرش زبان‌ها، یک مدل ساده انتزاعی برای کامپیوترهای رقمی می‌باشد. در واقع تمامی توابع ریاضی معمولی توسط ماشین تورینگ، محاسبه پذیر بوده و میزان پیچیدگی آنها، تاثیری بر این امر نخواهد داشت.

تابع f محاسبه پذیر توسط تورینگ گفته می شود، اگر ماشین تورینگ مفروض M وجود داشته باشد که برای همه w های موجود در دامنه تابع، داشته باشیم: $q_0 w \xrightarrow{*} q_f f(w)$

مثال

ماشین تورینگی برای محاسبه جمع دو عدد صحیح مثبت x و y طراحی کنید.

قرارداد ۱ : هر عدد صحیح مثبت x بوسیله $w(x) \in \{1\}^+$ نمایش داده می شود، بطوریکه $|w(x)| = x$. مثلا عدد 3 به صورت 111 و یا عدد 2 به صورت 11 نمایش داده می شود.

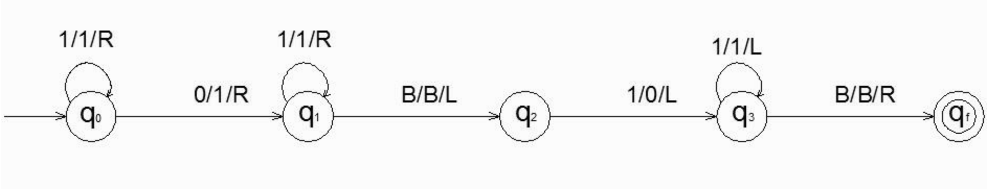
قرارداد ۲ : اعداد بوسیله یک 0 از هم جدا شده و هد روی چپ ترین علامت عدد x می باشد.

قرارداد ۳ : بعد از محاسبه، $w(x+y)$ روی نوار قرار گرفته و فقط یک 0 بعد از آن مشاهده می شود و هد در انتهای سمت چپ نتیجه قرار خواهد داشت.

بنابراین در واقع می‌خواهیم ماشین تورینگی را برای انجام محاسبه زیر طراحی کنیم:

$$q_0 w(x)0w(y) \overset{*}{\longmapsto} q_f w(x+y)0$$

حل : تنها کاری که باید انجام شود، انتقال صفر بین دو عدد، به انتهای سمت راست عدد y می‌باشد.



بررسی جمع 111 با 11 :

$$q_0111011 \mapsto 1q_011011 \mapsto 11q_01011 \mapsto 111q_0011 \mapsto 1111q_111 \mapsto 11111q_11$$
$$\mapsto 111111q_1B \mapsto 11111q_21 \mapsto 1111q_310 \mapsto \ldots \mapsto q_3B111110 \mapsto Bq_f111110.$$

■

تذکر:

کار با دنباله سمبل های یکتایی 1 ، هر چند در محاسبات مشکل به نظر می رسد، ولی در برنامه ریزی ماشین های تورینـگ بسیار سودمند است. برنامه های حاصل از این روش در مقایسه با روشهای دیگر کوتاهتر و ساده تر هستند.

مثال

ماشین تورینگ طراحی کنید که رشته هایی از 1 را کپی کند. مثلا با دادن 11 خروجی 1111 حاصل شود.

(یعنی برای هر $w \in \{1\}^{+}$ ، مقدار $q_0 w \overset{*}{\vdash} q_f ww$ را محاسبه کند.)

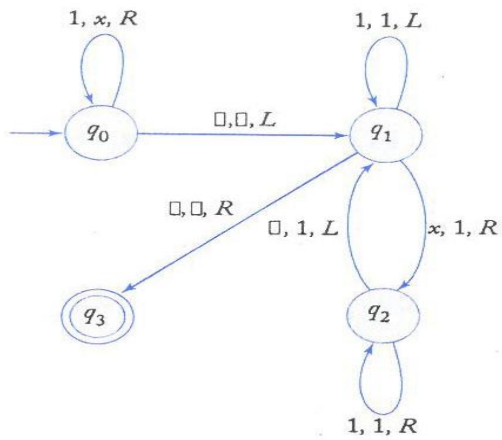
حل: مراحل کار:

۱- هر یک از 1 ها را با یک x جایگزین می کنیم.

۲- راست ترین x را پیدا کرده و آنرا با 1 جایگزین می کنیم.

۳- به اولین سمبل خالی سمت راست فعلی رفته و در آنجا یک 1 ایجاد می کنیم.

۴- مراحل ۲و۳ را آنقدر تکرار می کنیم تا هیچ x دیگری وجود نداشته باشد.

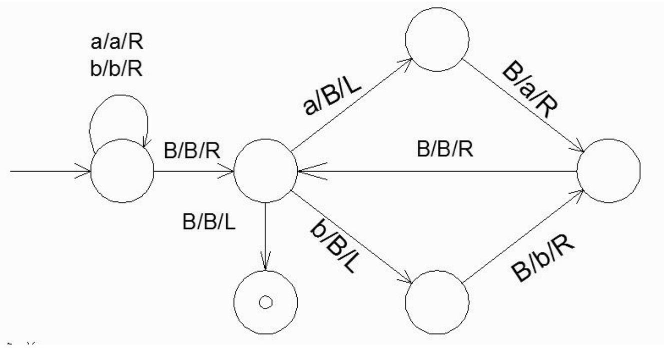


فاصله خالی با نماد مربع نشان داده شده است.

مثال

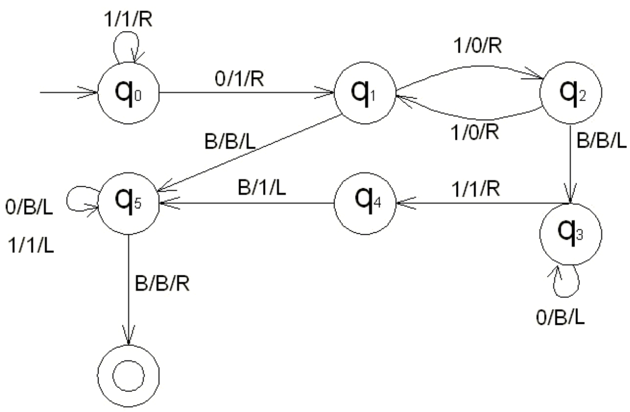
در الفبای {a,b}، ماشین تورینگی برای اتصال دو رشته به یکدیگر طراحی کنید. ($q_0 w_1 B w_2 \overset{*}{\vdash} q_f w_1 w_2$)

حل: بین دو رشته یک بلانک قرار دارد. ابتدا تمام حروف رشته اول را خوانده تا به B برسیم، سپس بعد از خواندن B ، اگر در ابتدای رشته دوم حرف a بود آن را به B تبدیل کرده و به سمت چپ برگشته و B را به a تغییر می دهیم و اگر b بود آن را B کرده و به سمت چپ برگشته و B را به b تغییر می‌دهیم. سپس به سمت راست حرکت کرده و این عملیات را تا رسیدن به B در q_2 ادامه می‌دهیم.



مثال

خروجی ماشین تورینگ زیر با ورودی $\{w = x0y : x = 1^*, y = 1^*\}$ با فرض $x = 11, y = 1111$ را مشخص کنید.



حل:

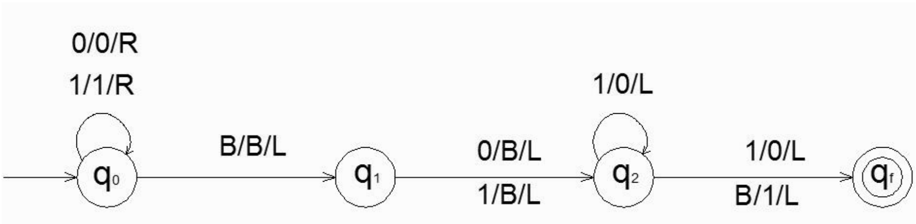
ورودی $w = 1101111$ ، یعنی x برابر 2 و y برابر 4 است که طبق تعریف با یک صفر از هم جدا شده اند. هد از روی همه یکها عبور کرده تا به صفر برسد(چرخه در q0). سپس ماشین به حالت q1 رفته و صفر بین x وy به یک تبدیل می شود. با حرکت بین دو حالت q1 وq2 ،یکهای y به صفر تبدیل شده و چون تعداد یکهای y زوج است، در پایان به q1 می رسد. هد به علامت فاصله خالی پایان رشته برسد و به چپ حرکت کرده و ماشین به حالت q5 مـی رود. $(w = 111000\underline{0})$ در ایـن حالت، تمام صفرها به سمت ابتدای رشته به فاصله خالی تبدیل شده تا به یک برسـد. $(w = 11\underline{1})$. از یکهـا نیـز بـه سـمت

ابتدای رشته عبور کرده تا به فاصله خالی اول رشته برسد. در این حالت ماشین به حالت پایانی می رود. رشته در این حالت برابر 111 یعنی 3 است. این ماشین تابع $f(x,y)=x+1+(y \mod 2)$ را محاسبه می‌کند:

$$f(2,4)=2+1+(4 \mod 2)=3+0=3$$

مثال

خروجی ماشین تورینگ زیر با ورودی 1101 و 111 را مشخص کنید.



اصلاح: یال از q_2 به q_f با برچسب $1/0/L$ باید $0/1/L$ شود.

