



فصل پنجم

تجزیه پایین به بالا

۵.۱ مقدمه

در روش تجزیه پائین به بالا مراحل تجزیه از ترمهای پایانی درون جمله داده شده آغاز، و به سر ترم گرامر خاتمه می یابد. تجزیه گرهایی پائین به بالا گرامرهای گسترده تری نسبت به تجزیه گرهایی بالا به پائین را می پوشانند. در این فصل انواع روشهای اتوماتیک تجزیه پائین به بالا مطرح خواهد شد و چگونگی ترسیم جداول تجزیه، به روشهایی موسوم به LR، LALR، SLR مطرح میگردد. نکته قابل توجه استفاده از گرامرهای مبهم است که موضوع آخر این فصل را به خود اختصاص میدهد.

روش LR(k) برای تجزیه پائین به بالا با استفاده از k ترم بعدی در جمله مورد کامپایل در هر مرحله از تجزیه استفاده می نماید. چون عمل خواندن لغات از داخل جمله ورودی مورد کامپایل از چپ به راست انجام میشود حرف L که مخفف Left to Right می باشد، در نام این روش ظاهر شده است. حرف R مخفف Rightmost derivation یا استنتاج راست است. کلمه LALR مخفف Lookahead LR است. روش LALR(k) نیز برای دسته ای محدودتر از گرامرها نسبت به روش LR(k) مورد استفاده قرار میگیرد. کلمه SLR مخفف Simple LR است. در این فصل روشهای LR(1)، LALR(1) و SLR(1) مورد بررسی قرار خواهد گرفت.



۵.۲ اصول تجزیه پایین به بالا

اصولاً در روش تجزیه بالا به پایین پس از اینکه تحلیلگر نحوی لغات را دریافت نمود دو عمل را ممکن است انجام دهد. تحلیلگر یا لغت را مستقیماً به داخل یک پشته بنام پشته تجزیه انتقال میدهد و یا اینکه بر اساس لغت در یافت شده ترمهای بالای پشته که با سمت راست یک قاعده از قواعد گرامر مطابقت دارند را از بالای پشته خارج نموده، با ترم میانی سمت راست قاعده جایگزین میکند. با جایگزینی ترمهای بالای پشته با ترم میانی سمت چپ قاعده مربوطه در واقع چند ترم بالای پشته به یک ترم کاهش یافته اند. لذا، این عمل را در اصطلاح عمل کاهش یا Reduce گویند. عمل انتقال لغات دریافتی از تحلیلگر لغوی به داخل پشته تجزیه را در اصطلاح عمل انتقال یا Shift گویند.

عمل کاهش یا Reduce بر اساس یک قاعده از گرامر انجام می گیرد. برای نمونه ترمهای بالای پشته تجزیه که عیناً مشابه ترمهای سمت راست قاعده شماره n هستند، به ترم سمت چپ قاعده که یک ترم میانی یا سرترم گرامر است کاهش داده می شود. این عمل را بطور خلاصه با دستور R_n مشخص میکنند. دستور العمل R_n مشخص میکند که عمل Reduce بر اساس قاعده شماره n انجام میشود.

اصولاً در روش تجزیه پایین به بالا عمل خواندن لغات مثل قبل از چپ به راست جمله داده شده انجام می شود. ترمها بر اساس قواعد زبان، دسته بندی و به یک ترم میانی در سمت راست قواعد کاهش می یابند و یا در اصطلاح خارجی Reduce داده میشوند. یک جمله یا برنامه در داخل یک فایل متن یا در اصطلاح Text قرار میگیرد. در انتهای هر فایل متن علامت خاتمه فایل ظاهر میشود. هر جمله داده شده نهایتاً در صورت صحت از لحاظ فرم گرامری به سرترم گرامر کاهش داده میشود. بنابراین پس از سرترم گرامر همواره انتظار مشاهده علامت خاتمه فایل می رود.

علامت خاتمه فایل در اینجا با کاراکتر $\$$ مشخص می شود. جهت کسب اطمینان از ظهور علامت خاتمه فایل پس از سرترم گرامر چنانچه برای نمونه سرترم گرامر s باشد، قاعده

$$S' \rightarrow S \$$$

را به ابتدای گرامر افزوده، در اصطلاح گرامر را به فرم توسعه یافته یا Extended تبدیل می نمایند. بنابراین به ابتدای گرامر عبارات چهار عمل اصلی با سرترم E ، قاعده:

$$E' \rightarrow E \$$$

افزوده شده، در اصطلاح گرامر عبارات به فرم توسعه یافته تبدیل می شود.

$$0 \quad E' \rightarrow E \$$$

$$1 \quad E \rightarrow E + T$$

$$2 \quad E \rightarrow E - T$$



- 3 $E \rightarrow T$
- 4 $T \rightarrow T * F$
- 5 $T \rightarrow T / F$
- 6 $T \rightarrow F$
- 7 $F \rightarrow Id$
- 8 $F \rightarrow No$
- 9 $F \rightarrow (E)$

بنا بر گرامر فوق ، صحت عبارت $(a - b) * c / d$ سنجیده میشود . برای این منظور از يك پشته تجزیه ، جهت حفظ فرمهاي جمله اي استفاده مي شود .

پشته تجزیه	ورودی	عملیات
(A	(a-b)*c/d \$	s,s
(A	-b)*c/d \$	R7,R6,R3
(A)*c/d \$	S,S
(A)*c/d \$	R7
(E-T)*c/d \$	R6
(E)*c/d \$	R2
(E))*c/d \$	S
F)*c/d \$	R9
T	/d \$	R6
T*c	/d \$	S,S
T*F	/d \$	R7
T	\$	R4
T/d	\$	S,S
T/F	\$	R7
T	\$	R5
E		R3,S,R0

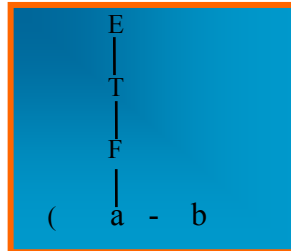
شکل 1 و 4- مراحل تجزیه جمله $(a - b) * c / d$

در جدول فوق ، نکته قابل توجه انتخاب عمل کاهش و یا انتقال در هر مرحله از تجزیه بالا به پایین است. تصمیم گیری در مورد انتخاب عمل کاهش (Reduce) و یا انجام عمل انتقال (Shift) ، به موقعیت کنونی پشته و فرم جمله ای موجود در آن و بالاخره چگونگی ترم موجود بر سر ورودی که در اصطلاح ترم پیش بینی یا Look ahead نامیده می شود ، وابسته است . برای نمونه در شروع عمل تجزیه ابتدا ، و سپس 'a' از ورودی خوانده می شود. چون ترم بعدی موجود بر سر ورودی اکنون علامت منها میباشد و طبق گرامر و بنابر قاعده :

$$2 \quad E \rightarrow E - T$$



حتماً قبل از منها يك E باید وجود داشته باشد، تصمیم به کاهش a بر اساس قاعده شماره هفت و یا بطور مختصر تصمیم به انجام دستور العمل R7 و سپس R6 و نهایتاً R3 گرفته شد. لذا، تا این مرحله درخت تجزیه بصورت زیر ایجاد میشود.



در مرحله بعدی با انجام عمل S یا در واقع عمل انتقال لغت '، از سر ورودی خوانده میشود. اکنون طبق گرامر پس از علامت '، باید در ورودی T ظاهر شود. لذا، ترم بعدی یعنی 'b' از ورودی به داخل پشته تجزیه انتقال یا در اصطلاح Shift داده میشود.

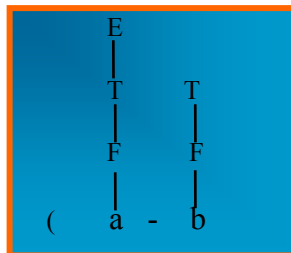
حالا، بر سر ورودی ترم '، قرار گرفته است. قبل از '، بر طبق قاعده :

$$9 \quad F \rightarrow (E)$$

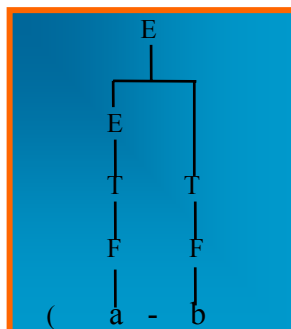
باید E ظاهر شود. تا این مرحله طبق جدول تجزیه بر سر پشته فرم جمله ای '(E - b'

قرار دارد و ترم پیش بینی '، است. لذا، جهت رسیدن به E قبل از '، عمل R7 و سپس عمل R6 انجام میشود. بنابراین فرم جمله ای موجود در پشته تجزیه بصورت '(E - T'

تبدیل میشود. تا این مرحله از تجزیه پائین به بالا درخت تجزیه بصورت زیر است. فرمهای جمله ای ایجاد شده نیز در داخل پشته تجزیه در بالا مشخص می باشند.



