Per costruire il parser abbiamo bisogno di due concetti, FIRST e FOLLOW.

FIRST(α) = insieme dei terminali che iniziano le stringhe che derivano da α:

$= \{ a : A \Rightarrow a w_2 \text{ per qualche } w_2 \} \cup \{ \varepsilon, \text{ se } A \Rightarrow^* \varepsilon \}$

2 situazioni per cui derivo ε

$A \to \varepsilon$

$A \to B$
$B \to C$
$C \to \varepsilon$

G: $S \to AB$ $\quad$ $L(G) = \{ab, b\}$
$A \to a \mid \varepsilon$
$B \to b$

PARSING TABLE:

	a	b	\$
S	$S \to AB$	$S \to AB$	
A	$A \to a$	$A \to \varepsilon$	
B		$B \to b$	

Nella mia tabella devo trovare le direttive che supportano le derivazioni:

$S \Rightarrow AB \Rightarrow aB \Rightarrow ab$

$S \Rightarrow AB \Rightarrow B \Rightarrow b$

FIRST: Grammatica $G = (V, T, S, P)$ $\qquad S \Rightarrow^* a$

Per ogni simbolo $X \in V$, l'insieme first(X) è calcolato come segue:

1. Se $X = b$ (è un terminale, cioè $X \in T$) allora first(X) = $\{x\}$
2. $\forall X \in V \setminus T$ (è un non-terminale) inizializzare first(X) come ϕ e procedere come segue:
   - Se $X \to \varepsilon \in P$ (è una produzione), allora aggiungere ε a first(X).
   - Se $X \to Y_1 \dots Y_n$ con $n \geq 1$ e G, allora applicare la procedura seguente:

```
j := 1
while (j ≤ n) {
        aggiungere a first(X) ogni b tale che b ∈ (first(Y_j) \ {ε})
        if (ε ∈ first(Y_j)) then j=j+1 else break;
}
if (j = n+1) then aggiungere ε a first(X).
```

First(α) con α stringa di simboli di V

Sia $\alpha = Y_1 \dots Y_n$ con $n \geq 1$.

Inizializzare first(α) come ϕ e applicare la seguente procedura:

```
j := 1
while (j ≤ n) {
        aggiungere a first(α) ogni b tale che b ∈ first(Y_j)
        if (ε ∈ first(Y_j)) then j=j+1 else break;
}
if (j = n+1) then aggiungere ε a first(α).
```

$E \to TE'$
$E' \to +TE' \mid \varepsilon$
$T \to FT'$
$T' \to *FT' \mid \varepsilon$
$F \to (E) \mid id$

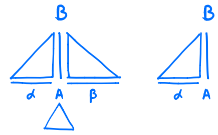
Ordine della compilazione

	first	infine
E	first(T) Se $\varepsilon \in$ first(T) guarda E'	id, (
E'	+, ε	+, ε
T	first(F). Se $\varepsilon \in$ first(F) guarda T'	id, (
T'	$*$, ε	$*$, ε
F	id, (	id, (

RICORDA:

first: sono terminali o ε : $\forall A$, first(A) $\subseteq T \cup \{\varepsilon\}$

FOLLOW(A) $G = (V, T, S, P)$



Per calcolarlo, seguire la seguente procedura:

- Per ogni $X \in V \setminus T \setminus \{S\}$, inizializzare $follow(X) = \emptyset$
- Inizializzare $Follow(S) = \{\$\}$
- repeat

  foreach $A \to \alpha X \beta \in P$ { % β è ε o nullable; produzioni in cui A compare nel body. α, β qualsiasi simbolo di G.

  aggiungere $first(\beta) \setminus \{\varepsilon\}$ a $follow(X)$

  if ($\beta = \varepsilon$ or $\varepsilon \in first(\beta)$) then % β è ε (stringa nulla) o first(β) contiene β

  aggiungere $follow(A)$ a $follow(X)$

  }

  until saturation (niente più cambia).

ESECUZIONE

	$A \to \alpha\ X\ \beta$	
1	$E \to T\ E'$	
2	$E' \to +\ T\ E'$	| ε
3	$T \to F\ T'$	
4	$T' \to *\ F\ T'$	| ε
5	$F \to (\ E\ )$	| id

		infine
E	\$ first(()\ε = (	\$)
E'	β= ε	\$)
T	first(E')= + β∈ first(E')	\$) +
T'	β= ε	\$) +
F	first(T')\ε = * β∈ first(T')	\$) + *

ESECUZIONE

	$A \to \alpha\ X\ \beta$	
1	$E \to T\ E'$	
2	$E' \to +\ T\ E'$	| ε
3	$T \to F\ T'$	
4	$T' \to *\ F\ T'$	| ε
5	$F \to (\ E\ )$	| id

		infine
E	\$)	
E'		
T		
T'		
F	*	