حل:

الف- ورودی 1101 (عدد دهدهی ۱۳)

در حالت q_0 از تمام یکها و صفرهای اول رشته به سمت راست عبور می کنیم. با رسیدن به blank آخر رشته، یک واحد به عقب بر می گردیم. هم اکنون در q_1 هستیم و هد روی 1 آخر رشته قرار دارد. در این حالت به جای 1 انتهای رشته، blank قرار داده و به چپ می رویم. هم اکنون در q_2 هستیم و رشته روی نوار 110 است. در این حالت 0 آخر رشته به 1 تبدیل شده و به حالت پایانی می رسیم. پس خروجی 111 است.

ب- ورودی 111 (عدد دهدهی ۷)

در حالت q_0 از تمام یکها و صفرهای اول رشته به سمت راست عبور می کنیم. با رسیدن به blank آخر رشته، یک واحد به عقب بر می گردیم. هم اکنون در q_1 هستیم. در این حالت به جای 1 انتهای رشته، blank قرار داده و به چپ می رویم. هم اکنون در q_2 هستیم و رشته برابر 11 است. در این حالت تمامی 1 های رشته از آخر به اول به 0 تبدیل می شود. سپس با رسیدن به بلانک اول رشته، این بلانک به 1 تبدیل شده و به حالت پایانی می رویم. پس خروجی 100 است.

توضیح در رابطه با عملکرد ماشین:

این ماشین تابع $F(x) = x \; div \; 2 + 1$ را محاسبه می کند. x یک عدد در مبنای دو است. این تابع، ابتدا مقدار x را نصف کرده و سپس یک واحد به آن اضافه می کند. عمل نصف کردن با حذف بیت کم ارزش (شیفت به راست) انجام می شود.(انتقال از q_1 به q_2).

عمل یک واحد اضافه کردن به صورت زیر است:

الف: اگر رشته با صفر تمام شود، فقط آن صفر به یک تبدیل می شود.

ب: اگر رشته با یک تمام شود، از آخر به اول رشته حرکت کرده و تمام یکها را صفر کرده و اولین صفر را یک می‌کنیم.

در حالتی که بعد از حذف بیت کم ارزش، بقیه بیتها یک باشند، همه یکها به صفر تبدیل شده و به blank اول رشته می‌رسیم، آن blank به یک تبدیل می شود. ■

مدل های دیگر ماشین تورینگ

در این بخش نشان می‌دهیم که ماشین‌های تورینگ استاندارد، معادل بـا مـدل‌های پیچیـده‌تر هسـتند. طبـق تـز تورینـگ، پیچیده کردن ماشین‌های تورینگ استاندارد از طریق تجهیز آنها به ابزار ذخیره سازی پیچیده تر، تاثیری بـر قـدرت ماشـین ندارد. چون هر نوع محاسبه ای که با این ماشین های جدید قابل انجام باشد، مدلی از محاسبه مکانیکی محسوب شـده و بـه همین علت بوسیله یک مدل استاندارد هم قابل انجام است.

تز تورینگ: این فرضیه می‌گوید که هر نوع محاسبه‌ای که بطور مکانیکی قابل انجام باشـد، بـا ماشـین تورینگ هم قابل انجام است.

ویژگی های ماشین تورینگ استاندارد:

۱- نامحدود بودن نوار ماشین از دو طرف (ممکن بودن حرکت به راست یا چپ به هر تعداد)

۲- معین بودن (به ازای هر پیکربندی فقط یک حرکت تعریف می‌شود.)

۳- عدم وجود هیچ فایل ورودی خاصی و همچنین عدم وجود هیچ وسیله خروجی خاصی.

مدل‌های دیگر ماشین های تورینگ عبارتند از:

۱- اعمال تغییرات جزئی در تعریف ماشین تورینگ (سکون دار- با نوار نیمه متناهی – آف لاین)

۲- ماشین های تورینگ با حافظه پیچیده تر(چند نواره و چند بعدی)

۳- ماشین های تورینگ نامعین

۴- ماشین تورینگ عمومی

۵- اتوماتای کراندار خطی

ماشین های تورینگ سکون دار

هد در این نوع ماشین ها می تواند پس از بازنویسی محتوای سلول، در جای خود باقی بماند و به جلو یا عقب حرکت نکند. گنجاندن این انتخاب جدید برای حرکت هد، قدرت ماشین را افزایش نمی دهد. دسته ماشین های تورینگ سکون دار، هم ارز با دسته ماشین های تورینگ استاندارد می باشند.

ماشین های تورینگ با نوار نیمه نامتناهی

نوار در این ماشین فقط از یک طرف نامحدود است. وقتی هد در انتها قرار می گیرد، حرکت به چپ مجاز نیست. این محدودیت هیچ تاثیری بر قدرت ماشین نمی گذارد.

ماشین های تورینگ آف لاین

ماشین تورینگ offline ، علاوه بر نوار شامل یک فایل ورودی فقط خواندنی نیز می‌باشد. در این نوع ماشین‌ها، تمامی حرکت‌ها توسط موارد زیر تصمیم گیری می شود:

الف- حالت درونی

ب- سمبلی که در حال حاضر از فایل ورودی خوانده می شود.

ج- آنچه که بوسیله هد خواندن-نوشتن مشاهده می شود.

ماشین های تورینگ با حافظه پیچیده تر

می توان ابزار ذخیره سازی ماشین تورینگ استاندارد را پیچیده تر کرد، اما این عمل قدرت ماشین را افزایش نمی دهد. با ذکر دو مثال(چند نواره و چند بعدی)، این موضوع را نشان می دهیم.

ماشین های تورینگ چند نواره

ماشین تورینگی با چند نوار که هر نوار، دارای هد خواندن- نوشتن می‌باشد که به طور مستقل کنترل می شود.

در یک ماشین دو نواره، تابع انتقال $\delta(q_0, a, e) = (q_1, x, y, L, R)$، به این معنی است که ماشین در حالت q_0 بوده و اولین هد یک a و دومین هد یک e را می بیند. سپس a روی اولین نوار با x جایگزین شده و هد به سمت چپ حرکت

خواهد کرد. در عین حال، e روی نوار دوم به y تغییر یافته و هد به سمت راست حرکت مـی کنـد. پـس از ایـن کـار واحـد کنترل به q_1 تغییر حالت می دهد.

مثال

به کمک ماشین تورینگ دو نواره پذیرش زبان $\{a^nb^n\}$ بسیار ساده تر می شود. در ابتدا رشته a^nb^n روی نوار اول قرار دارد. سپس تمامی a ها را از نوار اول خوانده و به نوار دوم کپی می کنیم. با رسیدن به اولین b روی نوار اول، آنها را با a های نوار دوم تطبیق می دهیم و به سادگی تعیین می کنیم آیا تعداد a ها و b ها برابر هستند یا خیر. بنابراین بدون نیاز به جابجـایی متوالی هد به راست و چپ، این عمل انجام شد. ■

ماشین های تورینگ چند بعدی

در این ماشین‌ها، نوار به صورت نامتناهی در بیش از یک بعد گسترش یافته است. در ماشین تورینگ دو بعدی، هد عـلاوه بـر حرکت به چپ و راست، می تواند به بالا و پایین نیز حرکت کند.

ماشین های تورینگ نامعین

ماشین تورینگ نامعین مشابه تورینگ معین است با این تفاوت که دارای تغییر وضعیت‌های متفاوتی می‌باشـد، یعنـی در هـر لحظه از محاسبات، ماشین می تواند یکی از چندین انتخاب را دنبال کند. بـه عبـارتی بـرد تـابع انتقـال آن، مجموعـه تمـام انتقالات ممکنی است که بوسیله ماشین انتخاب می شوند.

تـابع δ بــه صــورت $Q\times\Gamma \rightarrow 2^{Q\times\Gamma\times\{L,R\}}$ تعریــف مــی شــود. تـابع δ در ماشــین تورینــگ معــین بــه صــورت $Q\times\Gamma \rightarrow Q\times\Gamma\times\{L,R\}$ می باشد.

مثال

ماشین تورینگ با انتقالاتی به فرم زیر نامعین است:

$$\delta(q_0,a)=\{(q_1,b,R),(q_2,c,L)\}$$

■

چند نکته:

۱- دسته ماشین های تورینگ معین و دسته ماشین های تورینگ نامعین هم ارز هستند.

۲- یک ماشین تورینگ نامعین به هیچ وجه قدرتمندتر از نوع معین خود نیست. در واقع نـامعین بـودن بـه قـدرت ماشـین تورینگ اضافه نمی کند.

۳- هر ماشین تورینگ نامعین را می توان بوسیله یک ماشین تورینگ معین شبیه سازی کرد.

۴- اگر M_N یک ماشین تورینگ نامعین باشد، آنگاه یک ماشین تورینـگ معـین ماننـد M_D وجـود دارد، بـه طوری که: $L(M_N) = L(M_D)$

چند نکته:

۱- وقتی که بیش از یک حرکت ممکن باشد، ماشین بـه تعـداد لازم کپـی از خـودش تهیـه می‌کنـد و بـه هرکدام کاری را واگذار می‌کند.

۲- یک آتاماتای پشته‌ای، مانند یک ماشین تورینگ نامعین اسـت کـه نـوار آن بـه صـورت پشـته اسـتفاده می‌شود.

۳- دسته اتوماتای دو پشته ای، هم ارز با دسته ماشین‌های تورینگ است.

ماشین تورینگ عمومی

ماشین تورینگ عمومی M_u، اتوماتی است که با در اختیار داشتن توصیف هر ماشـین تورینـگ M بعنـوان ورودی و رشته w، قادر به شبیه سازی محاسبه M روی w می باشد.