حالا ، ترم پیش بینی '، و بر سر پشته تجزیه ترم T موجود است. اما بلافاصله T با E جایگزین نشده ، دستور العمل R3 اجراء نمی شود . باید توجه نمود که همواره تنها ترم نیش بینی عامل تصمیم گیرنده نیست بلکه ، چگونگی فرم جمله ای در بالای پشته تجزیه نیز در تصمیم گیری و انتخاب نوع عمل کاهش و یا انتخاب عمل انتقال موثر است . لذا، در این مرحله عمل R2 انجام شده ، بر سر پشته تجزیه E - T به E بر طبق قاعده شماره 2 کاهش داده میشود. تا این مرحله از تجزیه پائین به بالا درخت تجزیه بصورت زیر است :

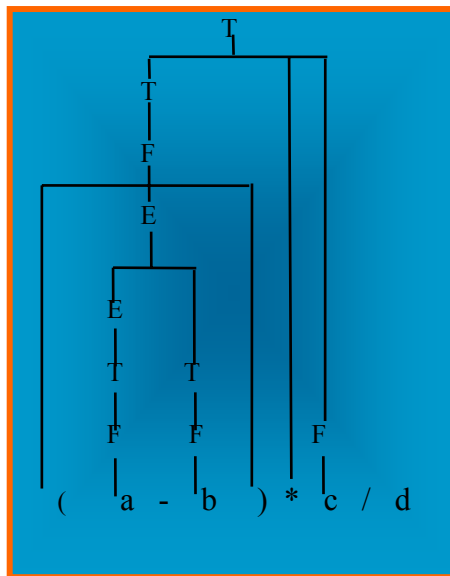




در مرحله بعدی، با انجام عمل s به داخل پشته تجزیه انتقال داده میشود. اکنون بر سر ورودی لغت $*$ قرار دارد. لذا، چون قبل از ضرب طبق قاعده

$$4 \quad T \rightarrow T * F$$

باید T وجود داشته باشد عملیات $R9$ و سپس $R6$ انجام میشود. حالا با انجام دو عمل S متوالی ترم های $*$ و C به داخل پشته تجزیه انتقال داده میشوند. اکنون بر سر پشته تجزیه فرم جمله ای $T * C$ و ترم نیش بینی $/$ است. لذا، عملیات $R7$ ، $R6$ و $R4$ متوالیاً انجام میشوند. سپس با انجام دو عمل S لغت $/$ و d به داخل پشته انتقال داده میشوند. تا این مرحله درخت تجزیه بصورت زیر است.



به همین ترتیب تا انتهای جمله داده شده عمل تجزیه ادامه دارد.

۵.۳ طرح روشی برای ایجاد جدول تجزیه $LR(1)$

در این بخش چگونگی ایجاد الگوریتم تجزیه پایین به بالا استدلال میشود. الگوریتم حاصل بطور خلاصه در انتها ارائه شده است. الگوریتم حاصل تجزیه $LR(1)$ ¹

¹ همانگونه که در مقدمه این فصل توضیح داده شد، در کلمه $LR(1)$ حرف L مخفف Left to Right و R مخفف Rightmost derivation است.



نامیده می شود. این الگوریتم مبتنی بر تحلیل روش تجزیه که در بالا توضیح داده شد ایجاد میشود.

5.3.1 تولید الگوریتم

در بخش قبل مشاهده کردید که در حالت کلی برای انجام عمل تجزیه نیاز به استفاده از یک پشته تجزیه می باشد. دو عمل صورت میگیرد یکی انتقال و دیگری کاهش. عمل انتقال و کاهش بر اساس چگونگی ترم پیش بینی بر سر ورودی و وضعیت کنونی پشته تجزیه مشخص می شد. در بعضی از تجزیه گرهایی LR نیاز به بیش از یک ترم پیش بینی برای عمل تجزیه است، برای نمونه اگر حداکثر K ترم پیش بینی نیاز باشد، تجزیه گر را $LR(k)$ می نامند.

اکنون با یک استدلال ساده نشان داده خواهد شد که چگونه میتوان بصورت اتوماتیک بر اساس جدول تجزیه که در مورد آن بحث خواهد شد عمل تجزیه پائین به بالا را انجام داد. برای نمونه گرامر زیر را در نظر بگیرید:

- 0 $S' \rightarrow SS$
- 1 $S \rightarrow aBb$
- 2 $S \rightarrow AB$
- 3 $A \rightarrow bA$
- 4 $A \rightarrow b$
- 5 $B \rightarrow Bb$
- 6 $B \rightarrow a$

بر طبق گرامر فوق نهایتاً در بالای درخت تجزیه باید سرترم s' ظاهر شود. بنابراین آخرین مرحله در تجزیه بالا به پائین کاهش ss به s' است. اما قبل از اینکه بتوان این عمل را انجام داد در ورودی ابتدا باید ترم s ظاهر شود. بنابراین میتوان گفت در تجزیه پائین به بالا، در آغاز کار تحلیلگر نحوی در انتظار کاهش ورودی یا به عبارت دیگر برنامه یا جمله مورد کامپایل به s است. لذا، میتوان این وضعیت انتظار را بصورت زیر برای تحلیلگر مشخص نمود.

$$S' \rightarrow \bullet SS$$

در این وضعیت، نشان میدهد که در انتظار s باید بود. اما برای تشکیل s باید طبق گرامر یا عمل R_1 یا عمل R_2 انجام شود. یعنی طبق گرامر

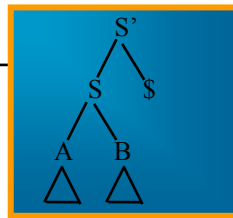
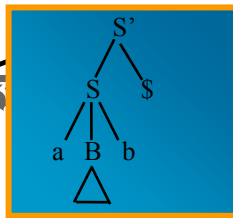
$$1 \quad S \rightarrow$$

aBb

$$2 \quad S \rightarrow$$

AB

رشته aBb یا رشته AB در ورودی ظاهر شود. به عبارت دیگر قبل از خاتمه عمل تجزیه باید یکی از درختهای زیر ایجاد شوند.



فصل 5- تجزیه پایین به بالا

پس از مشاهده سرترم s طبق وضعیت $S \rightarrow \bullet S \$$ انتظار می‌رود که علامت خاتمه فایل $\$$ در ورودی ظاهر شود. پس از مشاهده $\$$ است که می‌توان عمل R_0 را انجام داد. بنابراین تا این مرحله می‌توان گفت که وضعیت در شروع عمل تجزیه بصورت زیر می‌تواند باشد.

$$\begin{aligned} S' &\rightarrow \bullet S \$ \\ S &\rightarrow \bullet a B b , \$ \\ S &\rightarrow \bullet A B , \$ \end{aligned}$$

در بالا علامت $\$$ توسط ویرگول از دو گسترش متفاوت s جدا شده است. به این ترتیب مشخص شده که پس از کاهش سمت راست این دو قاعده به s انتظار می‌رود که در ورودی $\$$ ظاهر شود. یا به عبارت دیگر انتظار می‌رود که پس از تشخیص s ترم پیش بینی $\$$ باشد. این موضوع را وضعیت $S \rightarrow \bullet S \$$ مشخص نمود.

بنابراین تا کنون مشخص شده است که در حالت شروع و قبل از اینکه هیچ لغتی از جمله مورد کامپایل در ورودی ظاهر شود، تحلیلگر نحوی انتظار S' و لذا جهت تشکیل s' در آغاز انتظار کاهش ورودی به s را دارد. پس از مشاهده سرترم s انتظار می‌رود که علامت خاتمه فایل $\$$ در ورودی ظاهر شود.

برای مشاهده s در آغاز کار انتظار مشاهده a و یا تشکیل ترم میانی A می‌رود. طبق وضعیت :

$$S \rightarrow \bullet a B b , \$$$

ابتدا a سپس B و پس از آن b باید در ورودی ظاهر شوند تا بتوان عمل R_1 را انجام داد و s را ایجاد کرد. البته عمل R_1 در صورتی موثر است که در سر ورودی $\$$ وجود داشته باشد. به همین ترتیب طبق وضعیت $S \rightarrow \bullet A B , \$$ باید ابتدا A ایجاد شود پس از آن B تا بتوان عمل R_2 را انجام داد. بنابراین باید ابتدا در ورودی A تشخیص داده شود و پس از تشخیص A باید حتماً در سر ورودی ترمی متعلق به مجموعه $First(B)$ ظاهر شود تا بتوان مطمئن شد که می‌توان B را پس از A دید.

برای ایجاد A نیز طبق گرامر باید یا عمل R_3 و یا R_4 انجام شود. و بعد از انجام عمل حالا بر طبق انتظاری که در بالاتر طبق وضعیت $S \rightarrow \bullet A B , \$$ می‌رفت، تحلیلگر نحوی باید یک ترم متعلق به $First(B)$ را در ورودی مشاهده کند تا بتواند به کار خود ادامه دهد و B را در ورودی تشخیص دهد. بنابراین جهت رسیدن به A یکی از دو وضعیت زیر در آغاز کار و قبل از دریافت هر گونه لغتی از تحلیلگر لغوی پیش می‌آید.

$$\begin{aligned} A &\rightarrow \bullet b A , First(B) \\ A &\rightarrow \bullet b , First(B) \end{aligned}$$



بنابراین در حالت شروع و قبل از دریافت اولین لغت از تحلیلگر لغوی وضعیت های مورد انتظار تحلیلگر نحوی بصورت زیر میباشد.

$I_0 : S' \rightarrow \bullet S \$$
$S \rightarrow \bullet a B b , \$$
$S \rightarrow \bullet A B , \$$
$A \rightarrow \bullet b A , First(B)$
$A \rightarrow \bullet b , First(B)$

حالا فرض کنید که اولین ترم پایانی یا اولین لغت از تحلیلگر لغوی دریافت شود. چه اتفاقی میتواند رخ دهد؟ طبق انتظار تحلیلگر که در بالا مشخص شده است باید این لغت b یا a باشد. طبق وضعیت:

$$'S \rightarrow \bullet a B b , \$'$$

انتظار مشاهده a و طبق وضعیتهای:

$$'A \rightarrow \bullet b A , First(B)'$$

$$'A \rightarrow \bullet b , First(B)'$$

در آغاز انتظار مشاهده b میرود. اگر در ورودی b ظاهر شود حالت جدید زیر حاصل میشود:

$II : A \rightarrow b \bullet A , First(B)$
$A \rightarrow b \bullet , First(B)$

حالا اگر در ورودی $First(A)$ ظاهر شود به سراغ گسترش A در ادامه وضعیت $b \bullet$ باید رفت و اگر $First(B)$ در ورودی ظاهر شود باید طبق قاعده $A \rightarrow b$ عمل کاهش b به A را انجام داد. بنابر $'A \rightarrow b \bullet A , First(B)'$ برای مشاهده A در حالت II انتظار دیدن A دومی هم در سمت راست وضعیت میرود. اما برای مشاهده A در ورودی طبق قواعد 3 و 4 گرامر باید $b A$ یا b در ورودی ظاهر شود. پس از این A دوم $First(B)$ باید در ورودی ظاهر شود. بنابراین حالت II بصورت زیر تبدیل میشود:

