

Βοηθητικές έννοιες (i)

▶ Σύνολα **FIRST**

- ▶ Έστω συμβολοσειρά $\alpha \in (T \cup N)^*$
- ▶ Το σύνολο **FIRST**(α) $\subseteq T \cup \{\epsilon\}$ περιέχει τα τερματικά σύμβολα από τα οποία **αρχίζουν** οι συμβολοσειρές που παράγονται από την α
- ▶ Αν $\alpha \Rightarrow^* a\beta$ τότε $a \in \text{FIRST}(\alpha)$
- ▶ Αν $\alpha \Rightarrow^* \epsilon$ τότε $\epsilon \in \text{FIRST}(\alpha)$

Βοηθητικές έννοιες (ii)

▶ Σύνολα FOLLOW

- ▶ Έστω μη τερματικό σύμβολο A
- ▶ Το σύνολο $\text{FOLLOW}(A) \subseteq T \cup \{\text{EOF}\}$ περιέχει τα τερματικά σύμβολα που μπορούν να ακολουθούν το A στη διάρκεια μιας παραγωγής
- ▶ Αν το A μπορεί να είναι το τελευταίο σύμβολο σε μια παραγωγή, τότε $\text{EOF} \in \text{FOLLOW}(A)$
- ▶ Αν $S \Rightarrow^* \alpha A a \beta$ τότε $a \in \text{FOLLOW}(A)$
- ▶ Αν $S \Rightarrow^* \alpha A$ τότε $\text{EOF} \in \text{FOLLOW}(A)$

Υπολογισμός FIRST (i)

- ▶ $\text{FIRST}(\epsilon) = \{ \epsilon \}$
- ▶ $\text{FIRST}(a\beta) = \{ a \}$
- ▶ αν $\epsilon \notin \text{FIRST}(A)$
τότε $\text{FIRST}(A\beta) = \text{FIRST}(A)$
- ▶ αν $\epsilon \in \text{FIRST}(A)$
τότε $\text{FIRST}(A\beta) = (\text{FIRST}(A) - \{ \epsilon \}) \cup \text{FIRST}(\beta)$
- ▶ για κάθε κανόνα $A \rightarrow \alpha$, πρέπει $\text{FIRST}(\alpha) \subseteq \text{FIRST}(A)$

Υπολογισμός FIRST (ii)

► Παράδειγμα

$$\text{FIRST}(E) = \{\text{id}, (\}$$

$$\text{FIRST}(T) = \{\text{id}, (\}$$

$$\text{FIRST}(F) = \{\text{id}, (\}$$

$$\text{FIRST}(E') = \{+, \epsilon\}$$

$$\text{FIRST}(T') = \{*, \epsilon\}$$

$$E \rightarrow T E'$$

$$E' \rightarrow \epsilon$$

$$E' \rightarrow + T E'$$

$$T \rightarrow F T'$$

$$T' \rightarrow \epsilon$$

$$T' \rightarrow * F T'$$

$$F \rightarrow (E)$$

$$F \rightarrow \text{id}$$

Υπολογισμός FOLLOW (i)

- ▶ EOF \in FOLLOW(S)
- ▶ για κάθε κανόνα $A \rightarrow \alpha B \beta$
 - ▶ $(\text{FIRST}(\beta) - \{\epsilon\}) \subseteq \text{FOLLOW}(B)$
 - ▶ αν $\epsilon \in \text{FIRST}(\beta)$
τότε $\text{FOLLOW}(A) \subseteq \text{FOLLOW}(B)$

Υπολογισμός FOLLOW

(ii)