آتاماتای کراندار خطی (LBA)

یک اتومات کراندار خطی،یک ماشین تورینگ نامعین است، ولی با این محدودیت که مقدار نواری که می‌تواند استفاده کند تابعی از ورودی است. همچنین قسمت قابل استفاده نوار به سلول‌هایی که حاوی ورودی است محدود می‌باشند و برای حفاظت از این محدوده از دو علامت کروشه],[استفاده می‌شود.

(LBA : Linear Bounded Automata)

مثال

زبان $\{a^n b^n c^n : n \geq 1\}$ توسط یک LBA پذیرفته می‌شود. چون محاسباتی که برای پذیرش این زبان نیاز است، احتیاجی به فضائی خارج از ورودی اولیه ندارد. ■

چند نکته:

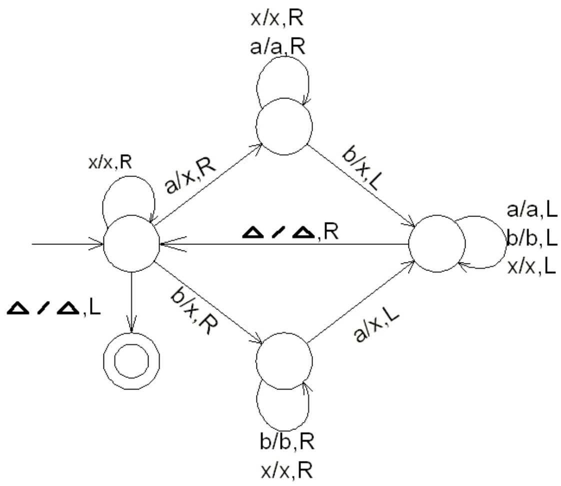
۱- هر زبان پذیرفته شده بوسیله یک اتومات پشته ای، بوسیله یک اتومات کراندار خطی هم پذیرفته می‌شود، اما زبان هایی وجود دارند که بوسیله اتوماتای کراندار خطی پذیرفته می شوند، اما هیچ اتوماتای پشته ای به ازای آن وجود ندارد. پس ماشین کراندار خطی، قویتر از ماشین پشته ای است.

۲- هر زبان مستقل از متن فاقد λ را می توان بوسیله یک LBA پذیرفت.

۳- هیچ روشی برای اثبات تناظر LBA های معین با نسخه نامعین خود وجود ندارد.

تمرین ۷

۱- ماشین تورینگ زیر چه زبانی را می پذیرد؟ (Δ نماد blank ماشین تورینگ است)



۲- در ماشین تورینگ M با دستورات زیر اگر محتوی نوار برابر رشته aaabbb باشد، پس از دقیقا 6 حرکت، محتوای چه خواهد بود؟

$$\delta(q_0,a)=(q_1,X,R) \qquad \delta(q_0,B)=(q_f,B,R)$$
$$\delta(q_1,a)=(q_1,a,R) \qquad \delta(q_1,b)=(q_2,Y,L)$$
$$\delta(q_1,Y)=(q_1,Y,R) \qquad \delta(q_1,B)=(q_f,B,R)$$
$$\delta(q_2,a)=(q_2,a,L) \qquad \delta(q_2,X)=(q_1,X,R) \qquad \delta(q_2,Y)=(q_2,Y,L)$$

۳- زبان ماشین تورینگ با قواعد زیر کدام است؟

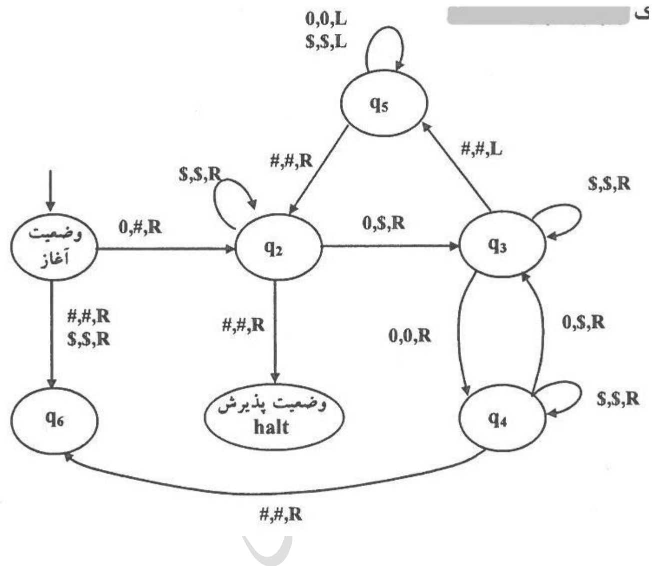
$$\delta(q_0,a)=(q_1,x,R) \qquad \delta(q_0,y)=(q_3,y,R)$$
$$\delta(q_1,a)=(q_1,a,R) \qquad \delta(q_1,y)=(q_1,y,R) \qquad \delta(q_1,B)=(q_2,y,L)$$

$\delta(q_2,a)=(q_2,a,L)$ $\delta(q_2,y)=(q_2,y,L)$

$\delta(q_3,x)=(q_0,x,R)$ $\delta(q_3,y)=(q_3,y,R)$ $\delta(q_3,B)=(q_f,B,R)$

۴- آیا ماشین تورینگ زیر، 000000 را می پذیرد؟ (نماد \$ یکی از حروف نوار ماشین است.)

(نماد # نشان دهنده خانه خالی)



۵- کدام یک از زبان های زیر توسط ماشین تورینگ فقط خواندنی پذیرفته نمی شوند؟

الف- $\{x \in \{0,1\}^*$: حاصلضرب تعداد صفرهای x در تعداد یک های آن بر 7 بخش پذیر باشد$\}$

ب- $\{x \in \{0,1\}^*$: تعداد صفرهای x منهای تعداد یک های x برابر 2 باشد$\}$

۶- یک ماشین تورینگ در حداکثر ۶۴ مرحله به پیکربندی نهایی می رسد. آیا زبان این ماشین بـا یـک عبـارت مـنظم قابـل توصیف است؟

۷- آیا تعریف الگوریتم توسط چرچ با تعریف الگوریتم توسط تورینگ معادل است؟

۸- آیا می توان ماشین تورینگی ساخت که خروجی نداشته باشد و به ازاء هیچ ورودی نیز در حلقه نیافتد، یعنی بـه ازای هـر ورودی متوقف شود؟

۹- آیا با استفاده از نمادها و حالات اضافی می توان برای هر ماشین تورینگ قطعی، یک ماشین تورینگ قطعی معادل ساخت که سرعت محاسبه آن برای هر ورودی حداقل دو برابر ماشین اولیه باشد؟

۱۰- یک ماشین تورینگ آف لاین با چه ویژگی‌هایی، متناظر با یک ماشین متناهی است؟

۱۱- یک ماشین تورینگ تک نواره که قادر به نوشتن در محلی از نوار که شامل رشته ورودی است نباشد، چه نـوع زبـانی را می پذیرد؟

۱۲- یک ماشین تورینگ با امکان باقی ماندن در محل قبلی به جای حرکت به سمت چپ، چگونه ماشین است؟

۱۳- ماشین تورینگ چند شیاره را تعریف کنید.

۱۴- یک ماشین تورینگ یکبار مصرف (write-once TM) ، چه نوع ماشینی است؟

۱۵- یک ماشین تورینگ با امکان بازنشاندن به سمت چپ (TM with left reset) ، چه نوع ماشینی است؟

۱۶- آیا زبان $L=\{x^i\ y^j\ z^{j+2}\ w^k\ v^{i+k} : i,j,k \ge 0\}$ توسط LBA پذیرفته می شود؟

۱۷- ماشین تورینگ پذیرنده زبان $L = \{0^n1^n : n \geq 0, n \notin \{2,5\}\}$ را طراحی کنید.

۱۸- یک ماشین تورینگ طراحی کنید که زبان $L = \{0^{2^n} : n \geq 0\}$ را تصمیم گیری کند.

۱۹- ایده کار در ماشین تورینگی که وسط رشته ای با طول زوج را پیدا می‌کند، چگونه است؟

پاسخ تمرین ۷

۱- ماشین رشته هایی را در الفبای $\Sigma=\{a,b,x\}$ می‌پذیرد که تعداد a ها با b ها برابر باشند. مانند رشته axb یا bxaab.

$$L=\{w\in(a+b+x)^*:n_a(w)=n_b(w)\}$$

۲- در زیر 6 حرکت آورده شده و در انتها $XaaYbb$ روی نوار خواهد بود:

$$q_0aaabbb \mapsto Xq_1aabbb \mapsto Xaq_1abbb \mapsto Xaaq_1bbb \mapsto Xaq_2aYbb$$
$$\mapsto Xq_2aaYbb \mapsto q_2XaaYbb$$

۳- ماشین داده شده رشته هایی را می پذیرد که تعداد a ها و تعداد b ها در آنها برابر است و با حرف a نیز شروع می شود. به عبارتی زبان این ماشین $\{a^nb^n:n\geq 1\}$ می باشد.