$II : A \rightarrow b \bullet A , First(B)$
$A \rightarrow b \bullet , First(B)$
$A \rightarrow \bullet b A , First(B)$
$A \rightarrow \bullet b , First(B)$

در حالت II طبق وضعیت $'A \rightarrow b \bullet A , First(B)'$ انتظار مشاهده A پس از b در سمت راست وضعیت میرود. پس از A انتظار مشاهده $First(B)$ میرود. بنابراین، برای تشکیل A در ورودی دو وضعیت:

$$A \rightarrow \bullet b A , First(B)$$

$$A \rightarrow \bullet b$$

$, First(B)$

به حالت II افزوده شده است. به این ترتیب در این حالت پس از اینکه طبق یکی از دو وضعیت فوق در ورودی A تشخیص داده شد وضعیت $'A \rightarrow b \bullet A , First(B)'$ که در انتظار تشکیل A در ورودی است تغییر حالت داده جالت جدید I_5 تشکیل میشود.



$$I5: A \rightarrow b A \bullet \text{ First}(B)$$

حالا می توان عمل R_3 را انجام داد. در حالت II طبق وضعیت ' $A \rightarrow b \bullet \text{ First}(B)$ ' مشخص شده که در ورودی b ظاهر شده است. اکنون، در صورتی که ترم پیش بینی عنصری متعلق به $\text{First}(B)$ باشد، می توان اذعان داشت که در ورودی A مشاهده شده است. بنابراین میتوان به حالت I_0 برگشت و اعلام کرد که برطبق انتظار A در ورودی مشاهده شد.

اکنون، در حالت I_0 وضعیت ' $S \rightarrow \bullet AB, \$$ ' که در انتظار مشاهده A در ورودی بوده به وضعیت جدید:

$$'S \rightarrow A \bullet B, \$'$$

تبدیل میشود. این وضعیت هسته حالت جدید تری بنام I_2 بصورت زیر میشود.

$I_2: S \rightarrow A \bullet B, \$$
$B \rightarrow \bullet B b, \$, b$
$B \rightarrow \bullet a, \$, b$

در هسته حالت I_2 طبق وضعیت ' $S \rightarrow A \bullet B, \$$ ' انتظار مشاهده B میرود پس از B نیز همانند S انتظار مشاهده $\$$ در این وضعیت میرود. بنابراین باید گسترشهای مختلف B با ترم پیش بینی $\$$ را ایجاد کرد. بنابراین دو وضعیت زیر به این حالت افزوده میشود.

$$B \rightarrow \bullet B b, \$$$

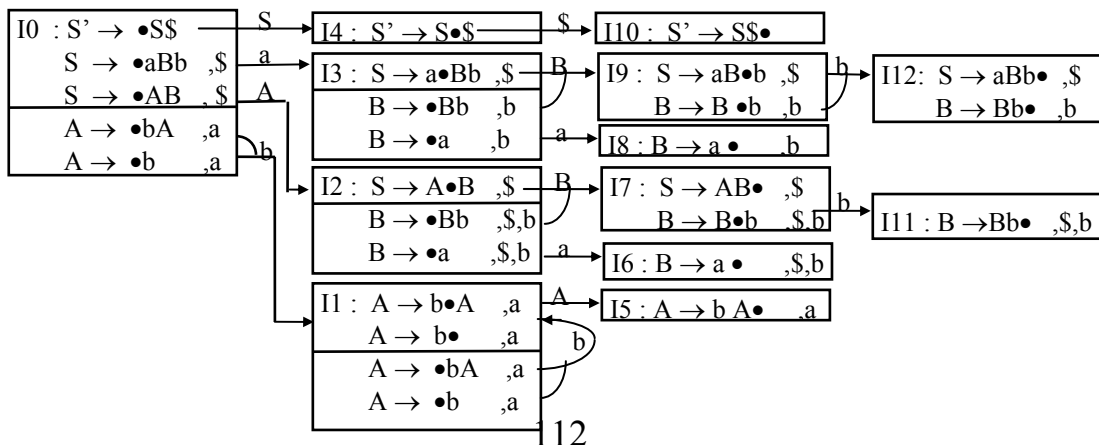
$$B \rightarrow \bullet a, \$$$

حالا، طبق وضعیت ' $B \rightarrow \bullet B b, \$$ ' انتظار مشاهده B در سمت راست میرود و پس از B انتظار دیدن b در ورودی میرود. برای مشاهده B طبق گرامر باید Bb یا a در ورودی ظاهر شود. بنابراین باید دو وضعیت ' $B \rightarrow \bullet B b, b$ ' و ' $B \rightarrow \bullet a, b$ ' به این حالت افزوده شوند. اما، چون قبلاً این دو وضعیت با ترم پیش بینی $\$$ وجود دارند می توان ترم پیش بینی a را فقط به آن افزود. بنابراین دو وضعیت زیر به حالت I_2 افزوده شده است.

$$B \rightarrow \bullet B b, \$, b$$

$$B \rightarrow \bullet a, \$, b$$

اگر به این ترتیب ادامه داده شود، گراف تجزیه LR(1) بصورت زیر ایجاد میشود:





شکل 5.1- گراف تجزیه LR(1)

5.3.2 الگوریتم تولید گراف تجزیه LR(1)

بطور خلاصه برای تولید گراف حالات که يك نمونه از آن در شکل 5.1 نمایش داده شد، باید مراحل زیر انجام شود:

1- عمل تولید گراف از ایجاد حالت شروع I_0 آغاز میشود. چنانچه سر ترم گرامر S باشد، در هسته یا در اسطلاح Kernel این حالت وضعیت

$$I_0: S' \rightarrow \bullet S \$$$

گنجانده میشود. علامت \bullet قبل از S مشخص میکند که در آغاز انتظار مشاهده S میرود.

2- برای هر وضعیت

$$A \rightarrow \alpha \bullet B \beta, \delta$$

از يك حالت که در آن α و یا β هر رشته ای از ترمها و یا اینکه تهی می تواند باشد، باید کلیه گسترشهای ترم میانی B بصورت وضعیتهای:

$$B \rightarrow \bullet \eta, \text{First}(\beta)$$

را به آن حالت افزود. ترم پیش بینی برای این گسترشها $\text{First}(\beta)$ است و چنانچه β وجود نداشته باشد، آنگاه ترم پیش بینی δ است. برای نمونه یکی از وضعیت های حالت I_0 در گراف شکل 5.1 وضعیت $S \rightarrow \bullet AB, \$$ است. چون پس از \bullet ترم میانی A قرار دارد، گسترشهای مختلف A با ترم پیش بینی $\text{First}(B)$ که مساوی a است، در نظر گرفته شده است.

$$A \rightarrow \bullet bA, a$$

$$A \rightarrow \bullet b, a$$

3- ترم پیش بینی برای وضعیت های خودبازگشتی با فرم کلی

$$A \rightarrow \bullet A\alpha, \delta$$

$$A \rightarrow \bullet \beta, \delta$$

بصورت

$$A \rightarrow \bullet A\alpha, \delta, \text{First}(\alpha)$$

$$A \rightarrow \bullet \beta, \delta, \text{First}(\alpha)$$

است. برای نمونه در حالت I_2 از گراف شکل 5.1 برای وضعیت $S \rightarrow A \bullet B, \$$ طبق معمول دوگسترش ترم میانی B با ترم پیش بینی $\$$ ایجاد میشود:

$$B \rightarrow \bullet Bb, \$$$

$$B \rightarrow \bullet a, \$$$

اکنون پس از \bullet ترم میانی B وجود دارد لذا، باید وضعیتهای:

$$B \rightarrow \bullet Bb, b$$

$$B \rightarrow \bullet a, b$$



را به حالت I_2 افزود. بنابراین با ترکیب این دو وضعیت با وضعیتهای بالا دو وضعیت زیر حاصل میشود:

$$\begin{aligned} B &\rightarrow \bullet Bb, b, \$ \\ B &\rightarrow \bullet a, b, \$ \end{aligned}$$

5.3.3 خلاصه عملیات در گراف تجزیه

اگر به گراف شکل 5.1 توجه نمایید. سه نوع عملیات در آن مستتر است.
 1- عمل کاهش یا Reduce: عمل کاهش در بالا توضیح داده شد. برای نمونه در حالت I_{11} طبق وضعیت ' $a, \$$ '، $B \rightarrow B a \bullet$ چنانچه بر سر ورودی یکی از دو ترم پایانی $\$$ و یا a ظاهر شود میتوان بر طبق قاعده ' $B \rightarrow B a$ ' عمل کاهش را انجام داد یا به عبارت دیگر میتوان طبق قاعده شماره 5 عمل R_5 را انجام داد. در حالت کلی عمل R_n به مفهوم کاهش بر اساس قاعده شماره n است.

2- عمل انتقال یا Shift: عمل دیگر که در گراف مستتر است، عمل انتقال است. برای نمونه در حالت I_0 با مشاهده b در ورودی، b به داخل پشته تجزیه انتقال داد میشود و به حالت I_1 گذر میشود. این عمل را بطور خلاصه با S_1 نمایش میدهند. در حالت کلی عمل S_n به مفهوم انتقال و سپس گذر به حالت n است.

3- عمل GoTo: عمل سوم عمل رفتن مستقیم یا در اصطلاح GoTo از یک حالت به حالت دیگر است. برای نمونه در حالت I_0 با تشکیل A در سر پشته گذری به حالت 2 بلافاصله انجام میشود. بطور خلاصه عمل $Goto_2$ انجام میشود. در حالت کلی با تشکیل یک ترم میانی مورد انتظار بر روی پشته میتوان عمل GoTo را به حالت جدیدتر انجام داد.