► Παράδειγμα

$$\text{FOLLOW}(E) = \{), \text{EOF} \}$$

$$\text{FOLLOW}(T) = \{ +,), \text{EOF} \}$$

$$\text{FOLLOW}(F) = \{ *, +,), \text{EOF} \}$$

$$\text{FOLLOW}(E') = \{), \text{EOF} \}$$

$$\text{FOLLOW}(T') = \{ +,), \text{EOF} \}$$

$$\text{FIRST}(E') = \{ +, \epsilon \}$$

$$\text{FIRST}(T') = \{ *, \epsilon \}$$

$$E \rightarrow T E'$$

$$E' \rightarrow \epsilon$$

$$E' \rightarrow + T E'$$

$$T \rightarrow F T'$$

$$T' \rightarrow \epsilon$$

$$T' \rightarrow * F T'$$

$$F \rightarrow (E)$$

$$F \rightarrow \text{id}$$

ΣΑ *top-down* (i)

- ▶ Η συντακτική ανάλυση ξεκινά από τη ρίζα
- ▶ Κάθε φορά, αναζητά:
 - ▶ το μη τερματικό σύμβολο που θα αντικατασταθεί
 - ⇒ συνήθως επιλέγεται το αριστερότερο
 - ▶ τον κανόνα παραγωγής που θα εφαρμοστεί
 - ⇒ βάσει των επόμενων k λεκτικών μονάδων στη συμβολοσειρά εισόδου: $LL(k)$
- ▶ Επαναλαμβάνεται μέχρι να εξαντληθούν τα μη τερματικά

Γραμματικές LL(1)

- ▶ Απαραίτητες προϋποθέσεις:
 - ▶ Απουσία **αριστερής αναδρομής** (άμεσης ή έμμεσης)
 - ▶ Απουσία **κοινού προθέματος** σε εναλλακτικούς κανόνες
- ▶ Μερικές φορές είναι δυνατός ο μετασχηματισμός μιας γραμματικής σε ισοδύναμη LL(1)
 - ⇒ απαλοιφή αριστερής αναδρομής
 - ⇒ αριστερή παραγοντοποίηση

Μετασχηματισμός σε LL(1)

▶ Αντικατάσταση

$$\begin{array}{l} A \rightarrow \alpha_1 \mid \dots \mid \alpha_n \\ B \rightarrow \beta_1 A \beta_2 \end{array} \quad \Rightarrow \quad \begin{array}{l} A \rightarrow \alpha_1 \mid \dots \mid \alpha_n \\ B \rightarrow \beta_1 \alpha_1 \beta_2 \mid \dots \mid \beta_1 \alpha_n \beta_2 \end{array}$$

▶ Αριστερή παραγοντοποίηση

$$\begin{array}{l} A \rightarrow \alpha \beta_1 \mid \dots \mid \alpha \beta_n \end{array} \quad \Rightarrow \quad \begin{array}{l} A \rightarrow \alpha B \\ B \rightarrow \beta_1 \mid \dots \mid \beta_n \end{array}$$

▶ Απαλοιφή άμεσης αριστερής αναδρομής

$$\begin{array}{l} A \rightarrow A\alpha_1 \mid \dots \mid A\alpha_n \mid \beta_1 \mid \dots \mid \beta_m \\ \Rightarrow \\ A \rightarrow \beta_1 B \mid \dots \mid \beta_m B \\ B \rightarrow \alpha_1 B \mid \dots \mid \alpha_n B \mid \epsilon \end{array}$$

ΣΑ αναδρομικής κατάβασης

$A \rightarrow \alpha_1 \mid \dots \mid \alpha_n$

μετατρέπεται σε κώδικα της μορφής:

if $token \in \text{FIRST}(\alpha_1)$ **then**

κώδικας για την αναγνώριση της α_1

...

else if $token \in \text{FIRST}(\alpha_n)$ **then**

κώδικας για την αναγνώριση της α_n

else if $\epsilon \notin \text{FIRST}(\alpha_1) \cup \dots \cup \text{FIRST}(\alpha_n)$ **then**