۴- خیر- عملکرد ماشین:

$$q_0000000\mapsto\#q_200000\mapsto\#\$q_30000\mapsto\#\$0q_4000\mapsto\#\$0\$q_300$$
$$\mapsto\#\$0\$0q_40\mapsto\#\$0\$0\$q_3\#\mapsto\#\$0\$0q_5\$\#\mapsto\#\$0\$q_50\$\#\mapsto\#\$0q_5\$0\$\#$$
$$\mapsto\#\$q_50\$0\$\#\mapsto\#q_5\$0\$0\$\#\mapsto q_5\#\$0\$0\$\#\mapsto\#q_2\$0\$0\$\#$$
$$\mapsto\#\$q_20\$0\$\#\mapsto\#\$\$q_3\$0\$\#\mapsto\#\$\$\$q_30\$\#\mapsto\#\$\$\$0q_4\$\#$$
$$\mapsto\#\$\$\$0\$q_4\#\mapsto\#\$\$\$0\$\#q_6$$

ماشین در حالت q6 متوقف شد و رشته ورودی پذیرفته نشد.

تذکر: ماشین رشته هایی را می پذیرد که تعداد صفرهای آن، توانی از ۲ باشد.

۵- در یک ماشین تورینگ فقط خواندنی، هد فقط می تواند بخواند و توانایی نوشتن یـا پـاک کـردن خانـه های نوار حافظه را ندارد. این ماشین مانند یک ماشـین متنـاهی اسـت و بنـابراین فقـط زبانهـای مـنظم را پذیرش می‌کند. پس زبان گزاره ب که مستقل از متن است را نمی پذیرد.

۶- بله- زبان ماشین تورینگی که در تعداد مراحل متناهی به پیکربندی نهایی می رسد، با یک عبارت منظم قابل توصیف است.

۷- بله- چرچ برای تعریف الگوریتم، از ریاضیات لاندا و تورینگ از ماشین استفاده کرد. طبـق نظریـه چـرچ-تورینـگ ایـن دو تعریف معادل هستند.

۸- بله

۹- بله- به طور نمونه هر بار با خواندن دو نماد ورودی، تغییر حالت دهد.

۱۰- ویژگی هایی که یک ماشین تورینگ آف لاین را با یک ماشین متناهی متناظر می کند:

الف- ورودی فقط یک مرتبه قابل خواندن باشد.

ب- ورودی از چپ به راست حرکت کند.

پ- ورودی قابل بازنویسی نباشد.

ت- حداکثر فقط از n سلول اضافی نوار، بعنوان فضای کاری استفاده کند.(n برای تمام ورودی ها ثابت است.)

۱۱- این ماشین قادر به تشخیص زبانهای مستقل از متن می باشد.

۱۲- مشابه تورینگ معمولی است ولی تابع انتقال آن به صورت $\delta: Q\times\Gamma \rightarrow Q\times\Gamma\times\{R,S\}$ می باشد. در هر لحظه ماشین می‌تواند هد را به سمت راست حرکت داده یا در محل قبلی باقی بماند. این نوع ماشین معادل ماشین تورینگ معمولی نیست.(منظور از S ، باقی ماندن در محل قبلی است.)

۱۳- اگر در ماشین تورینگ، هر یک از سلول های نوار به چند بخش به نام شیار تقسیم شود به ماشین حاصل، ماشین تورینگ چند شیاری می گویند. علائم نوار این ماشین، چندتایی‌های الفبای ساده تر دیگری می باشند. این تغییر در تعریف ماشین تورینگ، آن را گسترش نمی دهد، چون فقط لازم است، Γ را الفبایی فرض کنیم که هر یک از سمبل های آن از چندین بخش تشکیل شده است.

۱۴- یک ماشین تورینگ تک نواره است که هر خانه از نوار را فقط یکبار می تواند تغییر دهد. این نوع ماشین، معادل یک ماشین تورینگ استاندارد می‌باشد.

۱۵- مشابه یک ماشین تورینگ معمولی است با این تفاوت که تابع انتقال به صورت زیر است:

$\delta: Q\times\Gamma \rightarrow Q\times\Gamma\times\{R, reset\}$

تابع انتقال $\delta(q_i,a)=(q_j,b,reset)$ یعنی هر گاه ماشین در حالت q_i بوده و نماد a را بخواند، پس از نوشتن نماد b به جای a ، به ابتدای سمت چپ نوار بازگشته و به حالت q_j می رود. این ماشین نمی تواند یک نماد به چپ برود. این ماشین کلاس زبان های تشخیص پذیر تورینگ را تشخیص می‌دهد.

۱۶- زبان L را می توان به صورت $x^i y^j z^j z^2 w^k v^k v^i$ نوشت که زبانی مستقل از متن معین است. بنابراین توسط DPDA قابل پذیرش است. در نتیجه توسط LBA که از PDA قویتر است، نیز قابل پذیرش است.

۱۷- برای طراحی ماشین تورینگ پذیرنده این زبان، از ۴ ماشین تورینگ استفاده می کنیم:

ماشین اول: پذیرنده زبان $\{0^n1^n : n \geq 0\}$

ماشین دوم: پذیرنده 0^21^2

ماشین سوم: پذیرنده 0^51^5

ماشین چهارم: این ماشین رشته ورودی را به ۳ ماشین بـالا داده و اگـر ماشـین اول آن را پذیرفتـه و دو ماشـین دیگـر آنـرا نپذیرند، آنگاه این ماشین رشته را می پذیرد. اگر یکی از ماشین‌های دوم یا سوم رشته را بپذیرنـد، آنگـاه رشـته ورودی را رد (reject) می کند.

۱۸- رشته های زبان داده شده، رشته هایی از صفر هستند که طول آنها توان ۲ باشد. به طور نمونـه رشـته های:

0,00,0000,00000000,...

کارهای انجام شده روی رشته ورودی توسط ماشین:

۱- از سمت چپ به راست روی نوار حرکت می کند تا از تمام صفرها عبور کند و صفرها را به صـورت یـک در میان علامت می گذارد.

۲- اگر در مرحله ۱ فقط بیش از یک عدد صفر وجود دارد، به مرحله پذیرش می رود.

۳- اگر در مرحله ۱ بیش از یک عدد صفر وجود دارد و تعداد صفرها فرد باشد، به مرحله عدم پـذیرش مـی رود.

۴- هد به انتهای سمت چپ نوار بازگردانده می شود.

۵- به گام ۱ باز می گردد.

هر بار که گام اول اجرا می شود،تعداد صفرها نصف می شود. وقتی که در گام اول از روی تمام صفرها عبـور می کند، مشخص می شود که تعداد صفرها فرد یا زوج بوده است. اگر تعداد صـفرها فـرد و بزرگتـر از یـک باشد، تعداد صفرها در رشته اولیه نمی تواند توان ۲ باشد. بنابراین ماشین رشته ورودی را نمی پـذیرد. ولـی اگر فقط یک عدد صفر باقی مانده باشد،نشان دهنده این است که تعـداد صـفرها در رشـته ورودی تـوان ۲ است و رشته پذیرفته می شود.

۱۹- ایده کار به این صورت است که یک سمبل c در هر یک از دو سر رشته اضافه می کنیم. مثلا اگر رشته ورودی aaaa باشد، خواهیم داشت: caaaac. حال در سمت چپ به جای ca از ac و در سمت راست به جای ac از ca استفاده می کنیم. این کار را ادامه داده تا c ها به هم برسند. در نهایت یک c را حذف می کنیم. سمبل c باقی مانده، وسط رشته را نشان می دهد.

فصل ۸

زبان‌های بازگشتی-گرامر بدون محدودیت وحساس به متن

خانواده زبان‌های مرتبط با ماشین‌های تورینگ بسیار گسترده اند، چون ماشین های تورینگ قادر به انجام انواع محاسبات الگوریتمی هستند. البته زبانی وجود دارد که توسط هیچ ماشین تورینگی پذیرفته نشود چون تعداد زبان ها، بیشتر از ماشین‌های تورینگ است.

زبان های بازگشتی (RE) و بازگشتی شمارش پذیر(REC)

زبان بازگشتی شمارش پذیر(برشمردنی): زبانی که ماشین تورینگی وجود داشته باشد که آنرا پذیرش کند.

زبان بازگشتی : زبانی که ماشین تورینگی باشد که آن را پذیرفته و روی هر w موجود در Σ^+ ، در یک حالت پایانی یا غیر پایانی، متوقف شود.

بین زبانی که یک ماشین تورینگ پذیرنده به ازای آن وجود دارد و زبانی که یک الگوریتم عضویت به ازای آن وجود دارد، تفاوت هست. وجود یک ماشین تورینگ پذیرنده، به معنای وجود الگوریتم عضویت مربوطه نیست، چون ماشین‌های تورینگ لزوما برای ورودی که آنرا نمی پذیرند، توقف نمی کنند و ممکن است در حلقه بی افتند.

هر زبانی که بوسیله یک روش الگوریتمی مستقیم قابل توصیف باشد، بوسیله یک ماشین تورینگ هم قابل پذیرش بوده و بنابراین بازگشتی شمارش پذیر است.

زبان‌های بازگشتی زیر مجموعه زبان‌های بازگشتی شمارش پـذیر مـی باشـند. بـه عبـارتی هـر زبـان بازگشتی، بازگشتی شمارش پذیر نیز می‌باشد. یک زبان بازگشتی شمارش پـذیر وجـود دارد کـه بازگشـتی نیست.

هر زبان بازگشتی می تواند زبان یک ماشین تورینگ غیر قطعی باشد.