5.3.4 ایجاد جدول تجزیه

جدول تجزیه در واقع نمایش ماتریسی گراف تجزیه است. عملیات انتقال و GoTo موجب تغییر حالت و یا گذر بین گره های گراف میشود. برای نمونه جدول تجزیه شکل 5.1 را میتوان به جدول زیر تبدیل نمود.

- 0 $S' \rightarrow SS$
- 1 $S \rightarrow aBb$
- 2 $S \rightarrow AB$
- 3 $A \rightarrow bA$
- 4 $A \rightarrow b$

	a	b	\$	S	A	B
0	S3	S1	-	4	2	-
1	-	S1	-	-	5	-
2	S6	-	-	-	-	7
3	S8	-	-	-	-	9
4	-	-	S10	-	-	-
5	R3	-	-	-	-	-
6	-	R6	R6	-	-	-

5 $B \rightarrow Bb$ 6 $B \rightarrow a$

شکل 5.2- جدول تجزیه LR(1)

اکنون با استفاده از جدول فوق می‌توان به راحتی عمل تجزیه بالا به پایین را انجام داد. برای نمونه عبارت $abab$ را توسط جدول فوق بصورت زیر می‌توان مورد تجزیه پایین به بالا قرار داد.

تخته تجزیه	ورودی	دستور العمل
0	<u>a</u> abbb\$	S3
0 a 3	<u>a</u> bbb\$	S8
0 a 3 a 8	<u>b</u> bb\$	R6
0 a 3 B	<u>b</u> bb\$	GoTo 9
0 a 3 B 9	<u>b</u> bb\$	S12
0 a 3 B 8 b 12	<u>b</u> b\$	R5
0 a 3 B	<u>b</u> b\$	GoTo 9
0 a 3 B 9	<u>b</u> b\$	S12
0 a 3 B 8 b 12	<u>b</u> \$	R5
0 a 3 B	<u>b</u> \$	GoTo 9
0 a 3 B 9	<u>b</u> \$	S12
0 a 3 B 8 b 12	\$	R2
0 S	\$	S10

مثال - برای مشخص شدن پررنگی (تذکره) در جدول زیر توجه نمایید :

$$S \rightarrow CC$$

$$C \rightarrow aC$$

$$C \rightarrow d$$

گرامر ساده فوق را در نظر بگیرید ، هدف ایجاد جدول تجزیه LR(1) برای این گرامر است. همانگونه که قبلاً نیز تذکر داده شد ، ابتدا می‌بایست گرامر را به فرم توسعه یافته تبدیل نمود و پس از آن قواعد را شماره گذاری نمود . به این ترتیب این گرامر بصورت زیر تبدیل میشود.

$$0 \quad S' \rightarrow S \$$$

$$1 \quad S \rightarrow CC$$

$$2 \quad C \rightarrow aC$$

$$3 \quad C \rightarrow d$$

اکنون میتوان در مورد حالات ممکن تجزیه شروع به استدلال نمود. به این ترتیب که در حالت شروع وضعیت $S' \rightarrow \cdot S \$$ است . در اینجا علامت \cdot پس از فلش



، چگونگی وضعیت را مشخص میکند . هدف تشخیص s در نهایت مشاهده علامت $\$$ است.

پس باید در حالت شروع ، در ورودی يك s را مشاهده کرد یا انتظار دیدن يك s می رود و پس از آن انتظار می رود که در ورودی $\$$ ظاهر شود. اما طبق گرامر برای مشاهده s باید در ورودی CC را دید، لذا وضعیت در حالت شروع به صورت زیر تبدیل می شود :

$$\begin{aligned} S' &\rightarrow \bullet S \$ \\ S &\rightarrow \bullet CC \end{aligned}$$

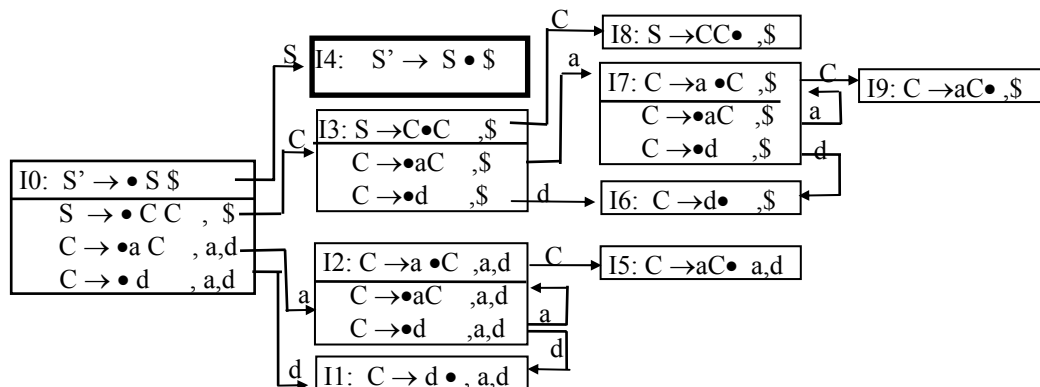
طبق وضعیت $S' \rightarrow \bullet S \$$ ، پس از s انتظار دیدن $\$$ می رود. پس ، پیش بینی میشود که بعد از C نیز در ورودی $\$$ ظاهر شود . حالا در این وضعیت انتظار می رود که در اولین مرحله در ورودی يك C ظاهر شود و بعد از این C اول C دومی ظاهر شود :

$$\begin{aligned} S' &\rightarrow \bullet S \$ \\ S &\rightarrow \bullet CC \quad , \$ \\ C &\rightarrow \bullet aC \quad , \text{First}(C) \\ C &\rightarrow \bullet d \quad , \text{First}(C) \end{aligned}$$

همانگونه که مشاهده می شود طبق وضعیت $S' \rightarrow \bullet S \$$ ، انتظار مشاهده يك C در ورودی می رود. لذا، باید C در ورودی تشخیص داده شود. برای تشخیص C باید طبق گرامر یا aC و یا ترم پایانی d طبق گرامر در ورودی ظاهر شود. در حالت شروع اگر در ورودی d ظاهر شود حالت جدید I_1 تولید می شود. حالت I_1 تنها شامل يك وضعیت :

$$C \rightarrow \bullet d \quad , \text{First}(C)$$

خواهد بود. پس از مشاهده d در ورودی ، میتوان به حالت I_0 برگشت و اعلام نمود که در وضعیت $S' \rightarrow \bullet S \$$ ، انتظار مشاهده يك C میرفت و حالا در ورودی C تشخیص داده شده است. به این ترتیب حالت جدید I_3 با وضعیت $S' \rightarrow C \bullet C \quad , \$$ در هسته آن ظاهر میشود. پس اینجا تلاش برای دیدن C دوم در ورودی و مشاهده $\$$ بعد از C دوم آغاز میشود. بطور کلی گراف تجزیه بر طبق استدلال فوق به صورت زیر ایجاد میشود :





شکل 5.5 - گراف تجزیه LR (1)

جدول تجزیه برای گراف فوق بصورت زیر است.

0	$S' \rightarrow S\$$
1	$S \rightarrow C C$
2	$C \rightarrow a C$
3	$C \rightarrow d$

	Action			GoTo	
	a	d	\$	S	C
0	S2	S1		4	3
1	R3	R3			
2	S2	S1			5
3	S7	S6			8
4	Accept				
5	R2	R2			
6			R3		
7	S7	S6			9
8			R1		
9			R2		

شکل 5.6 - جدول تجزیه LR(1)

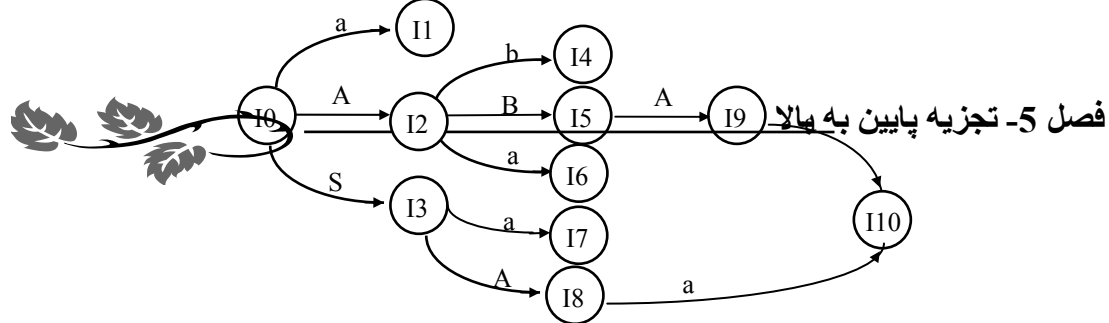
مثال : جدول تجزیه LR(1) برای گرامر زیر ایجاد کنید :

- 0 $S' \rightarrow S\$$
- 1 $S \rightarrow SA$
- 2 $S \rightarrow AB$
- 3 $B \rightarrow BA$
- 4 $B \rightarrow b$
- 5 $A \rightarrow Aa$
- 6 $A \rightarrow a$

گرامر فوق در قرم توسعه یافته است. پس می توان گراف تجزیه را بلافاصله ایجاد کرد. چون تعداد حالات نسبتاً زیاد است ، هر يك از حالات را بصورت جداگانه مشخص نموده سپس ، با استفاده از يك گراف ارتباط بین حالات ترسیم میشود.

