



بنام خدا

کنترل خطا در تجزیه با جدول LL1

1- panic mode

2- phrase level recovery

اوتن panic mode

۱- وقتی خطایی تشخیص داده شود توکن را نادیده می‌گیریم و قابل تجزیه  
 دیگر نیست ignore کنیم و از آن جابه‌جا به بعد تجزیه را ادامه دهیم.

ignore

مثال  $(1 + 2) + 3$  sync

توکن‌هایی که احتمالاً از آن توکن برعکس ورودی با توجه به جدول قابل تجزیه باشد: Synchronization tokens

(۱)-

استنتاج  
 $S\$ \xRightarrow{*} \alpha A \beta \rightarrow$   
 $\alpha \in T^* \quad \beta \in (T \cup N)^*$

ورودی:  $a_1 a_2 \dots a_i \dots a_{j-1} a_j \dots a_n$

$T[A, a_i]$  خالی

sync

از  $a_1$  تا  $a_i$  ignore و  $A$  را هم از استack pop می‌کنیم

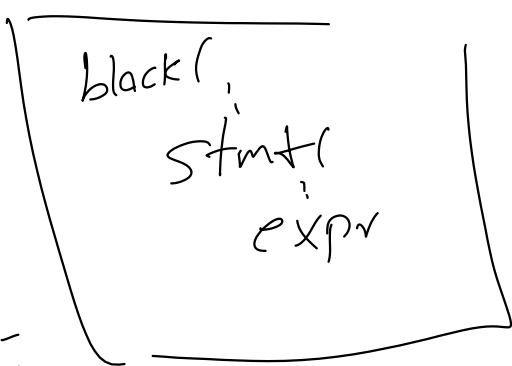
(فرض کردیم سینکس مربوط به غیرترسیال  $A$  درست نوشته نشده است) و تجزیه را از  $a_i$  با سینکس بعدی که در اصل حاضر روی استack است ادامه می‌دهیم.



- بنا بر این با توجه به بالا هم اعضای  $\text{Follow}(A)$  می توانند در محجره  $\text{Synchronization}$  قرار گیرند.  
یعنی اگر  $A$  ردی است بود که می توانیم  $\text{Follow}(A)$  را نادیده بگیریم  $\text{Skip}$  می کنیم و هم  $A$  هم  $\text{Pop}$  می شود (اگر  $A$  تکامل پذیر است) و  $A$   $\text{Pop}$  می شود در عوض  $a$  از ورودی  $\text{Ignore}$  می شود

(2)  $i = 100$  ;  $(a < b)$  ;  $\text{Exp}$  دارای خطا =  $i$  مثل

$\underbrace{\hspace{10em}}_{\text{stmt}} \quad \underbrace{\hspace{10em}}_{\text{stmt}}$



روش بالا همیشه کارایی نیست. مثلاً وقتی داریم  $\text{Skip}$  می کنیم باید بررسی کنیم که آیا  $\text{Follow}$  قرار دارد و همیشه منتظر نباشیم.  $\text{Sync}$  سیستم در این حالت می تواند از ساختارهای دیگری استفاده کند. مثلاً  $\text{Exp}$  درون  $\text{stmt}$  قرار دارد پس  $\text{stmt}$  سطح بالاتر  $\text{Exp}$  است پس می تواند  $\text{First}$  غیر ترسیال سطح بالاتر را در  $\text{Sync}$  غیر ترسیال سطح پایین تر در نظر گرفت.

$$\text{First}(\text{stmt}) \in \text{Sync}(\text{Exp})$$

(3) اگر  $\text{First}(A)$  به  $\text{Sync}$   $A$  اضافه شود می توان  $A$  را شروع از آن بکسل ادامه داد (بکسل می فیل از  $\text{First}(A)$  در ورودی  $\text{Ignore}$  شده)



(۴) اگر روی استک، ترسیال  $a$  باشد و در ورودی ترسیال  $b$  را بینیم ( $a \neq b$ )  
ترسیال روی استک را  $pop$  می‌کنیم و سعی در ادامه تجزیه می‌کنیم.

(۱) و (۲)، ۳: روی استک غیر ترسیال داریم

۴: روی استک ترسیال داریم

روش phrase level

در خانه دی‌خالی جدول LL1، فانکشن  $lookahead$  قرار می‌گیرد که همین تجزیه را می‌کند و در جدول بان  
خانه جدول، فانکشن  $lookahead$  قرار می‌گیرد.

به طور کلی فانکشن  $lookahead$  در رفع خطا از ورودی می‌کند مثلاً حذف یا اضافه یا تغییر توکن جاری  
در ورودی و یا  $pop$  کردن از استک. باید دید که چگونه تغییرات باعث افتادن در  $lookahead$   
نشود همچنین کمترین تغییر صورت گیرد و کامپایلر زمان هرگز

تولید خطا تشخیص خطای محول می‌باشد و سعی در اصلاح کردن آن را به هر

مثال  $E \rightarrow E * id \mid E id$

$E \rightarrow int \mid E + E \mid (E) \mid (error)$

$stmts \rightarrow stmt(i) stmts \mid stmt()$

$n = 2 \times b$   
+ C



(مثال)

$$\begin{cases} E \rightarrow TX \\ X \rightarrow + E \mid \varepsilon \\ T \rightarrow id \mid Y \mid (E) \\ Y \rightarrow *T \mid \varepsilon \end{cases}$$

	(	)	+	*	id	\$
E	TX	sync			TX	sync
T	(E)	sync	sync		idY	sync
X		$\varepsilon$	+E			$\varepsilon$
Y		$\varepsilon$	$\varepsilon$	*T		$\varepsilon$

ورودی ) id \* + id \$

استک	ورودی	action
E \$	) id * + id \$	error; skip) <span style="color: red;">E</span>
E \$	id * + id \$	$E \rightarrow TX$
TX \$	"	$T \rightarrow id \mid Y$
id Y X \$	"	match(id)
Y X \$	* + id \$	$Y \rightarrow *T$
*TX \$	"	match(*)
TX \$	+ id \$	error, popT <span style="color: red;">E</span>
X \$	+ id \$	
+ E \$	+ id \$	match(+)
E \$	id \$	
TX \$	id \$	

id Y X \$	id \$	match(id)
Y X \$	\$	$Y \rightarrow \varepsilon$
X \$	\$	$X \rightarrow \varepsilon$
\$	\$	✓