مثال

زبان‌های زیر، هم بازگشتی هستند و هم بازگشتی شمارش پذیر :

$$L=\{a^n b^n c^{2n} : n \ge 0\}$$

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

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

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

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

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

■

مثال

چرا زبان بازگشتی شمارش پذیر، را زبان تشخیص پذیر نیز می نامند؟

حل: یک تشخیص‌دهنده، الگوریتمی است که در صورت تعلق یک رشته به یک زبان می تواند تعلق آن را مشخص کند. اگر رشته متعلق به زبان نباشد، ماشین یا در حالت غیر پایانی متوقف شده و یا در حلقه بی‌نهایت(Loop) می افتد. بنابراین زبان بازگشتی شمارش پذیر، را زبان تشخیص پذیر نیز می نامند.■

مثال

چرا زبان بازگشتی را زبان تصمیم پذیر نیز می نامند؟

حل: یک تصمیم گیرنده برای یک زبان، الگوریتمی است که مشخص می کند آیا رشته w متعلق به زبان هست یا خیر و در هر صورت ماشین نهایتا متوقف می شود. بنابراین زبان بازگشتی را زبان تصمیم پذیر(decidable) نیز می نامند.■

ویژگی‌های زبان‌های بازگشتی و بازگشتی شمارش پذیر

۱- یک زبان بازگشتی خواهد بود اگر و تنها اگر یک الگوریتم عضویت به ازای آن وجود داشته باشد.

۲- اگر L بازگشتی باشد، لزوما L^+ هم بازگشتی است.

۳- مکمل هر زبان بازگشتی، بازگشتی است.

۴- مکمل یک زبان مستقل از متن، حتما بازگشتی است.

۵- اشتراک یک زبان بازگشتی شمارش پذیر با زبان بازگشتی ، لزوما غیر بازگشتی نیست.

۶- اشتراک یک زبان بازگشتی شمارش ناپذیر با زبان بازگشتی، لزوما بازگشتی شمارش پذیر نیست.

۷- اشتراک زبان بازگشتی شمارش ناپذیر با بازگشتی شمارش پذیر، لزوما بازگشتی شمارش پذیر نیست.

۸- اگر L یک زبان متناهی باشد، آنگاه L^+ بازگشتی شمارش پذیر است.

۹- اگر L یک زبان مستقل از متن باشد، آنگاه L^+ بازگشتی شمارش پذیر است.

۱۰- اگر L_1 بازگشتی و L_2 بازگشتی شمارش پذیر باشد، آنگاه $L_2 - L_1$ لزوما بازگشتی شمارش پذیر است.

۱۱- اگر L و $\overline{L}$ هر دو بازگشتی شمارش پذیر باشند، آنگاه هر دو بازگشتی هستند.

۱۲- اگر L بازگشتی باشد، آنگاه $\overline{L}$ هم بازگشتی است و در نتیجه هر دو بازگشتی شمارش پذیر هستند.

۱۳- اگر یک زبان بازگشتی شمارش پذیر نباشد، مکمل آن بازگشتی نیست.

۱۴- مکمل زبان بازگشتی شمارش پذیر، لزوما بازگشتی شمارش پذیر نیست.

۱۵- یک زبان بازگشتی شمارش پذیر وجود دارد که مکمل آن بازگشتی شمارش پذیر نیست.

۱۶- یک زبان تصمیم پذیر است اگر و تنها اگر خود زبان و مکمل آن هر دو تشخیص پذیر تورینگ باشند.

۱۷- فرض کنید L چنان باشد که یک ماشین تورینگ وجود دارد که اعضای L را در ترتیب مناسب شمارش می‌کند. در اینصورت، L بازگشتی است.

۱۸- مجموعه تمام زبان هایی که بازگشتی شمارش پذیر نیستند، قابل شمارش نمی باشند.

نکاتی در رابطه با شمارش پذیر بودن

۱- اگر S یک مجموعه شمارش پذیر نامتناهی باشد، آنگاه مجموعه توانی آن، (2^S) شمارش پذیر نیست.

۲- مجموعه اعداد حقیقی R ، ناشمارا می باشد، چون تناظری بین R و N موجود نیست.

۳- مجموعه تمامی رشته ها روی الفبا Σ و Σ^* ، شمارش پذیر هستند.

۴- مجموعه تمامی دنباله های نامحدود، شمارش پذیر نیستند.

۵- تمامی زبان های سلسله مراتب چامسکی، شمارش پذیر هستند.

۶- مجموعه تمام ماشین های تورینگ، شمارش پذیر هستند.

۷- تمامی ماشین های منظم،پشته ای، LBA و تورینگ، نامتناهی ولی شمارش پذیر هستند.

۸- ضرب دکارتی تعدادی مجموعه شمارش پذیر، باز هم شمارش پذیر است.

گرامر بدون محدودیت

گرامر مفروض $G=(V,T,S,P)$ بدون محدودیت خوانده می شود، اگر تمامی قوانین آن به فرم $u \to v$ باشند که در آن u، عضو $(V \cup T)^+$ و v عضو $(V \cup T)^*$ می باشد. در گرامرهای بدون محدودیت، اساسا هیچ شرط و محدودیتی برای قواعد تولید قائل نمی شویم. بعلاوه هر تعداد غیر پایانی و پایانی را می‌توان با هر ترتیبی در طرفین راست و چپ قرار داد. فقط λ نمی تواند در سمت چپ قواعد تولید رخ دهد. این گرامر ها بسیار قدرتمندتر از گرامرهای منظم و مستقل از متن هستند.

گرامرهای بدون محدودیت، متناظر با بزرگترین خانواده زبان‌ها بوده و بوسیله ابزار مکانیکی قابل تشخیص می باشند.

مثال

زبان تولید شده توسط گرامر بدون محدودیت زیر را تعیین کنید.

$S \to aSBC \mid \lambda$

$aB \to ab$

$bC \to bc$

$bB \to bb$

$cC \rightarrow cc$

$cB \rightarrow Bc$

حل: گرامر رشته هایی را تولید می کند که تعداد a ها و b ها و c ها با یکدیگر برابر باشند و ابتدا a هـا، سـپس b هـا و در نهایت c ها ظاهر شوند. بنابراین داریم: $L = \{a^n b^n c^n : n \geq 0\}$ ■

مثال

زبان گرامر بدون محدودیت زیر را مشخص کنید.

$$S \rightarrow AB$$
$$A \rightarrow aAb$$
$$bB \rightarrow bbbB$$
$$aAb \rightarrow aa$$
$$B \rightarrow \lambda$$

حل: یک مسیر اشتقاق را طی می کنیم:

$$S \Rightarrow AB \Rightarrow aAbB \Rightarrow aaAbbB \overset{*}{\Rightarrow} a^n Ab^n B \Rightarrow a^{n+1} b^{n-1} B \Rightarrow a^{n+1} b^{n+1} B \Rightarrow \ldots$$

بر این اساس می توان ادعا کرد که گرامر فوق قادر به مشتق سازی زبان زیر است:

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

■

- گرامرهای بدون محدودیت، صرفا خانواده زبان های بازگشتی شمارش پذیر را ایجاد می کنند.
- هر زبانی که بوسیله یک گرامر بدون محدودیت ایجاد شود، بازگشتی شمارش پذیر است.
- برای هر زبان بازگشتی شمارش پذیر L، یک گرامر بدون محدودیت G وجود دارد، بطوریکه

L=L(G) .

گرامر حساس به متن

گرامر مفروض $G=(V,T,S,P)$ حساس به متن(وابسته به متن) خوانده می شود، اگر تمامی قوانین آن به فرم $x \to y$ باشند که در آن x و y عضو $(V \cup T)^+$ باشند و $|x| \le |y|$.

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

تعریف: زبان مفروض L حساس به متن خوانده می شود، اگر گرامر حساس به متن G وجود داشته باشد، بطوریکه $L=L(G)$ یا $L=L(G)\cup\{\lambda\}$.

مثال

یک گرامر حساس به متن برای زبان $L=\{a^n b^n c^n : n \ge 1\}$ پیدا کنید.

$S \to abc \mid aAbc$
$Ab \to bA$
$Ac \to Bbcc$
$bB \to Bb$
$aB \to aa \mid aaA$

تذکر: چون زبان حساس به متن $\{a^n b^n c^n\}$، مستقل از متن نیست، می توان نتیجه گرفت که خانواده زبانهای مستقل از متن، یکی از زیر مجموعه های خانواده زبان های حساس به متن است.

- تمامی زبان‌های حساس به متن، بازگشتی هستند.
- یک زبان بازگشتی وجود دارد که حساس به متن نمی‌باشد.
- هر زبان حساس به متنی بوسیله یک ماشین تورینگ پذیرفته شده و بنابراین، بازگشتی شمارش پذیر محسوب می شود.

مثال

یک گرامر حساس به متن برای زبان $L=\{a^n b^n c^n d^n : n>0\}$ پیدا کنید.

حل:

$S \to aSBCD \mid abcd$

$cB \to Bc$

$dB \to Bd$

$dC \to Cd$

$bB \to bb$

$cC \to cc$

$dD \to dd$

■

مثال

یک گرامر حساس به متن برای زبان $L = \{a^n b^m c^n d^m : n \geq 1, m \geq 1\}$ پیدا کنید.

حل:

$S \to aAcD \mid aBcD$
$A \to aAc \mid aBc$
$Bc \to cB$
$Bb \to bB$
$BD \to Ed$
$cE \to Ec$
$bE \to Eb$
$aE \to ab$

■

به ازای هر زبان L حساس به متن دارای λ، یک اتومات کراندار خطی M وجود دارد بطوریکه L=L(M).

اگر زبان L بوسیله یک اتومات کراندار خطی مفروض به نام M پذیرفته شود، آنگاه یک گرامر حساس به متن وجود دارد که L را تولید می کند.

اتوماتای کراندار خطی عملا ضعیف‌تر از ماشین‌های تورینگ بوده و فقـط قـادر بـه پـذیرش یکـی از زیـر مجموعـه‌های مناسب زبان های بازگشتی می باشند.

