Warm-Up Problem

Consider the following Grammar

$$
\begin{array}{lcll}
S' & \rightarrow & \vdash S \dashv & (0) \\
S & \rightarrow & bSd & (1) \\
S & \rightarrow & pSq & (2) \\
S & \rightarrow & C & (3) \\
C & \rightarrow & rC & (4) \\
C & \rightarrow & \varepsilon & (5)
\end{array}
$$

Compute the nullable and first sets for non-terminal symbols of this grammar.

(And, follow if you're feeling confident; we haven't seen it yet!)

CS 241 Lecture 12

Top-Down Parsing, First, and Follow Continued

With thanks to Brad Lushman, Troy Vasiga, Kevin Lanctot, and Carmen Bruni

g253wang g253wang g253wang g253wang

Recall

Our notation:

$First(\beta) = \{a \in \Sigma : \beta \Rightarrow^* a\gamma, \text{ for some } \gamma \in V^*\}$

$Nullable(\beta) =$ true iff $\beta \Rightarrow^* \varepsilon$ and false otherwise

$Follow(A) = \{b \in \Sigma : S' \Rightarrow^* \alpha A b \beta \text{ for some } \alpha, \beta \in V^*\}$

Definition

We say that that a $\beta \in V^*$ is **nullable** if and only if Nullable(β) = true.

Updated Predictor Table Definition

Definition

$$\text{Predict}(A, \alpha) = \{A \to \beta : \alpha \in \text{First}(\beta)\} \cup \{A \to \beta : \text{Nullable}(\beta) \text{ and } \alpha \in \text{Follow}(A)\}$$

This is the full, correct definition. Notice that this still requires that the table only have one member of the set per entry to be useful as a deterministic algorithm.

g253wang g253wang g253wang

Computing First

Algorithm 3 First(A) for all $A \in N'$

```
1:  Initialize First(A) = {} for all A ∈ N'.
2:  repeat
3:      for each rule A → B1 B2 ··· Bk in P do
4:          for i ∈ {1, .., k} do
5:              if Bi ∈ T' then
6:                  First(A) = First(A) ∪ {Bi}; break
7:              else
8:                  First(A) = First(A) ∪ First(Bi)
9:                  if Nullable(Bi) == False then break
10:             end if
11:         end for
12:     end for
13: until nothing changes
```

Computing First (2)

Algorithm 4 First(β) where $\beta = B_1 \cdots B_n \in V^*$

```
1: result = ∅
2: for i ∈ {1, .., n} do
3:     if B_i ∈ T' then
4:         result = result ∪ {B_i}; break
5:     else
6:         result = result ∪ First(B_i)
7:         if Nullable(B_i) == False then break
8:     end if
9: end for
```

g253wang g253wang g253wang

Computing Follow

Algorithm 5 Follow(A) for all $A \in N$ (Recall N $= N' \setminus \{ S' \}$)

1: Initialize Follow(A) $= \{\}$ for all $A \in N$.
2: **repeat**
3: **for** each production $A \to B_1 B_2 \cdots B_k$ in P' **do**
4: **for** $i \in \{1, .., k\}$ **do**
5: **if** $B_i \in N$ **then**
6: Follow(B_i) $=$ Follow(B_i) $\cup$ First($B_{i+1} \cdots B_k$)
7: **if** $\bigwedge_{m=i+1}^{k}$ Nullable(B_m) == True or i == k **then**
8: Follow(B_i) $=$ Follow(B_i) $\cup$ Follow(A)
9: **end if**
10: **end if**
11: **end for**
12: **end for**
13: **until** nothing changes

Example of Follow

$S' \rightarrow \vdash S \dashv$
$S \rightarrow c$
$S \rightarrow QRS$
$Q \rightarrow R$
$Q \rightarrow d$
$R \rightarrow \varepsilon$
$R \rightarrow b$

Follow Table:

Iter	0	1	2
S	{}	{⊣}	{⊣}
Q	{}	{b, c, d}	{b, c, d}
R	{}	{b, c, d}	{b, c, d}

The above makes use of the fact that

$$\text{First}(RS) = \text{First}(R) \cup \text{First}(S) = \{b, c, d\}$$

where the first equality holds since R is nullable.

