



بنام خدا

تجزیه Bottom up

(مثال) $E \rightarrow E + T \mid T$
 $T \rightarrow T * F \mid F$
 $F \rightarrow (E) \mid id$

درودی $id * id$

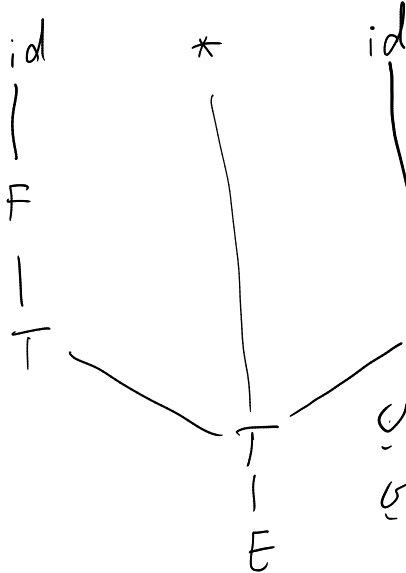
$E \rightarrow T \rightarrow T * F \rightarrow T * id \rightarrow F * id \rightarrow id * id$ ✓

Left-to-right

تجزیه

right most

استنتاج



مراحل تجزیه به صورت BU برعکس مراحل استنتاج $right\ most$ می باشد

چون درخت از برگ به ریشه ساخته می شود و از آنجایی که همچنان درودی از صلب به است خوانده می شود اولین ترسیمی که جایگزین می شود است چه درودی قرار داد که در مراحل استنتاج آخرین سبیل می هستند که تولید شده اند.

ایده تجزیه BU

یک - BU راست ترین استنتاج را به صورت عکس دنبال می کند

آخر - $\alpha \beta \omega$ یک مرحله از پارس BU باشد ($\alpha \beta \omega$ یک رشته (حرم جمله ای) در استنتاج است)

فرض کنید مرحله بعدی پارس $\beta \rightarrow X$ انجام می شود یعنی β با X جایگزین می شود

 $X \in N$

$S \xrightarrow{*} \alpha X \omega \rightarrow \alpha \beta \omega$

 $X \rightarrow \beta$

ω رشته ای از ترمینال است

$\alpha \underbrace{\beta \omega}_{\gamma \in T}$

 $Y \rightarrow \alpha \beta \omega$

$\alpha \beta \omega \rightarrow \alpha X \omega$ مرحله ای در راست ترین استنتاج است یعنی وقتی

X را جایگزین می کنیم همه سبیل می ω باید ترمینال باشند که نسبت استنتاج به X رسیده باشد یعنی در تجزیه در این مرحله می خواهیم β را با X جایگزین کنیم

سوال چرا nAy را به عنوان β انتخاب کردیم و $nAya$ را انتخاب نکردیم؟
 (بفرض این production می باشد $\rightarrow nAya$ لازم داریم)

$\alpha \quad \beta \quad \omega$

اوشی برای تقسیم رسته فعلی به دو زیر رسته نیاز داریم

سوال اگر در تجربه رشته ای مثل $\frac{ET}{\sqrt{2}}$ $\leq \frac{\{NOT\}}{2\beta}$ رسیده باشیم می توانیم در مرحله بعد n را هم از دور دی خوانیم یا نباید بخوانیم. پس خط اول را همین رشته همین قسم است.

در تحریر BU دو محل مساوی = انجام می شود

۱- shift یک توکن دیگر از درودی کجوالتم (جدا ساز به سمت راست حرکت کند)
انتقال

۲. **reduce** در رشته فعلی (در خرم می جمله ای اشتقاق) یک زیر رشته را که درست است، است یعنی از production می بینیم
 کاهش
 با غیر ترسیال است چه production جایگزین کنیم

2) $E \rightarrow T + E \mid T$
 $T \rightarrow id * T \mid id$

$$G_{3,9} \quad id * id \quad + id$$

① $id * id + id$ سنت

② $\text{id} \mid * \text{id} \neq \text{id}$ S

id * (id + id) ^{سنت}
T (* id + id) ^{reduco}

۴ ارادہ دہم بن بست فی ضمیمہ

③ $\text{id} \neq \text{id} + \text{id}$ \checkmark

⑧ $T + id$ | ✓

④ $\text{id} \times \text{id} \mid + \text{id}$ ✓

⑨ $T+T$ | r

⑤ $\text{id}_* T| + \text{id}$ ✓

① $T+E$ ✓

⑥ T | tid S

⑪ E ✓

⑦ $T_f \mid id \quad S$



به هم ضربه

عملیات جداسازی ورودی به دو زیررشته را می توان از طریق استک پیاده سازی کرد
پسین مرتب که در انتهای استک خالی است و یک رشته ورودی داریم. حوازمی توان از زیررشته ورودی با عمل shift یک بزرگ
به استک اضافه کرد یا زیررشته ای از روی استک برداشت و کاهش داد