ارتباط بین زبان ها، گرامر ها و ماشین ها

جدول زیر ارتباط بین زبان ها، گرامرها و ماشین ها را نشان می دهد:

گرامر	زبان	ماشین
منظم	منظم	متناهی (FA)
خطی	خطی	
	مستقل از متن معین	پشته ای معین (DPDA)
مستقل از متن	مستقل از متن	پشته ای (PDA)
حساس به متن	حساس به متن	کراندار خطی (LBA)
	بازگشتی	تورینگ تصمیم گیرنده
بدون محدودیت	بازگشتی شمارش پذیر	تورینگ تشخیص دهنده

ماشین های تورینگ تشخیص دهنده، قدرت بیشتری از ماشین های تورینگ تصمیم گیرنده دارند.

سلسله مراتب چامسکی

نوام چامسکی زبان ها را در چهار گروه، از نوع صفر تا نوع سه، دسته بندی کرد.

زبان های نوع صفر : شامل زبان‌های بازگشتی شمارش پذیر، می‌باشند.

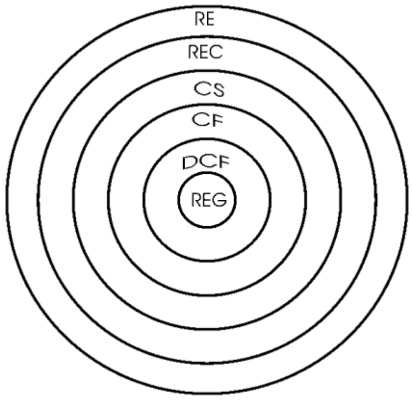
زبان های نوع یک : شامل زبان‌های حساس به متن می‌باشند.

زبان های نوع دو : شامل زبان‌های مستقل از متن می‌باشند.

زبان های نوع سه : شامل زبان‌های منظم می‌باشند.

هر یک از خانواده‌های زبان‌های نوع i ، یکی از زیر مجموعه های مناسب خانواده نوع $i-1$ محسوب می شوند.

نمودار زیر این رابطه را مشخص می کند:



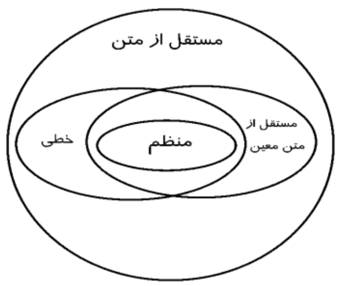
ارتباط بین زبان‌ها را می‌توان به کمک رابطه زیر نمایش داد:

$$REG \subseteq DCF \subseteq CF \subseteq CS \subseteq REC \subseteq RE$$

بازگشتی شمارش پذیر ⊇ بازگشتی ⊇ حساس به متن ⊇ مستقل از متن ⊇ مستقل از متن معین ⊇ منظم

هر زبان منظمی، زبانی مستقل از متن است. (چون زبان‌های منظم حالت خاصی از زبانهای مستقل از متن می‌باشند.)

در شکل زیر جایگاه زبان‌های خطی و مستقل از متن معین(قطعی) نشان داده شده است.



با توجه به شکل می توان گفت که:

الف: تمامی زبان های خطی، مستقل از متن نیز هستند.

ب: زبان خطی وجود دارد که مستقل از متن معین نیست. مانند $L = \{a^n b^n\} \cup \{a^n b^{2n}\}$

پ: زبان مستقل از متن معینی وجود دارد که خطی نیست. مانند: $L=\{w:n_a(w)=n_b(w)\}$

بررسی بسته بودن زبان ها تحت عملگرها

جدول زیر خواص بسته بودن شش نوع زبان را تحت عملگرهای مختلف نشان می دهد:

	منظم	مستقل از متن معین	مستقل از متن	حساس به متن	بازگشتی	بازگشتی شمارش پذیر
الحاق	✓	-	✓	✓	✓	✓
اجتماع	✓	-	✓	✓	✓	✓
اشتراک	✓	-	-	✓	✓	✓
مکمل	✓	✓	-	✓	✓	-
معکوس	✓	-	✓	✓	✓	✓
بستار ستاره	✓	-	✓	✓	✓	✓
همریختی	✓	-	✓	-	-	✓

همه انواع زبان های جدول بالا، تحت اشتراک منظم و تحت همریختی معکوس بسته هستند.

تمرین‌های فصل ۸

۱- چه زبانی توسط گرامر حساس به متن زیر تولید می‌شود؟

$$S \to aSBD \mid abD$$
$$bB \to bb$$
$$bD \to bc$$
$$DB \to BD$$
$$cD \to cc$$

۲- چه زبانی توسط گرامر بدون محدودیت زیر تولید می‌شود؟

$$S \to AB$$
$$A \to aAb$$
$$bB \to bbbB$$
$$aAb \to aa$$
$$B \to \lambda$$

۳- چه زبانی توسط گرامر بدون محدودیت زیر تولید می‌شود؟

$$S \to aSC \mid aAC$$
$$A \to bAD \mid bBD$$
$$BC \to c$$
$$DC \to CD$$
$$cC \to cc$$
$$D \to d$$

۴- چه زبانی توسط گرامر بدون محدودیت زیر تولید می‌شود؟

$$S \to ABaC$$
$$Ba \to aaB$$
$$BC \to DC \mid E$$
$$aD \to Da$$
$$AD \to AB$$
$$aE \to Ea$$
$$AE \to \lambda$$

۵- آیا زبان $L = \{a^n b^m c^n d^m : n, m \geq 0\}$ بازگشتی است؟

۶- با فرض اینکه L_1 و L_2 زبان‌های بازگشتی شمارش پذیر باشند، آیا $L_1.(L_2 \bigcup L_1^*)$ بازگشتی شمارش پذیر است؟

۷- اگر <M> یک کودینگ از ماشین تورینگ باشد، آیا زبان L ، تصمیم پذیر است؟

$L = \{< M, w >:$ ماشین M پس از 8 گام w را می پذیرد $\}$

۸- در ماشین تورینگ، اگر نوار از یک سمت محدود شود، آیا کلاس زبانهای مشخص شده با کلاس زبانهای بازگشتی شمارش پذیر، تفاوت خواهد کرد؟

۹- اگر تعداد نمادهای ماشین تورینگ محدود شود، آیا کلاس زبانهای مشخص شده با کلاس زبانهای بازگشتی شمارش پذیر تفاوت خواهد کرد؟

۱۰- اگر کلاس زبان‌های شمارش پذیر بازگشتی، کلاس زبان های پذیرفته شده توسط ماشین های تورینگ قطعی یک نـواره باشد، آیا محدود کردن حرکت هد ماشین فقط به سمت راست، منجر به تغییر این کلاس می شود؟

۱۱- کدام گزاره درست است؟

الف- برای هر زبان بازگشتی شمارش پذیر، ماشین تورینگ قطعی وجود دارد که برای هر ورودی داخل زبان در زمان متناهی متوقف می شود.

ب- برای زبان بازگشتی شمارش پذیر، ممکن است ماشین با یک رشته خارج از زبان متوقف نشود.

۱۲- اگر A و B دو مجموعه تصمیم پذیر(decidable) باشند، آیا $A \bigcap B^c$ لزوما تصمیم پذیر است؟

حل تمرین فصل ۸

۱- چند اشتقاق نمونه به صورت زیر است:

$$S \Rightarrow abD \Rightarrow abc$$

$$S \Rightarrow aSBD \Rightarrow aabDBD \Rightarrow aabBDD \Rightarrow aabbDD \Rightarrow aabbcD \Rightarrow aabbcc$$

بنابراین زبان گرامر $\{a^n b^n c^n : n \geq 1\}$ است.

۲- زبان تولید شده برابر است با:

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

۳- یک اشتقاق نمونه به صورت زیر است:

$$S \Rightarrow aSC \Rightarrow aaACC \Rightarrow aabBDCC \Rightarrow aabBCDC \Rightarrow aabcDC$$
$$\Rightarrow aabcCD \Rightarrow aabccD \Rightarrow aabccd$$

بنابراین رشته a^2bc^2d تولید شد. زبان گرامر $\{a^n b^m c^n d^m : n, m > 0\}$ است.

۴- با تولید چند جمله گرامر می توان حدس زد که زبان گرامر $\{a^{2^n} : n \geq 1\}$ است.

$$S \Rightarrow ABaC \Rightarrow AaaBC \Rightarrow AaaE \Rightarrow AaEa \Rightarrow AEaa \Rightarrow aa$$

۵- بله - چون این زبان را با LBA می توان پذیرفت.

۶- بله- زبان‌های بازگشتی شمارش پذیر، تحت بستار ستاره ای بسته هستند، پس زبان L_1^* بازگشتی شمارش پذیر است. همچنین زبان‌های بازگشتی شمارش پذیر، تحت اجتماع بسته هستند، پس زبان $L_2 \cup L_1^*$ بازگشتی شمارش پذیر است. و در نهایت زبان‌های بازگشتی شمارش پذیر، تحت الحاق بسته هستند، پس زبان $L_1.(L_2 \cup L_1^*)$ بازگشتی شمارش پذیر است.

۷- بله- چون ماشین تورینگ بعد از چند گام مشخص، رشته w را می پذیرد، پس زبان L ،تصمیم پذیر است.

۸- خیر- تحمیل این شرط باعث نمی شود که قدرت تورینگ کم شود.

۹- بله- محدود کردن تعداد نمادهای ماشین، ممکن است قدرت ماشین را کم کند.