- | | | |
|--|--|--|
| I0: $S' \rightarrow \bullet S\$$
$S \rightarrow \bullet SA$, \$, a
$S \rightarrow \bullet AB$, \$, a
$A \rightarrow \bullet Aa$, b, a
$A \rightarrow \bullet a$, b, a | I3: $S' \rightarrow S \bullet \$$
$S \rightarrow S \bullet A$, \$, a
$A \rightarrow \bullet Aa$, \$, a
$A \rightarrow \bullet a$, \$, a | I7: $A \rightarrow a \bullet$, \$, a
I8: $S \rightarrow SA \bullet$, \$, a
$A \rightarrow A \bullet a$, \$, a
I9: $B \rightarrow BA \bullet$, \$, a
$A \rightarrow A \bullet a$, \$, a
I10: $A \rightarrow Aa \bullet$, \$, a |
| I1: $A \rightarrow a \bullet$, b, a
I2: $S \rightarrow A \bullet B$, \$, a
$A \rightarrow A \bullet a$, b, a
$B \rightarrow \bullet BA$, \$, a
$B \rightarrow \bullet b$, \$, a | I4: $B \rightarrow b \bullet$, \$, a
I5: $S \rightarrow AB \bullet$, \$, a
$B \rightarrow B \bullet A$, \$, a
$A \rightarrow \bullet Aa$, \$, a
$A \rightarrow \bullet a$, \$, a
I6: $A \rightarrow Aa \bullet$, b, a | |

گراف تجزیه بصورت زیر است :



جدول تجزیه LR(1) و گرامر در زیر ارائه شده است.

	a	b	\$	S	A	B
0	S1			13	12	
1	R6	R6				
2	S6	S4				5
3	S7		تذیر		8	
4	R4		R4			
5	R2		R2		9	
6	R5		R5			
7	R6		R6			
8	R1,S10		R1			
9	R3,S10		R3			
10	R5		R5			

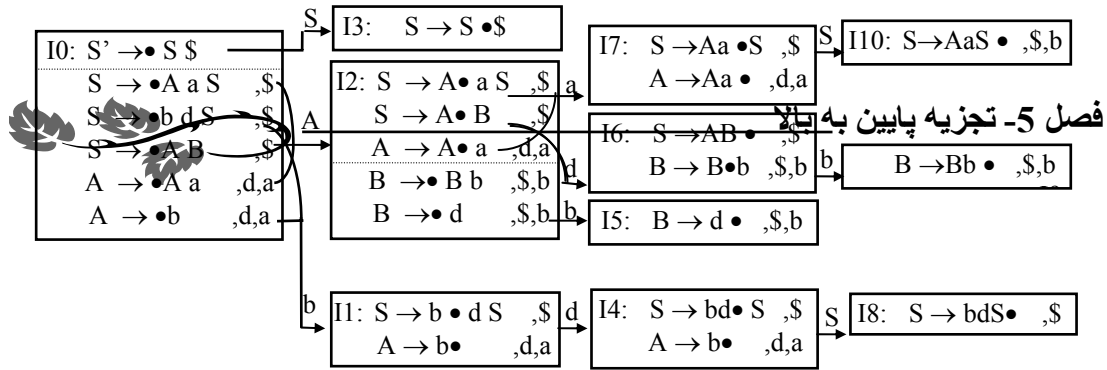
همانگونه که مشاهده می کنید ، در دو حالت I8 و I9 در داخل جدول تجزیه مشخص نیست که با مشاهده a آیا باید عمل کاهش و یا عمل انتقال را انجام داد. در اصطلاح ، در این دو حالت اختلال در کاهش و انتقال² وجود دارد لذا، گرامر LR(1) نیست.

مثال - جدول تجزیه LR(1) برای گرامر زیر ایجاد کنید.

- 0 $S' \rightarrow S \$$
- 1 $S \rightarrow A a S$
- 2 $S \rightarrow b d S$
- 3 $S \rightarrow A B$
- 4 $A \rightarrow A a$
- 5 $A \rightarrow b$
- 6 $B \rightarrow B b$
- 7 $B \rightarrow d$

گراف تجزیه LR(1) برای گرامر فوق بصورت زیر است :

² اختلال در کاهش و انتقال را در اصطلاح Shift Reduce Conflict نیز می گویند. این مشکل موجب میشود که با در دست داشتن يك ترم نیش بینی نتوان تصمیم گرفت عمل Shift باید انجام شود یا Reduce. ممکن است Reduce Reduce Conflict نیز در برخی موارد ایجاد شود.



گرامر فوق LR(1) نیست زیرا ، در حالت I1 با مشاهده d در ورودی مشخص نیست که آیا عمل R5 و یا عمل S4 باید انجام شود. بعبارت دیگر نمی توان مشخص کرد که آیا عمل انتقال باید انجام شود و یا عمل کاهش. بنابراین ، در این حالت اختلال در کاهش و انتقال و یا در اصطلاح Shift Reduce Conflict وجود دارد.

۵.۴ مشکل گرامرهای LR (۱)

اگر توجه نموده باشید در گرامر LR(1) جدول تجزیه نسبت به تعداد قواعد گرامر بسیار بزرگ می باشد. برای نمونه برای گرامر کوچک :

$$\begin{aligned} S &\rightarrow CC \\ C &\rightarrow aC \\ C &\rightarrow d \end{aligned}$$

جدول تجزیه دارای ده ردیف و پنج ستون بود . به این ترتیب جدول تجزیه برای یک گرامر با فقط سه قاعده تعداد پنجاه خانه داشت. برای گرامرهای واقعی با حدود صد قاعده ، جدول تجزیه حداقل ده هزار خانه از حافظه را اشغال می کند. جهت رفع این مشکل سعی شده که گرامر را محدود کنند و گرامرها با امکانات کمتر از گرامرهای LR(1) جهت بیان قواعد استفاده شود و یا اینکه گرامرها به صورت مبهم استفاده شوند. برای مثال گرامر عبارات بصورت :

$$\begin{aligned} E &\rightarrow E+T|E-T|T \\ T &\rightarrow T*F|T/F|F \\ F &\rightarrow Id|No|(E) \end{aligned}$$

را می توان بصورت مبهم و با یک ترم میانی به صورت زیر تعریف کرد :

$$E \rightarrow E+E|E-E|E/E|E*E|(E)|id|No$$

همانطور که مشاهده می کنید در این گرامر بجای سه ترم میانی E و F و T فقط یک ترم میانی استفاده شده و بجای 9 قاعده برای فرم کلی عبارات ، 7 قاعده استفاده شده است که جدول را بسیار کوچکتر مینماید البته ابهام در گرامر مشکلات قابل حلی بوجود خواهد آورد که در مورد آنها کاملاً بحث خواهد شد.

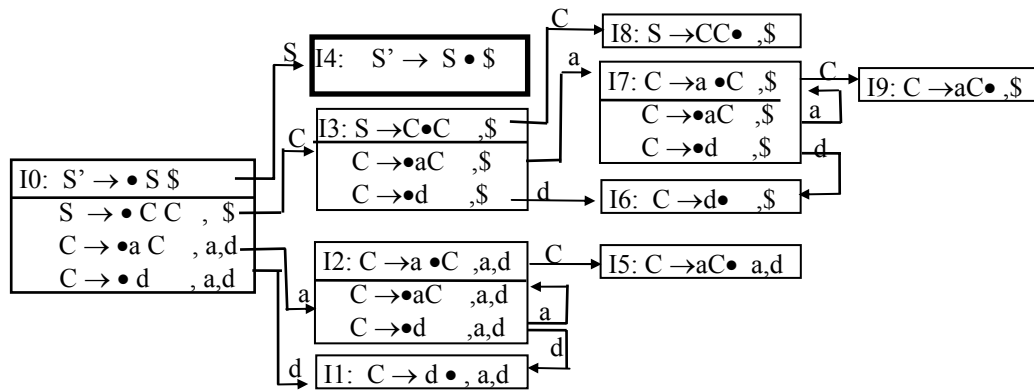


روش دیگر برای کوچکتر نمودن جدول تجزیه پائین به بالا ، استفاده از گرامرهای فرم محدودتری به اسامی SLR و LALR است. این نوع گرامرها در ادامه بحث توضیح داده خواهند شد.

۵.۴ گرامرهای (۱) LALR

در هنگام تولید جدول تجزیه LR (1) چنانچه بتوان بدون هیچگونه مشکلی حالات با وضعیت های مشابه ، اما ترمهای پیش بینی متفاوت را با یکدیگر ادغام نمود ، گرامر را LALR (1) گویند . مشکل به این صورت می تواند باشد که با در دست داشتن یک ترم پیش بینی یا عبارت دیگر یک ورودی بعدی نتوان تصمیم گرفت که چه عملی باید انجام شود. برای نمونه نتوان تصمیم گرفت که آیا عمل کاهش باید انجام شود یا عمل انتقال . برای درک بهتر مسئله به گرامر و گراف تجزیه مربوط به آن در شکل 5.7 توجه کنید.

- 0 $S' \rightarrow S \$$
- 1 $S \rightarrow CC$
- 2 $C \rightarrow aC$
- 3 $C \rightarrow d$



شکل 5.7 - گراف تجزیه LR (1)

در گراف شکل 5.7 دو حالت I2 و I7 دارای وضعیتهای یکسان با ترمهای پیش بینی متفاوت هستند. حالات I1 و I6 و همچنین دو حالت I5 و I9 نیز به همین صورت می باشند. اگر این حالات را با یکدیگر ادغام نمایم ، حاصل بصورت زیر خواهد بود :

الف: حالت I2 با I7 ادغام شده ، حالت جدید I2_7 ایجاد میشود.

$$\begin{array}{l}
 17: C \rightarrow a \bullet C, \$ \\
 C \rightarrow \bullet aC, \$ \\
 C \rightarrow \bullet d, \$
 \end{array}
 +
 \begin{array}{l}
 I2: C \rightarrow a \bullet C, a, d \\
 C \rightarrow \bullet aC, a, d \\
 C \rightarrow \bullet d, a, d
 \end{array}
 \rightarrow
 \begin{array}{l}
 I2_7: C \rightarrow a \bullet C, \$, a, d \\
 C \rightarrow \bullet aC, \$, a, d \\
 C \rightarrow \bullet d, \$, a, d
 \end{array}$$



ب: حالت I1 با I6 ادغام شده ، حالت جدید I1_6 ایجاد میشود.

$$I1: C \rightarrow d \bullet, a, d \quad + \quad I6: C \rightarrow d \bullet, \$ \quad \rightarrow \quad I1_6: C \rightarrow d \bullet, \$, a, d$$

ج: حالت I5 با I9 ادغام شده ، حالت جدید I5_9 ایجاد میشود.