حال در سوال پیش می آید
در هر مرحله چه احساس تقسیم کنیم که shift یا reduce انجام دهیم Conflict shift-reduce
اگر امکان کاهش از طریق پسین از یک production متفاوت وجود داشته باشد
با کدام یک reduce را انجام دهیم Conflict reduce-reduce

هنگل handle

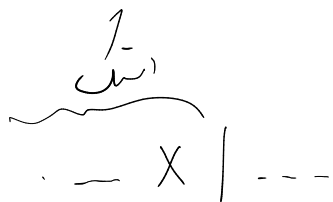
$$S \xRightarrow{*} \alpha X \omega \rightarrow \alpha \beta \omega$$

$$X \rightarrow \beta$$

در عملیات تجزیه اگر به نرم جمله ای $\alpha \beta \omega$ رسیدیم و
ا ω از روی خواننده خوانده نشده بتوانیم β را به X کاهش دهیم
در این صورت β را هنگل $\omega \alpha$ گوئیم

هنگل، زیررشته از قرینال دو غیر قرینال است که اگر آن را کاهش دهیم، کاهش دی بعد از آن می توانیم بنایا به
S ختم شوند.

ما می خواهیم فقط روی هنگل کاهش دهیم



در تجزیه $\omega \beta$ با استفاده از عمل shift/reduce، هنگل β فقط روی استک
ظاهر می شوند و نه درون آن.

- با فاصله بعد از کاهش یک هنگل، یک غیر انتهای روی استک قرار می گیرد (X)
X در اشتقاق، راست ترین غیر انتهای بوده است. هنگل بعدی باید سمت راست X باشد
زیر اشتقاق یک اشتقاق right most بوده است
پس دنباله ای از حرکات شیفت مارا به هنگل بعدی می رساند.

مثال $T \rightarrow (E)$
ا و (E) می تواند روی استک ظاهر شوند. (E) می تواند ظاهر شود
چون (E) هنگل است که باید قبلاً کاهش داده شود و اگر از X گذر کنیم دیگر هیچ وقت نمی توانیم (E) را بارس کنیم



باز به تعریف یک مفهوم جدید برای تشخیص کردن محتوای ممکن درون استک داریم

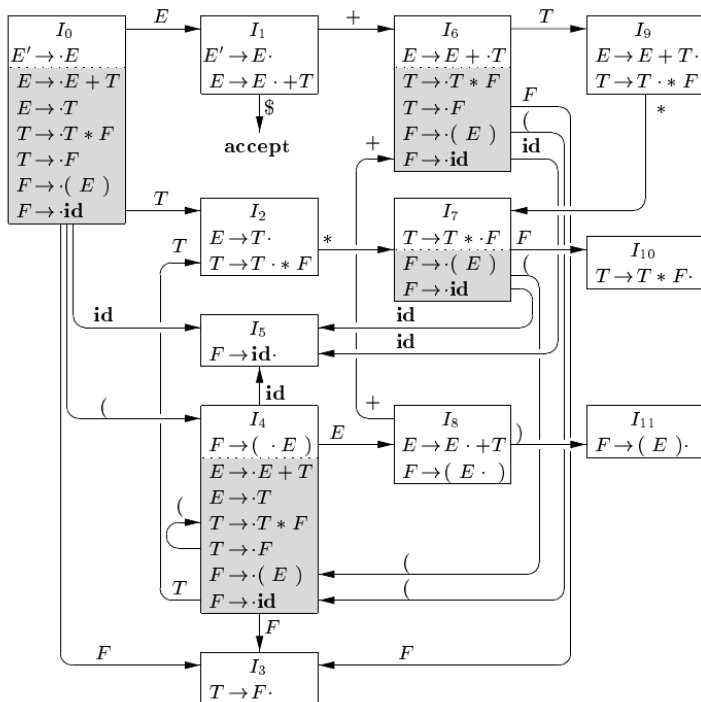
viable prefix عرض کنید در یک مرحله تجزیه $shift/reduce$ حرکت داریم، id داریم، $($ محتوای درون استک و $)$ رشته ای از ورودی است که هنوز حرکت نکرده، آنگاه id را **viable prefix** گوئیم