συντακτικό σφάλμα

else if $token \notin \text{FOLLOW}(A)$ **then**

συντακτικό σφάλμα

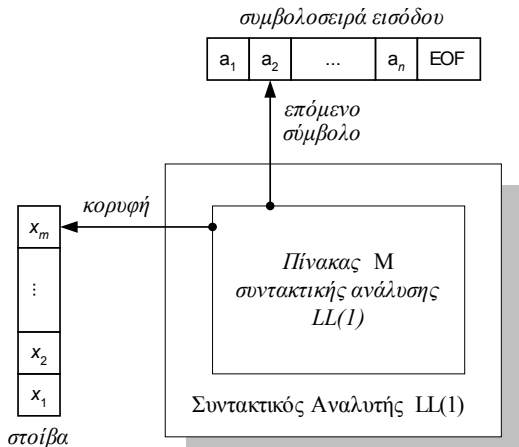
end if

ΣΑ LL(1) (i)

- ▶ Χρησιμοποιούν μια **στοίβα** όπου τοποθετούν σύμβολα της γραμματικής — αρχικά μόνο το S
- ▶ Κάθε φορά εξετάζεται η κορυφή της στοίβας:
 - ▶ Αν είναι **τερματικό** σύμβολο και είναι το ίδιο με το επόμενο της συμβολοσειράς εισόδου, τότε αφαιρούνται και τα δύο
 - ▶ Αν είναι **μη τερματικό σύμβολο**, τότε ανάλογα με το επόμενο της συμβολοσειράς εισόδου εφαρμόζεται κάποιος κανόνας
- ▶ **Επιτυχία:** η στοίβα και η συμβολοσειρά εισόδου είναι άδειες

ΣΑ LL(1) (ii)

Ο αλγόριθμος κατασκευής του πίνακα M ορίζει την οικογένεια των γλωσσών LL(1)



Κατασκευή ΣΑ LL(1)

$E \rightarrow T E'$	$\text{FIRST}(E) = \text{FIRST}(T) = \text{FIRST}(F) = \{\text{id}, (\}$
$E' \rightarrow + T E' \mid \epsilon$	$\text{FIRST}(E') = \{+, \epsilon\}$
$T \rightarrow F T'$	$\text{FIRST}(T') = \{*, \epsilon\}$
$T' \rightarrow * F T' \mid \epsilon$	$\text{FOLLOW}(E) = \text{FOLLOW}(E') = \{), \text{EOF}\}$
$F \rightarrow (E) \mid \text{id}$	$\text{FOLLOW}(T) = \text{FOLLOW}(T') = \{+,), \text{EOF}\}$
	$\text{FOLLOW}(F) = \{*, +,), \text{EOF}\}$

	id	+	*	()	EOF
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \epsilon$	$E' \rightarrow \epsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'		$T' \rightarrow \epsilon$	$T' \rightarrow *FT'$		$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$
F	$F \rightarrow \text{id}$			$F \rightarrow (E)$		

Λειτουργία ΣΑ LL(1)

0	E	id + id * id EOF	$E \rightarrow T E'$
1	$E' T$	id + id * id EOF	$T \rightarrow F T'$
2	$E' T' F$	id + id * id EOF	$F \rightarrow \text{id}$
3	$E' T' \text{id}$	id + id * id EOF	
4	$E' T'$	+ id * id EOF	$T' \rightarrow \epsilon$
5	E'	+ id * id EOF	$E' \rightarrow + T E'$
6	$E' T +$	+ id * id EOF	
7	$E' T$	id * id EOF	$T \rightarrow F T'$
8	$E' T' F$	id * id EOF	$F \rightarrow \text{id}$
9	$E' T' \text{id}$	id * id EOF	
10	$E' T'$	* id EOF	$T' \rightarrow * F T'$
11	$E' T' F *$	* id EOF	
12	$E' T' F$	id EOF	$F \rightarrow \text{id}$
13	$E' T' \text{id}$	id EOF	
14	$E' T'$	EOF	$T' \rightarrow \epsilon$
15	E'	EOF	$E' \rightarrow \epsilon$
16	ϵ	EOF	αναγνώριση