$$I5: C \rightarrow aC \bullet, a, d \quad + \quad I9: C \rightarrow aC \bullet, \$ \quad \rightarrow \quad I5_9: C \rightarrow aC \bullet, \$, a, d$$

اکنون در جدول تجزیه باید بجای حالتهاي ادغام شده ، حالت جدید ادغامي را قرار داد. برای نمونه در بالا حالتهاي I5 و I9 از جدول تجزیه LR(1) حذف و با I5_9 جایگزین می شوند. به این ترتیب ، هر دستورالعمل انتقال S5 با S9 با دستورالعمل S5_9 جایگزین میشود. هر عمل GoTo به این حالتها نیز به GoTo به I5_9 تبدیل می شود. جدول تجزیه قبل از جایگزینی و پس از جایگزینی حالتهاي ترکیب شونده در زیر مشخص شده است:

	Action			GoTo	
	a	d	\$	S	C
0	S2	S1		4	3
1	R3	R3			
2	S2	S1			5
3	S7	S6			8
4	Accept				
5	R2	R2			
6			R3		
7	S7	S6			9
8			R1		
9			R2		

	Action			GoTo	
	a	d	\$	S	C
0	S2_7	S1_6	-	4	3
1_6	R3	R3	R3	-	-
2_7	S2_7	S1_6	-	-	5_9
3	S2_7	S1_6	-	-	8
4	Accept				
5_9	R2	R2	R2		
8			R1		

ب- قبل از ادغام

الف- پس از ادغام

شکل 5.8- جدول تجزیه LR(1) و LALR(1)

قبل از ادغام ، باید حالتهاي کاندید را در داخل جدول مشخص کرد. هر جا که دستورالعمل GoTo به یکی از حالتهاي ادغامي وجود دارد ، نام حالت ادغامي به جای آن قرار می گیرد. همین عمل را برای دستورالعمل هاي انتقال یا در اصطلاح Shift نیز باید انجام داد.

مثال - جدول تجزیه LALR(1) برای گرامر زیر ایجاد کنید:

- 0 E' → E \$
- 1 E → E+ T
- 2 E → T
- 3 T → T * F
- 4 T → F
- 5 F → (E)
- 6 F → id



حالات گراف تجزیه LR(1) برای این گرامر در زیر مشخص شده است:

I0: $E' \rightarrow \bullet E \$$ $E \rightarrow \bullet E + T \quad , \$, +$ $E \rightarrow \bullet T \quad , \$, +$ $T \rightarrow \bullet T * F \quad , \$, +, *$ $T \rightarrow \bullet F \quad , \$, +, *$ $F \rightarrow \bullet (E) \quad , \$, +, *$ $F \rightarrow \bullet Id \quad , \$, +, *$	I1: $F \rightarrow Id \bullet \quad , \$, +, *$	I2: $F \rightarrow (\bullet E) \quad , \$, +, *$ $E \rightarrow \bullet E + T \quad ,) , +$ $E \rightarrow \bullet T \quad ,) , +$ $T \rightarrow \bullet T * F \quad ,) , +, *$ $T \rightarrow \bullet F \quad ,) , +, *$ $F \rightarrow \bullet (E) \quad , } , +, *$ $F \rightarrow \bullet Id \quad , } , +, *$
I3: $T \rightarrow F \bullet \quad , \$, +, *$	I4: $E \rightarrow T \bullet \quad , \$, +$ $T \rightarrow T \bullet * F \quad , \$, +, *$	I5: $E' \rightarrow E \bullet \$$ $E \rightarrow E \bullet + T \quad , \$, +$
I6: $F \rightarrow Id \bullet \quad ,) , +, *$	I7: $F \rightarrow (\bullet E) \quad ,) , +, *$ $E \rightarrow \bullet E + T \quad ,) , +$ $E \rightarrow \bullet T \quad ,) , +$ $T \rightarrow \bullet T * F \quad ,) , +, *$ $T \rightarrow \bullet F \quad ,) , +, *$ $F \rightarrow \bullet (E) \quad , } , +, *$ $F \rightarrow \bullet Id \quad , } , +, *$	I8: $T \rightarrow F \bullet \quad ,) , +, *$ I9: $E \rightarrow T \bullet \quad ,) , +$ $T \rightarrow T \bullet * F \quad ,) , +, *$ I10: $F \rightarrow (E \bullet) \quad , \$, +, *$ $E \rightarrow E \bullet + T \quad ,) , +$
I11: $T \rightarrow T * \bullet F \quad , \$, +, *$ $F \rightarrow \bullet (E) \quad , \$, +, *$ $F \rightarrow \bullet Id \quad , \$, +, *$	I12: $E \rightarrow E + \bullet T \quad , \$, +$ $T \rightarrow \bullet T * F \quad , \$, +, *$ $T \rightarrow \bullet F \quad , \$, +, *$ $F \rightarrow \bullet (E) \quad , \$, +, *$ $F \rightarrow \bullet Id \quad , \$, +, *$	I13: $F \rightarrow (E \bullet) \quad ,) , +, *$ $E \rightarrow E \bullet + T \quad ,) , +$
I14: $T \rightarrow T * \bullet F \quad ,) , +, *$ $F \rightarrow \bullet (E) \quad , } , +, *$ $F \rightarrow \bullet Id \quad , } , +, *$	I15: $E \rightarrow E + \bullet T \quad ,) , +$ $T \rightarrow \bullet T * F \quad ,) , +, *$ $T \rightarrow \bullet F \quad ,) , +, *$ $F \rightarrow \bullet (E) \quad , } , +, *$ $F \rightarrow \bullet Id \quad , } , +, *$	I16: $F \rightarrow (E) \bullet \quad , \$, +, *$ I17: $T \rightarrow T * F \bullet \quad , \$, +, *$ I18: $E \rightarrow E + T \bullet \quad , \$, +$ $T \rightarrow T \bullet * F \quad , \$, +, *$
I19: $F \rightarrow (E) \bullet \quad ,) , +, *$	I20: $T \rightarrow T * F \bullet \quad ,) , +, *$	I21: $E \rightarrow E + T \bullet \quad ,) , +$ $T \rightarrow T \bullet * F \quad ,) , +, *$

در زیر جدول تجزیه LR(1) و LALR(1) برای این گرامر ارائه شده است.

	Action						Goto		
	id	()	+	*	\$	T	E	
0	S1	S2					3	4	5
1			R6	R6	R6	R6			
2	S1	S2					3	4	10
3			R4	R4	R4	R4			
4			R2	R2	S11	R2			
5				S12		acc			
10			S16	S12					
11	S1	S2					17		
12	S1	S2					3	18	
16			R5	R5	R5	R5			
17			R3	R3	R3	R3			
18			R1	R1	S11	R1			

	Action						Goto			
	id	()	+	*	\$	F	T	E	
0	S1	S2					3	4	5	
1				R6	R6	R6				
2	S6	S7					8	9	10	
3				R4	R4	R4				
4				R2	S11	R2				
5				S12		acc				
6				R6	R6	R6				
7	S6	S7					8	9	13	
8				R4	R4	R4				
9				R2	R2	S14				



10			S16	S15					
11	S1	S2					17		
12	S1	S2					3	18	
13			S19	S15					
14	S6	S7					20		
15	S6	S7					8	21	
16				R5	R5	R5			
17				R3	R3	R3			
18				R1	S11	R1			
19				R5	R5	R5			
20				R3	R3	R3			
21				R1	R1	S14			

شکل 5.9- جداول تجزیه LR(1) و LALR(1)

برای ادغام حالات ، ابتدا حالاتی را که بدون ایجاد مسأله اختلال در کاهش و انتقال و به سادگی قابل ادغام هستند باید در نظر گرفت. حالات I1 و I6 ، I3 و I8 ، I16 و I19 و بلاخره I17 و I20 بلافاصله قابل ادغام هستند. با جایگزینی حالات معادل مشاهده میشود که حالات I11 و I14 نیز قابل ادغام هستند. با ادغام این دو حالت می توان نتیجه گرفت که دو حالت I18 و I21 نیز یکسان هستند.

5.5 گرامرهای SLR(1)

گرامرهای Simple LR(1) یا بطور مختصر SLR(1) روشی دیگر برای تجزیه پائین به بالا را ارائه می کنند. این گرامرها بسیار محدودتر از گرامرهای LALR(1) و بالنتیجه گرامرهای LR(1) هستند زیرا، در این نوع گرامرها فرض بر این است که ترمهای پیش بینی برای یک ترم میانی A وابسته به محل قرار گرفتن A در سمت راست قواعد نمی باشد. بلکه برای هر ترم میانی A ، مستقل از چگونگی قرار گرفتن آن در سمت راست قواعد مختلف، همواره مجموعه Follow(A) شاخص ترمهای پیش بینی برای A است. بنابراین تعداد حالات و یا ردیف های جدول تجزیه مساوی با جدول LALR(1) است زیرا ، در اینجا همانند LALR(1) ترمهای پیش بینی عاملی برای تفکیک دو حالت از یکدیگر نیستند. به عبارت دیگر نمی توان دو حالت مشخص نمود که دارای وضعیت های یکسان اما ترم های پیش بینی متفاوت باشند.

جهت درک بهتر نقش ترمهای پیش بینی در گرامرها به مثال زیر توجه کنید :

Statement \rightarrow IfSt | WhileSt

IfSt \rightarrow If condition Then Statement ElsePart



WhileSt \rightarrow While condition Do Statement

با دقت در این مثال می بینیم که پس از مشاهده if درون IfSt. ترم میانی Condition. قرار گرفته است. درون While St. نیز پس از کلمه While ترم میانی condition ظاهر شده است اما، در قالب جمله if بعد از Condition انتظار دیدن Then و در قالب جمله While پس از Condition انتظار مشاهده DO می رود. بنابراین مشاهده می شود که وابسته به اینکه یک ترم میانی سمت راست کدام قاعده قرار گرفته ترمهای پیش بینی آن متفاوت است. اگر ترم میانی Condition در وضعیت زیر باشد :

IfSt \rightarrow If • condition Then Statement ElsePart

آنگاه ترم پیش بینی برای گسترشهای متفاوت condition کلمه Then خواهد بود. یعنی باید گسترشهای متفاوت condition با ترم پیش بینی Then را پس از وضعیت فوق به حالت مربوطه افزود. در صورتیکه ، اگر Condition در قالب جمله While و در وضعیت زیر قرار گیرد :

WhileSt \rightarrow While • condition Do Statement

آنگاه ترم پیش بینی برای گسترشهای متفاوت condition کلمه Do خواهد بود. یعنی باید گسترشهای متفاوت condition با ترم پیش بینی Do را پس از وضعیت فوق به حالت مربوطه افزود.