سوال: چه رشته هایی در **viable prefix** باشند؟

اگر در یک مرحله با کیفیت بودن بعدی درون استک، محتوای استک تبدیل به رشته ای شود که p کیفیت آن

رشته ای هرگز از یک DFA می یابیم که تمام رشته ای که در طایف DFA پذیرش می شوند id می آید آن کرامت هستند

مثالی از یک DFA



گرامر

$$\begin{aligned} E &\rightarrow E + T \mid T \\ T &\rightarrow T * F \mid F \\ F &\rightarrow (E) \mid id \end{aligned}$$

در یک ساخت DFA مورد نظر

تعریف آئیم: در تجزیه LR، $state$ ای دنبال می شوند تا به یاد داشته باشیم کجای دست یابیم هر $state$ مجموعه ای از آئیم است.

یک آئیم برای هر $production$ که گرامر همان $production$ است که در منتهی از بهینه آن - اضافه شده است.

به این آئیم، آئیم های $LR(0)$ گوئیم

مثال $T \rightarrow (E)$

$$\begin{cases} T \rightarrow \cdot (E) \\ T \rightarrow (\cdot E) \\ T \rightarrow (E \cdot) \\ T \rightarrow (E) \cdot \end{cases}$$

وَتَنی دایم $A \rightarrow 0.XYZ$ یعنی انتظار داریم X را ببینیم و تَنی داریم $A \rightarrow X.0.YZ$ یعنی X را دیدیم، انتظار داریم YZ را ببینیم

حرفه بعدی کا را ببینیم آوردن state های DFA که شامل کسم هستند، زیرا متناهی است DFA



به نام خدا

ساخت انرای $LR(0)$

- ۱- برای هر فرم α یکیل شروع α ، کرار اخذ α (augmented) یکیل شروع جدید $\alpha' \rightarrow \alpha$ ساخته می شود. بدین صورت پذیرش هر رشته در کرار فقط زمانی است که α به α' کاهش پیدا کند.
- ۲- $closure(I)$ برای هر مجموعه آتم I در کرار I (مجموعه ای از آتم) به صورت زیر ساخته می شود:
 - هر آتم α در I در $closure(I)$ است
 - اگر $A \rightarrow \alpha.B\beta$ ($B \in N$) در $closure(I)$ باشد و $\beta \rightarrow \gamma$ یک تولید (production) در زبان باشد، آنگاه $\alpha.B\gamma$ به $closure(I)$ اضافه می شود. این قانون را آنقدر تکرار می کنیم که آتم جدیدی به $closure$ اضافه نشود.

- تابع $goto(I, X)$: X یک یکیل در I مجموعه ای از آتم است
- اگر $[A \rightarrow \alpha.X\beta] \in I$ آنگاه $goto(I, X)$ برابر با $closure$ مجموعه آتم های $A \rightarrow \alpha.X\beta$ است.