۱۰- بله- اگر در یک ماشین تورینگ، هد نتواند به سمت چپ حرکت کند، قدرت ماشین تغییر خواهد کرد.

۱۱- هر دو گزاره درست هستند.

۱۲- بله- مکمل هر مجموعه تصمیم پذیر، تصمیم پذیر است، بنابراین B^c تصمیم پذیر است و چون مجموعه های تصمیم پذیر تحت اشتراک بسته هستند، بنابراین $A \cap B^c$ تصمیم پذیر است.

فصل ۹

تصمیم پذیری- کاهش پذیری

زبان هایی وجود دارند که نمی توان تعلق یا عدم تعلق یک رشته به آن زبان ها را تعیین کرد، یعنی دارای الگوریتم عضویت نمی باشند و تصمیم ناپذیرند. در این فصل به چند نمونه از این زبان ها اشاره می شود.

زبان های تصمیم ناپذیر

با فرض اینکه M و M_1 و M_2 یک ماشین تورینگ باشند، زبان های زیر تصمیم پذیر(Undecidable) نیستند.

$L = \{< M, w >:$ M رشته ورودی w را می پذیرد. $\}$

$L = \{< M, w >:$ M روی رشته ورودی w متوقف شود. $\}$

$L=\{<M>:$ دو رشته به طول یکسان دارد. $L(M)$ $\}$

$L=\{<M>:$ یک زبان منظم است. $L(M)$ $\}$

$L=\{<M>:$ است. $L(M)=\phi$ $\}$

$L=\{<M>:$ متناهی است. $L(M)$ $\}$

$L=\{<M>:$ بیش از 10 عضو دارد. $L(M)$ $\}$

$L=\{<M>:$ هر رشته به طول 10 را می پذیرد. $\}$

$L=\{<M>:$ است. $L(M)=\sum^*$ $\}$

$L=\{<M>:$ ماشین رشته w^R را می پذیرد اگر رشته w را بپذیرد. $\}$

$L=\{<M,w>:$ رشته ای به طول k دارد. $L(M)$ $\}$

$L=\{<M_1,M_2>:$ است. $L(M_1)=L(M_2)$ $\}$

$L=\{<M_1,M_2>:$ است. $L(M_1)\subseteq L(M_2)$ $\}$

با فرض اینکه G و G_1 و G_2 یک گرامر مستقل از متن باشند، زبان‌های زیر تصمیم پذیر نیستند:

$L=\{<G>:$ مبهم است. G $\}$

$L=\{<G>:L(G)=\sum^*\}$

$L=\{<G_1,G_2>:L(G_1)=L(G_2)\}$

$L=\{<G_1,G_2>:L(G_1)\subseteq L(G_2)\}$

$L=\{<G_1,G_2>:L(G_1)\bigcap L(G_2)=\phi\}$

زبان‌های زیر تصمیم پذیر نیستند:

$L=\{<G_1,G_2>: L(G_1)\bigcap L(G_2)=\phi,$ گرامر منظم است G_2 و G_1، گرامر بدون محدودیت و $\}$

$L=\{<G>:L(G)=L(G)^R,$ یک گرامر بدون محدودیت است G $\}$

$L=\{<G>: L(G)=\phi$ یک گرامر بدون محدودیت است. G \}

$L=\{<M>: L(M)=\sum^*,$ یک ماشین پشته ای است. M \}

$L=\{<M>: L(M)=\phi,$ یک LBA است. M \}

زبان های تصمیم پذیر(بازگشتی)

با فرض اینکه M و M_1 و M_2 یک dfa باشند، زبان‌های زیر تصمیم پذیر هستند:

$L=\{<M>: L(M)=\phi\}$

$L=\{<M>:$ ماشین $\sum^*$ را تشخیص می‌دهد.\}

$L=\{<M>:$ M هیچ رشته‌ای که تعداد فرد یک داشته باشد را نمی‌پذیرد.\}

$L=\{<M,w>:$ M رشته ورودی w را می‌پذیرد. \}

$L=\{<M,w>:$ اگر M رشته w را بپذیرد، آنگاه w^R را نیز می‌پذیرد. \}

$L=\{<M_1,M_2>:$ $L(M_1)=L(M_2)$ است. \}

با فرض اینکه G یک گرامر مستقل از متن باشد، زبان‌های زیر تصمیم پذیر هستند:

$L=\{<G>: L(G)=\phi\}$

$L=\{<G>:$ گرامر G ، رشته λ را تولید می کند.\}

$L=\{<G>:$ $L(G)$ متناهی است، یعنی $|L(G)<\infty|$ \}

$L=\{<G>:$ رشته $w\in L(G)$ با طول کمتر از n وجود دارد.\}

$L=\{<G,w>:$ گرامر G، رشته w را تولید می کند.\}

زبان‌های زیر تصمیم پذیر(بازگشتی) هستند:

$L=\{<G>: L(G)=\sum^*$ و G یک گرامر منظم است \}

$L=\{<R,w>:$ R عبارت منظمی است که رشته ورودی w را تولید می کند. \}

$L=\{<R,S>: L(R)\subseteq L(S)$ و R و S دو عبارت منظم هستند \}

$L = \{<M, w>:$ M یک NFA است که رشته ورودی w را می پذیرد. $\}$

$L = \{<M, w>:$ M یک ماشین تورینگ بوده و پس از 10 گام رشته ورودی w را می پذیرد. $\}$

$L = \{<M, w>:$ M یک LBA بوده که رشته ورودی w را می پذیرد. $\}$

$L = \{<M>:$ M یک اتوماتای پشته ای است که دارای حالت بی فایده است. $\}$

$L = \{<M>:$ M یک ماشین تورینگ است که دارای حالت بی فایده است. $\}$

حالت بی فایده در یک ماشین، حالتی است که به ازای هیچ رشته ورودی، ماشین به آن حالت وارد نمی شود.

تصمیم پذیری در زبان های منظم

با فرض اینکه L و L_1 و L_2 زبان منظم باشند، الگوریتم های وجود دارد که تعیین کند که:

۱- آیا $\lambda \in L$ ؟

۲- آیا $w^R \in L$ ؟

۳- آیا $L = L^*$ ؟

۴- تعیین تهی بودن L.

۵- تعیین متناهی یا نامتناهی بودن L.

۶- تعیین وجود یا عدم وجود w در L . $(w \in \sum^*)$

۷- تعیین $L_1 = L_2$.

۸- آیا $w \in L_1 - L_2$ ؟

۹- آیا $L_1 \subseteq L_2$ ؟

۱۰- آیا $L_1 = L_{1/} L_2$ ؟

۱۱- آیا $L = L_1 L_2$ ؟

کاهش پذیری

کاهش دادن (reduction) ،روشی برای تبدیل یک مسئله به مسئله دیگری است به طوری که حل مسئله دوم به حل مسئله اولیه کمک کند. به چند مثال توجه کنید:

۱- مسئله اندازه گیری مساحت یک مستطیل به مسئله اندازه گیری طول و عرض آن کاهش می یابد.

۲- مسئله حل یک سیستم معادلات خطی به معکوس کردن یک ماتریس کاهش می یابد.

۳- مسئله پیدا کردن مسیر در شهر، به مسئله خرید نقشه شهر کاهش می یابد.

در نظریه محاسبه پذیری، اگر A به B کاهش پذیر بوده و B تصمیم پذیر باشد،آنگاه A نیز تصمیم پذیر خواهد بود. همچنین اگر A تصمیم پذیر نبوده و به B کاهش یابد،آنگاه B نیز تصمیم پذیر نخواهد بود. برای نشان دادن اینکه یک مسئله تصمیم ناپذیر است،نشان می دهیم که یک مسئله تصمیم ناپذیر به این مسئله کاهش یافته است.

زبان A ، نگاشت کاهش پذیر به زبان B می باشد و به صورت $A \le_m B$ نوشته می شود، اگر یک تابع محاسباتی $f : \Sigma^* \to \Sigma^*$ موجود باشد که برای هر رشته w : $w \in A \Rightarrow f(w) \in B$.

یک نگاشت کاهش پذیر از A به B، روشی برای تبدیل سئوالات مربوط به عضویت A به سئوالات مربوط به عضویت B می باشد.

اگر مسئله ای، نگاشت کاهشی به مسئله دیگری که قبلا حل شده داشته باشد، می توانیم مسئله اول را نیز حل شده فرض کنیم.

چند نکته:

۱- رابطه $\le_m$ یک رابطه متعدی(ترایا) است.

۲- اگر $A \le_m B$ بوده و B تصمیم پذیر باشد، آنگاه A نیز تصمیم پذیر است.

۳- اگر $A \le_m B$ بوده و A تصمیم ناپذیر باشد، آنگاه B نیز تصمیم ناپذیر است.

۳-اگر $A \le_m B$ بوده و B تشخیص پذیر تورینگ باشد، آنگاه A نیز تشخیص پذیر تورینگ است.

۴- اگر $A \leq_m B$ بوده و A تشخیص پذیر تورینگ نباشد، آنگاه B تشخیص پذیر تورینگ نخواهد بود.

۵- اگر A تشخیص پذیر تورینگ باشد و $A \leq_m \overline{A}$ باشد، آنگاه A تصمیم پذیر است.

تعریف: زبان A را کاهش پذیر تورینگ (Turing Reducible) به زبان B گویند و با $A \leq_T B$ نشان می دهند، اگـر A نسـبت به B تصمیم پذیر باشد. اگر $A \leq_T B$ بوده و B تصمیم پذیر باشد،آنگاه A نیز تصمیم پذیر است.