در گرامرهای (1) SLR گرامر آنقدر محدود است که مهم نیست ترم میانی در داخل چه قاعده ای قرار داشته باشد. در این نوع از گرامرها همواره برای یک ترم میانی ترمهای پیش بینی مساوی با مجموعه Follow آن ترم میانی است. در مثال فوق اگر بجای Then در تعریف IfSt کلمه Do بصورت زیر قرار داده میشود

Statement \rightarrow IfSt | WhileSt

IfSt \rightarrow If condition Do Statement ElsePart

WhileSt \rightarrow While condition Do Statement

آنگاه گرامر (1) SLR می بود. برای نمونه در زیر جدول تجزیه (1) SLR برای گرامر عبارات ایجاد شده است. در شکل 5.9 جدول های تجزیه LR(1) و LALR(1) برای عبارات مشخص شد. در اینجا گرامر عبارات بصورت زیر مطرح است :

- 0 $E' \rightarrow E \$$
- 1 $E \rightarrow E + T$
- 2 $E \rightarrow T$
- 3 $T \rightarrow T * F$
- 4 $T \rightarrow F$
- 5 $F \rightarrow (E)$
- 6 $F \rightarrow id$



در مورد گرامرهای SLR(1) قبل از ایجاد گراف تجزیه باید اقدام به تولید مجموعه پیرو³ یا در اصطلاح Follow برای ترمهای میانی نمود تا در واقع ترمهای پیش بینی برای هر ترم میانی مشخص شود .
مجموعه پیرو برای ترمها در گرامر فوق بصورت زیر است.

$$\begin{aligned} \text{Follow}(E) &= \{ \$, +,) \} \\ \text{Follow}(T) &= \{ * \} + \text{Follow}(E) = \{ \$, +,), * \} \\ \text{Follow}(F) &= \text{Follow}(T) \end{aligned}$$

اکنون با در دست داشتن مجموعه پیرو برای ترمهای میانی بسادگی می توان گراف تجزیه SLR(1) را تولید نمود.

I0: E' → • E \$ E → • E+ T E → • T T → • T * F T → • F E → • (E)	I1: F → Id •	I2: F → (• E) E → • E+ T E → • T T → • T * F T → • F E → • (E)	I3: T → F • I4: E → T • T → T • * F I5: E' → E • \$ E → E • + T
I6: F → (E) • E → E • + T I7: T → T • * F F → • (E) F → • Id	I8: E → E+ • T T → • T * F T → • F F → • (E) F → • Id	I9: F → (E) • I10: T → T * F • I11: E → E+ T • T → T • * F	

در زیر جدول تجزیه SLR(1) برای گرامر فوق ارائه شده است . باید توجه داشته باشید که برای این دسته از گرامرها مجموعه های پیرو ترم های پیش بینی را تشکیل میدهند. لذا، در داخل جدول تجزیه برای کاهش id به F بر طبق قاعده شماره 6 در زیر ستونهای مربوط به عناصر مجموعه پیرو F دستور العمل R6 قرار داده شده است.

	Action					Goto		
	id	()	+	*	\$	F	T	E
0	S1	S2				3	4	5
1			R6	R6	R6			
2	S1	S2				3	4	6
3			R4	R4	R4			
4			R2	R2	S7	R2		
5				S8	acc			
6			S9	S8				
7	S1	S2				10		
8	S1	S2				3	11	
9			R5	R5	R5			
10			R3	R3	R3			
11			R1	R1	S7	R1		

³ همانگونه که در فصل قبل نیز توضیح داده شد، برای هر ترم میانی مجموعه پیرو برای یک ترم میانی به مجموعه ترم هایی اطلاق می شود که می تواند بعد از آن ترم میانی ظاهر شود . بنابراین ، برای بدست آوردن مجموعه پیرو برای یک ترم میانی باید به قوانین نگریست و مجموعه First برای ترمهای بعدی آن را بدست آورد . در حالت کلی :

- الف- از قاعده $A \rightarrow \alpha B \beta$ می توان نتیجه گرفت $\text{Follow}(B) \supset \{b\}$
- ب - از قاعده $A \rightarrow \alpha B \beta$ می توان نتیجه گرفت $\text{Follow}(B) \supset \text{First}(\beta)$
- ج - از قاعده $A \rightarrow \alpha B$ می توان نتیجه گرفت $\text{Follow}(B) \supset \text{First}(A)$



شکل 5.9- جداول تجزیه (SLR(1) برای گرامر ساده عبارات

۵.۶ گرامرهای مبهم

همانگونه که قبلاً توضیح داده شد، گرامر مبهم گرامری است که بر اساس آن بیش از یک درخت تجزیه برای جمله داده شده و در نتیجه دو مفهوم متناقض بتوان داشت. با استفاده از گرامرهای مبهم می توان اندازه جدول تجزیه پائین به بالا را کوچکتر نمود. برای نمونه گرامر ساده عبارات را که در بخشهای قبل جدول تجزیه LR(1)، LALR(1) و SLR(1) برای آن ایجاد شد را بخاطر بیایورید. جدول تجزیه SLR(1) و LALR(1) برای این گرامر دارای یازده حالت و در مجموع 108 خانه از حافظه را اشغال می کرد. با نوشتن گرامر عبارات به صورت مبهم زیر:

$$E \rightarrow E + E \mid E - E \mid E * E \mid E / E \mid (E) \mid \text{NO} \mid \text{id}$$

تعداد ترم های میانی کاهش می یابد و نشان داده خواهد شد که . اما، مبهم بودن گرامر و نوشتن آن به صورت فوق دو مشکل ایجاد می کند . این مشکلات توضیح داده خواهند شد.

الف- در گرامر مبهم فوق الویت عملگرها یکسان است.

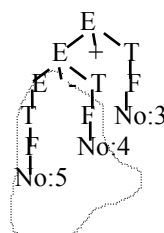
ب- اجتماع عملگرهای جمع و ضرب مشخص نیست.

قواعد مربوط به گرامر عبارات در حالت غیر مبهم بیانگر الویت و اجتماع عملگرها است.

- 0 $E' \rightarrow E \$$
- 1 $E \rightarrow E + T$
- 2 $E \rightarrow E - T$
- 3 $E \rightarrow T$
- 4 $T \rightarrow T * F$
- 5 $T \rightarrow T / F$
- 6 $T \rightarrow F$
- 7 $F \rightarrow (E)$
- 8 $F \rightarrow \text{id}$
- 9 $F \rightarrow \text{No}$

در گرامر فوق + و - دارای اجتماع چپ هستند. برای نمونه درخت تجزیه برای عبارت زیر را در نظر بگیرید :

5 - 4 + 3





همانگونه که مشاهده میکنید ابتدا باید عبارت 4-5 را محاسبه نمود تا بتوان پس از آن حاصل را با 3 جمع کرد. در واقع از چپ به راست عمل جمع و تفریق انجام میشود. این بخاطر بیان قواعد مربوط به E بصورت خودبازگشتی چپ است. اگر قواعد مربوط به E بصورت خودبازگشتی راست بیان میشد. یعنی بصورت :

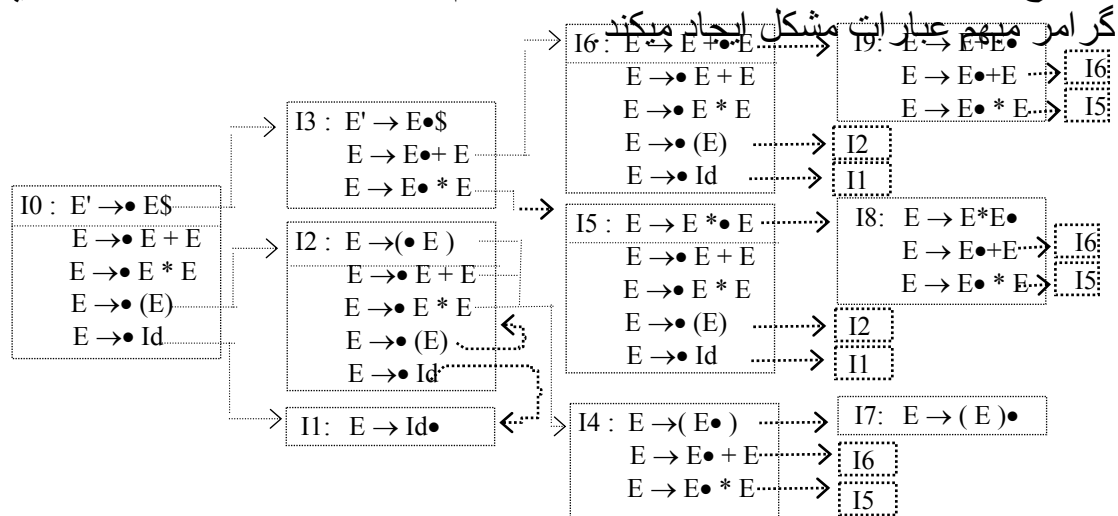
$$E \rightarrow T+E \mid T-E \mid T$$

آنگاه عبارت 3+4-5 از راست به چپ محاسبه می شد. یعنی ، ابتدا 3+4 و سپس 5-7 محاسبه میشد. بر طبق گرامر فوق قبل از اینکه بتوان E را انجام داد باید T را محاسبه کرد. در ون T عملیات ضرب و تقسیم و درون E عملیات تفریق و جمع نهفته است. لذا، ضرب و تقسیم بر طبق این گرامر بر جمع و تفریق الویت دارند.