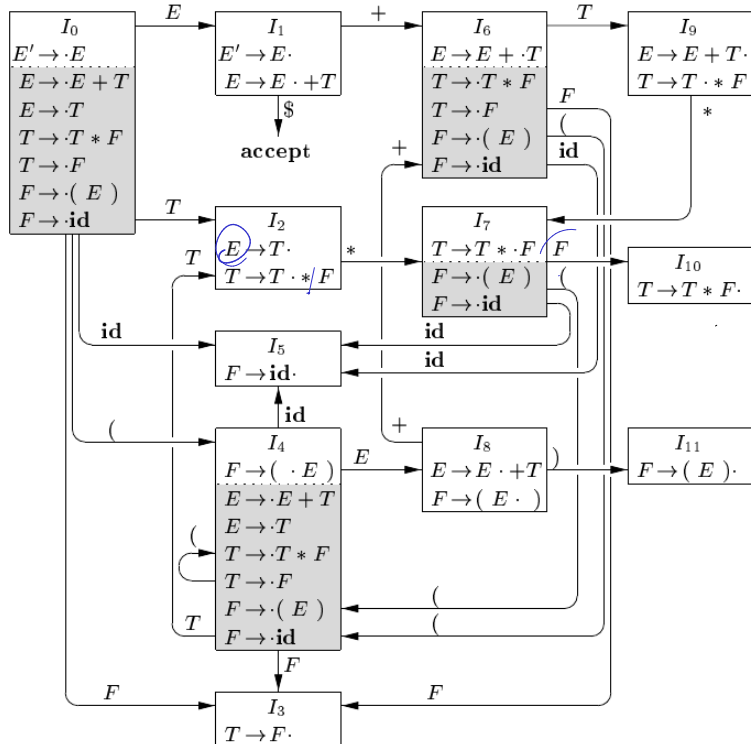
ساخت DFA

state DFA ساز مجموعه آتم است و تابع $goto$ ، انتقال δ را در DFA نشان می دهد

- state شروع در DFA حالتی است که $\alpha \rightarrow \alpha'$ است

- $goto(I, X)$ انتقالی از I است وقتی X دیده می شود

- هر state δ می تواند برای تشخیص $variable$ ، حالت نویی باشد

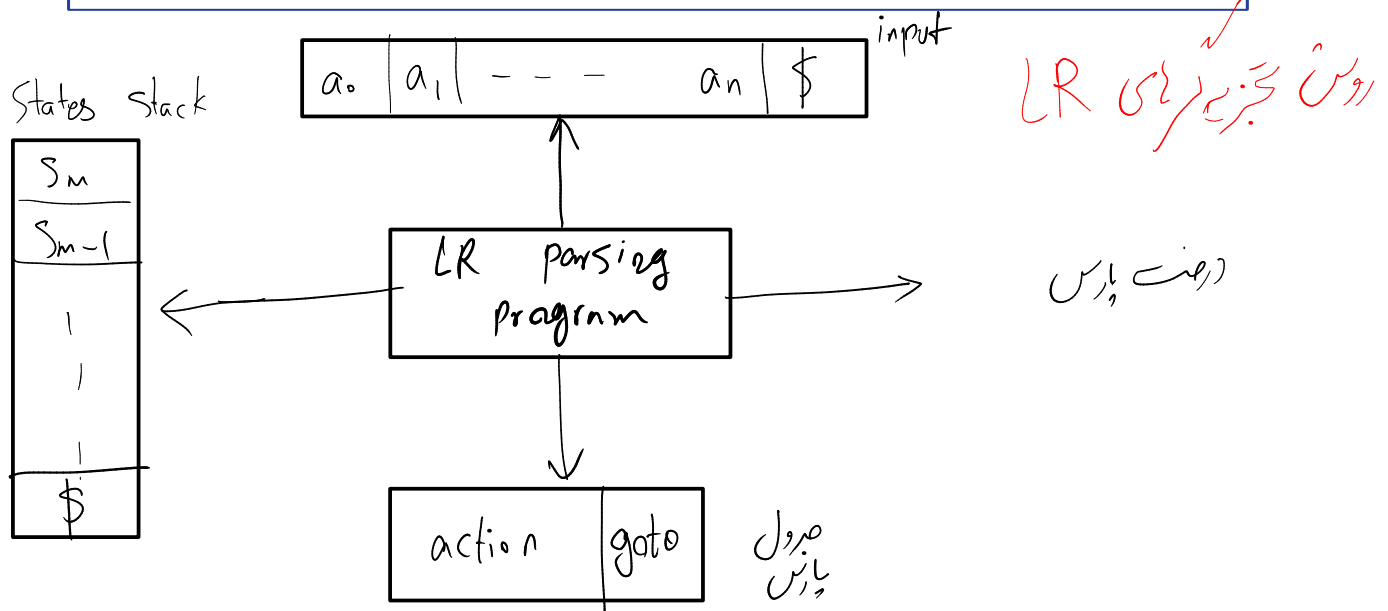


مثال تجزیه از طریق انزای $L R(0)$

id

state	stack	symbols stack	انزای	ation
\$0	\$		id * id \$	shift id
\$05	\$id		* id \$	reduce to F
\$03	\$F		* id \$	reduce to T
\$02	\$T		* id \$	shift * X
\$027	\$T*		id \$	shift id
\$0275	\$T*id		\$	reduce to F
\$02710	\$T*F		\$	reduce to T
\$02	\$T		\$	reduce to E
\$01	\$E		\$	reduce to E'
\$0	\$E'			accept

آخرین مرحله بارش وقتی است که state می باشد
 $S \rightarrow S'$ برسم در این state اگر ورودی نمانده باشد
 بسته پذیرش شده است



action: برای $action[i, a]$ که زینب state و a یک ترسیال است.

۱- shift: زینب state است در این حالت با ورودی a به این شیفت می‌دهد و به state جدید وارد می‌شود.

۲- reduce $A \rightarrow \beta$ اگر β روی استک است آن را به A کاهش بدهیم.

۳- accept

۴- error

فست goto: اگر $goto(I, A) = I'$ یعنی goto در محول پارس حالت را به I' می‌برد.

سکین غیر ترسیال A به حالت و متناهی می‌باشد.

state	action	goto
	terminal	non terminal