Recap

S'	$\rightarrow$	$\vdash S \dashv$	(0)
S	$\rightarrow$	c	(1)
S	$\rightarrow$	QRS	(2)
Q	$\rightarrow$	R	(3)
Q	$\rightarrow$	d	(4)
R	$\rightarrow$	ε	(5)
R	$\rightarrow$	b	(6)

Data:

	Nullable	First	Follow
S'	False	{⊢}	{}
S	False	{b, c, d}	{⊣}
Q	True	{b, d}	{b, c, d}
R	True	{b}	{b, c, d}

Definition

$$\text{Predict}(A, \alpha) = \{A \rightarrow \beta : \alpha \in \text{First}(\beta)\}$$
$$\cup \{A \rightarrow \beta : \beta \text{ is nullable and } \alpha \in \text{Follow}(A)\}$$

Compute the Predict Table for the grammar.

Predict Table

Definition

$$\text{Predict}(A, \alpha) = \{A \to \beta : \alpha \in \text{First}(\beta)\} \cup \{A \to \beta : \beta \text{ is nullable and } \alpha \in \text{Follow}(A)\}$$

$S' \to \vdash S \dashv$	(0)
$S \to c$	(1)
$S \to QRS$	(2)
$Q \to R$	(3)
$Q \to d$	(4)
$R \to \varepsilon$	(5)
$R \to b$	(6)

	$\vdash$	b	c	d	$\dashv$
S'	{0}				
S		{2}	{1, 2}	{2}	
Q		{3}	{3}	{3, 4}	
R		{5, 6}	{5}	{5}	

Cheat Sheet and Examples

Nullable:

- $A \rightarrow \varepsilon$ implies that Nullable(A) = true.
  Further Nullable(ε) = true.
- If $A \rightarrow B_1 \ldots B_n$ and each of Nullable(B_i) = true then
  Nullable(A) = true.

First:

- $A \rightarrow \alpha\alpha$ then $\alpha \in$ First(A)
- $A \rightarrow B_1 \ldots B_n$ then First(A) = First(A) $\cup$ First(B_i) for each
  $i \in \{1, .., n\}$ until Nullable(B_i) is false.

Follow:

- $A \rightarrow \alpha B \beta$ then Follow(B) = First(β)
- $A \rightarrow \alpha B \beta$ and Nullable(β) = true, then
  Follow(B) = Follow(B) $\cup$ Follow(A)

Predict(A, α) = $\{A \rightarrow \beta : \alpha \in$ First(β)$\}$
$\cup \{A \rightarrow \beta : \beta$ is nullable and $\alpha \in$ Follow(A)$\}$

For More Practice

Check out

http://smlweb.cpsc.ucalgary.ca/start.html

g253wang g253wang g253wang

Practice

Construct the four tables (Nullable, First, Follow and Predict) for the following examples:

G_1

$$
\begin{array}{lcll}
S' & \rightarrow & \vdash S \dashv & (0) \\
S & \rightarrow & Bb & (1) \\
S & \rightarrow & Cd & (2) \\
B & \rightarrow & aB & (3) \\
B & \rightarrow & \varepsilon & (4) \\
C & \rightarrow & cC & (5) \\
C & \rightarrow & \varepsilon & (6)
\end{array}
$$

G_2

$$
\begin{array}{lcll}
S' & \rightarrow & \vdash S \dashv & (0) \\
S & \rightarrow & TZ' & (1) \\
Z' & \rightarrow & +TZ' \mid \varepsilon & (2,3) \\
T & \rightarrow & FT' & (4) \\
T' & \rightarrow & *FT' \mid \varepsilon & (5,6) \\
F & \rightarrow & a \mid b \mid c & (7,8,9)
\end{array}
$$

For G_1

	Nullable	First	Follow
S'	False	$\{\vdash\}$	$\{\}$
S	False	$\{a, b, c, d\}$	$\{\dashv\}$
B	True	$\{a\}$	$\{b\}$
C	True	$\{c\}$	$\{d\}$

Predict

	$\vdash$	a	b	c	d	$\dashv$
S'	0					
S		1	1	2	2	
B		3	4			
C				5	6	

g253wang g253wang g253wang

For G_2

	Nullable	First	Follow
S′	False	{⊢}	{}
S	False	{a, b, c}	{⊣}
Z′	True	{+}	{⊣}
T	False	{a, b, c}	{⊣, +}
T′	True	{∗}	{⊣, +}
F	False	{a, b, c}	{⊣, +, ∗}

Predict (Recall: Nullable(ε) is true).

	⊢	a	b	c	+	∗	⊣
S′	0						
S		1	1	1			
Z′					2		3
T		4	4	4			
T′					6	5	6
F		7	8	9			

g253wang g253wang g253wang