بنابر این با بیان گرامر خلاصه عبارات بصورت مبهم :

- 0 $E' \rightarrow E\$$
- 1 $E \rightarrow E + E$
- 2 $E \rightarrow E * E$
- 3 $E \rightarrow (E)$
- 4 $E \rightarrow Id$

اولاً الویبهها یکسان و ثانیاً اجتماع عملگرها نامشخص خواهد بود. نامشخص بودن اجتماع و یکسان بودن الویت عملگرها در هنگام ایجاد جدول تجزیه SLR(1) برای



جدول تجزیه برای گراف فوق در شکل 9.10 دارای دو نوع عملکرد برای عملگرهای + و * در حالات 8 و 9 است. در این دو حالت به اختلال در کاهش و انتقال وجود دارد و مشخص نیست که با مشاهده دو عملگر + و * آیا عمل کاهش را باید انجام داد یا انتقال. باید توجه داشته باشید که در گرامرهای SLR(1) ترمهای پیش بینی را مجموعه های پیرو مشخص میکنند. در گرامر فوق برای ترم میانی E باید مجموعه پیرو محاسبه شود.

$$\text{Follow}(E) = \{ \$, +, *,) \}$$



	Action						Goto
	id	()	+	*	\$	E
0	S1	S2					3
1			R4	R4	R4	R4	
2	S1	S2					5
3				S6	S5	Accept	
4			S7	S6	S5		
5	S1	S2					8
6	S1	S2					9
7			R3	R3	R3	R3	
8			R2	S6, R2	S5, R2		
9			R1	S6, R1	S5, R1		

شکل 5.10- جداول تجزیه SLR(1) برای گرامر مبهم عبارات

در حالت 8 با توجه به وضعیت “ $E \rightarrow E * E$ ” میتوان گفت که قبلاً يك علامت * در ورودی دیده شده است. طبق وضعیت “ $E \rightarrow E \bullet + E$ ” انتظار مشاهده + در سر ورودی می‌رود. لذا، با مشاهده يك عملگر + در ورودی چون ضرب بر جمع الویت دارد عمل کاهش R2 انجام میشود. برای نمونه عبارت $4 + 3 * 2$ را در نظر بگیرید. مسلماً در اینجا قبل از انتقال علامت + به داخل پشته تجزیه ، باید عمل ضرب انجام شود لذا، بجای S6 عمل R2 انجام میشود .

در حالت 9 با توجه به وضعیت “ $E \rightarrow E + E$ ” میتوان گفت که قبلاً يك علامت + در ورودی دیده شده است. طبق وضعیت “ $E \rightarrow E \bullet * E$ ” انتظار مشاهده * در سر ورودی می‌رود. لذا، با مشاهده يك عملگر * در ورودی چون ضرب بر جمع الویت دارد عمل انتقال S5 انجام میشود. برای نمونه عبارت $4 * 3 + 2$ را در نظر بگیرید. مسلماً در اینجا قبل از انجام عمل جمع باید عمل ضرب انجام شود لذا، علامت * به داخل پشته تجزیه انتقال داده شده ، بجای R1 عمل S5 انجام میشود .

در حالت 8 با توجه به وضعیت “ $E \rightarrow E * E$ ” میتوان گفت که قبلاً يك علامت * در ورودی دیده شده است. طبق وضعیت “ $E \rightarrow E \bullet * E$ ” انتظار مشاهده * دیگری در سر ورودی می‌رود. لذا، با مشاهده يك عملگر * در سر ورودی چون ضرب دارای اجتماع چپ است ، عمل کاهش R2 انجام میشود. برای نمونه عبارت $4 * 3 * 2$ را در نظر بگیرید. مسلماً در اینجا قبل از انتقال علامت * به داخل پشته تجزیه ، باید عمل ضرب انجام شود لذا، بجای S5 عمل R2 انجام میشود .

در حالت 9 با توجه به وضعیت “ $E \rightarrow E + E$ ” میتوان گفت که قبلاً يك علامت + در ورودی دیده شده است. طبق وضعیت “ $E \rightarrow E \bullet + E$ ” انتظار مشاهده + دیگری در سر ورودی می‌رود. لذا، با مشاهده يك عملگر + در سر ورودی چون جمع دارای اجتماع چپ است ، عمل کاهش R1 انجام میشود. برای نمونه عبارت $4 + 3 + 2$ را در نظر بگیرید. مسلماً در اینجا قبل از انتقال علامت + به داخل پشته تجزیه ، باید عمل جمع انجام شود لذا، بجای S6 عمل R1 انجام میشود .

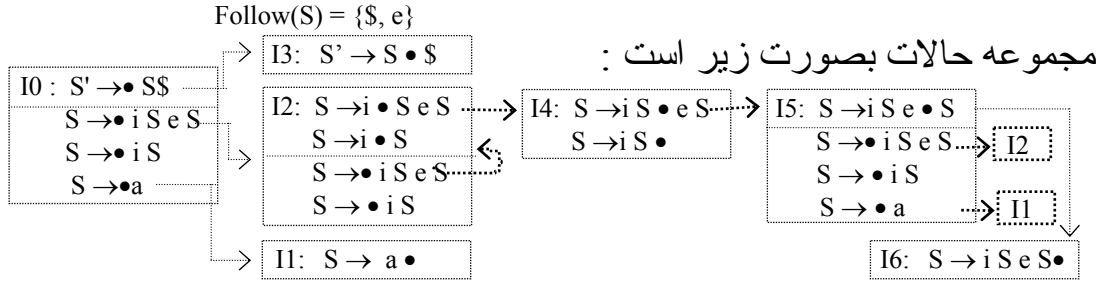
مثال - برای گرامر مبهم جملات شرطی IF جدول تجزیه SLR(1) ایجاد کنید.

$$0 \quad S' \rightarrow S \$$$



- 1 $S \rightarrow iSeS$
- 2 $S \rightarrow iS$
- 3 $S \rightarrow a$

در گرامر فوق s نمایانگر جمله ، i مخفف if و e مخفف expression است. در اولین مرحله باید مجموعه پیرو را برای ترم های میانی و سرترم گرامر بدست آورد. این مجموعه در واقع ترم های پیش بینی را برای ترم میانی مربوطه مشخص می کند.



در ردیف 4 از جدول تجزیه چون else یا e متعلق به نزدیکترین جمله if است ، $S5$ انتخاب شده است.

	Action				Goto
	i	e	a	\$	S
0	S2		S1		3
1		R3		R3	
2	S2		S1		4
3				Accept	
4		S5, R2		R2	
5	S2		S1		6
6		R1		R1	

شکل 5.11- جدول تجزیه SLR(1) برای گرامر مبهم جملات شرطی If



5.7 تمرین

تمرین 1 - نشان دهید که گرامر زیر $LALR(1)$ است :

$$S \rightarrow A a | b a c | d c | b d a$$

$$A \rightarrow d$$

تمرین 2 - آیا گرامر زیر $LR(1)$ است.

$$S \rightarrow SAB | a B$$

$$A \rightarrow a A | A d$$

$$B \rightarrow B a | b A d$$

تمرین 3 - آیا گرامر زیر $LALR(1)$ است.

$$S \rightarrow a A B | S D b$$

$$A \rightarrow a D B | A b$$

$$B \rightarrow B D a | a b D$$

$$D \rightarrow D a | d b$$

تمرین 4 - چرا جدول تجزیه $LALR(1)$ و $SLR(1)$ برای یک گرامر به یک اندازه هستند. برای نمونه جدول تجزیه $LALR(1)$ و $SLR(1)$ برای عبارات به یک اندازه است.

تمرین 5 - اگر گرامری $LR(1)$ باشد و حالات مشابه آن با یکدیگر ادغام شوند تا جدول تجزیه $LALR(1)$ حاصل شود در هنگام ادغام حالات تنها امکان مشکل اختلال در کاهش و انتقال و نه اختلال در کاهش و کاهش و یا در اصطلاح Reduce-Reduce Conflict بوجود خواهد آمد. منظور از اختلال در کاهش و کاهش این است که برای نمونه با مشاهده a در ورودی بتوان دو عمل متفاوت R_2 و R_3 را انجام داد.

تمرین 6 - نشان دهید که هر گرامر $LL(1)$ یک گرامر $LR(1)$ است، اما بالعکس صادق نیست.

تمرین 7 - با در نظر گرفتن اینکه عملگر توان الویت بیشتری نسبت به ضرب و تقسیم دارد و اجتماع آن راست میباشد، گرامر عبارات را تکمیل نموده، جدول تجزیه $SLR(1)$ برای این گرامر ایجاد کنید.

تمرین 8 - در گرامر زیر علامت \wedge نمایانگر توان است. جدول تجزیه $SLR(1)$ ایجاد کنید.

$$E \rightarrow E + E | E - E | E * E | E / E | E * E | (E) | NO | id$$

تمرین 9 - جدول تجزیه $SLR(1)$ برای گرامر زیر تشکیل دهید.

$$E \rightarrow E \theta_1 E | E \theta_2 E | \dots | E \theta_n E | (E) | id$$



فرض کنید که عملگرهای θ_i دارای اجتماع راست و θ_i الویت بیشتری نسبت به θ_j دارد اگر اندیس i بزرگتر از j باشد.

تمرین 10- نشان دهید که گرامر زیر LL(1) است و SLR(1) نیست.

$$S \rightarrow A a A b \mid B b B a$$

$$A \rightarrow \varepsilon$$

$$B \rightarrow \varepsilon$$

آیا گرامر فوق LR(1) است.

راهنمایی - توجه داشته باشید در هر حالتی که گسترش تهی برای ترمهای میانی A یا B وجود داشته باشد، می توان در داخل جدول تجزیه در زیر ستون مربوط ترمهای پیش بینی برای این دو ترم عمل کاهش قرار داد.