برای هر دو زبان دلخواه A و B ،زبانی مانند C وجود دارد بطوریکه : $A \leq_T C$ و $B \leq_T C$

تمرین‌های فصل ۹

۱- کدام یک از زبان های زیر تصمیم پذیر هستند؟

$L_1 = \{< M >:$ یک ماشین تورینگ بوده و $L(M)$ متناهی است. $M\}$

$L_2 = \{< M, w >:$ یک ماشین تورینگ بوده و رشته ورودی w را می پذیرد. $M\}$

$L_3 = \{< M_1, M_2 >:$ و M_2 دو ماشین تورینگ بوده و $L(M_1) = L(M_2)$ است. $M_1\}$

$L_4 = \{< M, w >:$ یک ماشین تورینگ بوده و پس از 10 گام رشته ورودی w را می پذیرد. $M\}$

۲- کدام یک از زبان های زیر تصمیم پذیر هستند؟

$L_1 = \{< G >:$ یک گرامر مستقل از متن بوده و $L(G) = \phi$ است. $G\}$

$L_2 = \{< M >:$ یک ماشین پشته ای بوده و $L(M) = \Sigma^*$ است. $M\}$

$L_3 = \{< G >:$ یک گرامر مستقل از متن بوده و $L(G) = \Sigma^*$ است. $G\}$

$L_4 = \{< G_1, G_2 >:$ و G_2 دو گرامر مستقل از متن بوده و $L(G_1) \cap L(G_2) = \phi$ است. $G_1\}$

۳- در جدول زیر، مسائلی که تصمیم پذیر هستند، را مشخص کنید.

	$w \in L$	$L = \phi$	$L = \Sigma^*$	متناهی بودن L

منظم				
مستقل از متن				
حساس به متن				
بازگشتی				
بازگشتی شمارش پذیر				

۴- دو گزاره زیر چه رابطه ای با هم دارند؟

الف- ابهام در گرامرهای مستقل از متن یک مساله تصمیم ناپذیر است.

ب- حداقل یک مساله تصمیم ناپذیر قابل کاهش(Reducable) به مساله ابهام در گرامرهای مستقل از متن وجود دارد.

۵- برشمارنده (enumerator) را تعریف کنید.

۶- کدام گزاره(ها) درست است؟

الف- یک زبان تشخیص پذیر تورینگ است اگر و تنها اگر برشمارنده ای برای برشمردن آن موجود باشد.

ب- یک زبان تصمیم پذیر است اگر و تنها اگر یک بر شمارنده برای بر شمردن آن زبان با ترتیب فرهنگ لغت(lexicographic) موجود باشد.

۷- کدام گزینه نادرست است؟

۱) اگر $A \leq_m B$ بوده و B تصمیم پذیر باشد، آنگاه A نیز تصمیم پذیر است.

۲) اگر $A \le_m B$ بوده و A تصمیم ناپذیر باشد، آنگاه B نیز تصمیم ناپذیر است.

۳) اگر $A \le_m B$ بوده و B تشخیص پذیر تورینگ باشد، آنگاه A نیز تشخیص پذیر تورینگ است.

۴) اگر $A \le_m B$ بوده و A تشخیص پذیر تورینگ باشد، آنگاه B تشخیص پذیر تورینگ نخواهد بود.

۸- برای کدام یک از گزاره های زیر، الگوریتم وجود دارد؟

الف- تعیین مستقل از متن بودن $L_1 \cap L_2$ (L_1 مستقل از متن و L_2 منظم است)

ب- وجود عضو مشترک در میان یک زبان منظم و یک زبان مستقل از متن.

۹- برای کدام یک از گزاره های زیر، الگوریتم وجود دارد؟

الف- آیا $L = tail(L)$؟

ب- تشخیص زبان های منظم دو طرفه.

ج- آیا L حاوی رشته w ای است که $\widehat{w}$ زیر رشته ای از آن باشد؟

۱۰- اگر L یک زبان منظم باشد، برای کدام یک از گزاره های زیر، الگوریتم وجود دارد؟

الف- آیا L حاوی رشته‌ای با طول زوج است یا خیر؟

ب- آیا رابطه $|L| \ge 5$ در هر زبان منظم برقرار است یا خیر؟

ج- آیا L حاوی تعداد متناهی رشته با طول زوج است یا خیر؟

۱۱- اگر L_1 یک زبان مستقل از متن و L_2 یک زبان منظم باشند، نشان دهید الگوریتمی برای تعیین اینکه آیا L_1 و L_2 عنصر مشترکی دارند یا خیر، وجود دارد.

۱۲- نشان دهید که الگوریتمی وجود دارد که مشخص کند که آیا زبان تولید شده توسط یک گرامر مستقل از متن، شامل کلماتی با طول کمتر از n هست؟

پاسخ تمرین‌های فصل ۹

۱- زبان L_4 ، تصمیم پذیر است.

۲- زبان L_1 ، تصمیم پذیر است.

۳- در جدول زیر، مسائلی که تصمیم پذیر هستند، با یک تیک مشخص شده اند:

	$w \in L$	$L = \phi$	$L = \Sigma^*$	متناهی بودن L
منظم	✓	✓	✓	✓
مستقل از متن	✓	✓		✓
حساس به متن	✓	✓		
بازگشتی	✓			
بازگشتی شمارش پذیر				

به طور نمونه اگر L یک زبان مستقل از متن باشد، مسئله $L = \Sigma^*$ تصمیم پذیر نیست، چون برای تست این موضوع باید عضویت همه رشته های عضو Σ^* در L را بررسی کرد که چون تعداد آنها متناهی نمی‌باشد، این عمل شدنی نیست.

۴- دو گزاره معادل‌اند.

۵- برشمارنده یک ماشین تورینگ با یک چاپگر است. یک برشمارنده با یک نوار خالی شروع می کند. زبـان برشمرده شده، مجموعه تمامی رشته هایی است که در نهایت چاپ می شود. البته ممکن است بعضی رشته ها را تکراری چاپ کند. اگر برشمارنده متوقف نشود، ممکن است لیستی شامل بی نهایت رشته چاپ کند.

۶- هر دو گزاره درست است.

۷- گزینه ۴ نادرست است. درست آن به صورت زیر است:

اگر $B \leq_m A$ بوده و A تشخیص پذیر تورینگ نباشد، آنگاه B تشخیص پذیر تورینگ نخواهد بود.

۸- برای هر دو گزاره الگوریتم وجود دارد.

۹- برای هر سه گزاره الگوریتم وجود دارد. در رابطه با هر گزاره باید بدانیم که:

الف- تابع $tail(L)$ به صورت $tail(L)=\{v: uv \in L, u,v \in \Sigma^*\}$ تعریف می شود.

ب- یک زبان در صورتی دو طرفه نامیده می شود که $L = L^R$.

ج- L حاوی رشته w ای است که $\widehat{w}$ زیر رشته ای از آن باشد، یعنی: $w = u\widehat{w}v$. $(u,v \in \Sigma^*)$

۱۰- برای هر سه گزاره الگوریتم وجود دارد.

۱۱- زبان $L_1 \cap L_2$ را تشکیل می دهیم و گرامر مستقل از متن G را به آن نسبت می دهیم. حال تعیین می کنیم که آیا این زبان تهی است یا خیر.

۱۲- گرامر را به فرم نرمال چامسکی در می آوریم. عضویت تمام رشته هایی با طول کمتر از n را با الگوریتم CYK ، بررسی می کنیم.

دسته‌بندی موضوعی آموزش‌های فرادرس، در ادامه آمده است:			
برنامه‌نویسی برنامه نویسی – کلیک (+)	آموزش‌های دانشگاهی و تخصصی آموزش های دانشگاهی و تخصصی – کلیک (+)	هوش مصنوعی و یادگیری ماشین هوش مصنوعی و یادگیری ماشین – کلیک (+)	مهندسی برق الکترونیک و روباتیک مهندسی برق الکترونیک و روباتیک – کلیک (+)
دروس دانشگاهی دروس دانشگاهی – کلیک (+)	مباحث مشترک مباحث مشترک – کلیک (+)	مهارت‌های دانشگاهی مهارت های دانشگاهی – کلیک (+)	نرم‌افزارهای تخصصی نرم افزارهای تخصصی – کلیک (+)
مهندسی نرم‌افزار مهندسی نرم افزار – کلیک (+)	نرم‌افزارهای عمومی نرم افزارهای عمومی – کلیک (+)	طراحی و توسعه وب طراحی و توسعه وب – کلیک (+)	آموزش‌های عمومی آموزش های عمومی – کلیک (+)

منبع مطالعاتی تکمیلی مرتبط با این کتاب

آموزش نظریه زبان‌ها و ماشین‌ها

در این درس با سه موضوع "زبان، گرامر و ماشین" آشنا می‌شوید. این درس پیش نیاز درس طراحی کامپایلر است. با یادگیری زبان‌ها و گرامرها می‌توانید نحوه کار کامپایلر و همچنین طراحی زبان‌های برنامه‌سازی را متوجه شد. یادگیری این درس بدون مدرس کار ساده‌ای نمی‌باشد و ما در این آموزش تجربه حداقل پانزده سال تدریس این درس را در اختیار شما گذاشته‌ایم. به امید اینکه دعای خیری برای ما شود.

مدرس: مهندس فرشید شیر افکن

مدت زمان: ۹ ساعت

جهت مشاهده آموزش ویدئویی این آموزش – کلیک کنید **faradars.org /fvsft110**