

CS 442/542

Parsing Part 3

First Set

- For any grammar symbol α , $\text{First}(\alpha)$ is the set of terminals or epsilon that can appear at the start of a sentence derived from α

Follow Set

- For any nonterminal α , $\text{Follow}(\alpha)$ is the set of terminals that can occur immediately after α in a sentence

First⁺

- $\text{First}^+(A \rightarrow B) = \{ t \mid \text{if } \varepsilon \notin \text{First}(B) \text{ then } t \in \text{First}(B) \text{ else } t \in \text{First}(B) \cup \text{Follow}(A) \}$
- Also called the Predict set
- Note this definition includes ε in the First⁺ set. Some definitions exclude it and in the way this set will be used in the algorithm to build the LL(1) parse table ε will not be important

Build First Set

```
for each  $\alpha \in (T \cup \epsilon)$  do;
    FIRST( $\alpha$ )  $\leftarrow$   $\alpha$ ;
end;
for each  $A \in NT$  do;
    FIRST( $A$ )  $\leftarrow$   $\emptyset$ ;
end;
while (FIRST sets are still changing) do;
    for each  $p \in P$ , where  $p$  has the form  $A \rightarrow \beta$  do;
        if  $\beta$  is  $\beta_1 \beta_2 \dots \beta_k$ , where  $\beta_i \in T \cup NT$ , then begin;
            rhs  $\leftarrow$  FIRST( $\beta_1$ ) -  $\{\epsilon\}$ ;
            i  $\leftarrow$  1;
            while ( $\epsilon \in$  FIRST( $\beta_i$ ) and  $i \leq k-1$ ) do;
                rhs  $\leftarrow$  rhs  $\cup$  (FIRST( $\beta_{i+1}$ ) -  $\{\epsilon\}$ );
                i  $\leftarrow$  i + 1;
            end;
            if i = k and  $\epsilon \in$  FIRST( $\beta_k$ )
                then rhs  $\leftarrow$  rhs  $\cup$   $\{\epsilon\}$ ;
            FIRST( $A$ )  $\leftarrow$  FIRST( $A$ )  $\cup$  rhs;
        end;
    end;
end;
```

Build the First Sets for the Following Grammar

1. Goal \rightarrow E
2. E \rightarrow T E1
3. E1 \rightarrow + T E1
4. E1 \rightarrow - T E1
5. E1 \rightarrow ϵ
6. T \rightarrow F T1
7. T1 \rightarrow * F T1
8. T1 \rightarrow / F T1
9. T1 \rightarrow ϵ
10. F \rightarrow (E)
11. F \rightarrow num
12. F \rightarrow name

First(Goal) = (, num, name

First(E) = (, num, name

First(E1) = +, -, ϵ

First(T) = (, num, name

First(T1) = *, / , ϵ

First(F) = (, num, name

Build the First Sets for the Following Grammar

1. Goal \rightarrow E
2. E \rightarrow T E1
3. E1 \rightarrow + T E1
4. E1 \rightarrow - T E1
5. E1 \rightarrow ϵ
6. T \rightarrow F T1
7. T1 \rightarrow * F T1
8. T1 \rightarrow / F T1
9. T1 \rightarrow ϵ
10. F \rightarrow (E)
11. F \rightarrow num
12. F \rightarrow name

First(Goal) = First(E)

First(E) = First(T)

First(E1) = +, -, ϵ

First(T) = First(F)

First(T1) = *, / , ϵ

First(F) = (, num, name

Build the First Sets for the Following Grammar

1. Goal \rightarrow E
2. E \rightarrow T E1
3. E1 \rightarrow + T E1
4. E1 \rightarrow - T E1
5. E1 \rightarrow ϵ
6. T \rightarrow F T1
7. T1 \rightarrow * F T1
8. T1 \rightarrow / F T1
9. T1 \rightarrow ϵ
10. F \rightarrow (E)
11. F \rightarrow num
12. F \rightarrow name

First(Goal) = First(E)

First(E) = First(T)

First(E1) = +, -, ϵ

First(T) = First(F)

= (, num, name

First(T1) = *, / , ϵ

First(F) = (, num, name

Build the First Sets for the Following Grammar

1. Goal \rightarrow E
2. E \rightarrow T E1
3. E1 \rightarrow + T E1
4. E1 \rightarrow - T E1
5. E1 \rightarrow ϵ
6. T \rightarrow F T1
7. T1 \rightarrow * F T1
8. T1 \rightarrow / F T1
9. T1 \rightarrow ϵ
10. F \rightarrow (E)
11. F \rightarrow num
12. F \rightarrow name

First(Goal) = First(E)

First(E) = First(T)

= (, num, name

First(E1) = +, -, ϵ

First(T) = First(F)

= (, num, name

First(T1) = *, / , ϵ

First(F) = (, num, name

Build the First Sets for the Following Grammar

1. Goal \rightarrow E
2. E \rightarrow T E1
3. E1 \rightarrow + T E1
4. E1 \rightarrow - T E1
5. E1 \rightarrow ϵ
6. T \rightarrow F T1
7. T1 \rightarrow * F T1
8. T1 \rightarrow / F T1
9. T1 \rightarrow ϵ
10. F \rightarrow (E)
11. F \rightarrow num
12. F \rightarrow name

First(Goal) = First(E)
= (, num, name

First(E) = First(T)
= (, num, name

First(E1) = +, -, ϵ

First(T) = First(F)
= (, num, name

First(T1) = *, / , ϵ

First(F) = (, num, name

Build Follow Sets

```
for each  $A \in NT$  do;
    FOLLOW( $A$ )  $\leftarrow \emptyset$ ;
end;

FOLLOW( $S$ )  $\leftarrow \{\text{eof}\}$ ;

while (FOLLOW sets are still changing) do;
    for each  $p \in P$  of the form  $A \rightarrow \beta_1 \beta_2 \dots \beta_k$  do;
        TRAILER  $\leftarrow$  FOLLOW( $A$ );
        for  $i \leftarrow k$  down to 1 do;
            if  $\beta_i \in NT$  then begin;
                FOLLOW( $\beta_i$ )  $\leftarrow$  FOLLOW( $\beta_i$ )  $\cup$  TRAILER;
                if  $\epsilon \in \text{FIRST}(\beta_i)$ 
                    then TRAILER  $\leftarrow$  TRAILER  $\cup$  (FIRST( $\beta_i$ ) -  $\epsilon$ );
                else TRAILER  $\leftarrow$  FIRST( $\beta_i$ );
            end;
            else TRAILER  $\leftarrow$  FIRST( $\beta_i$ ); // is  $\{\beta_i\}$ 
        end;
    end;
end;
```

Build the Follow Sets for the Following Grammar

1. Goal \rightarrow E
2. E \rightarrow T E1
3. E1 \rightarrow + T E1
4. E1 \rightarrow - T E1
5. E1 \rightarrow ϵ
6. T \rightarrow F T1
7. T1 \rightarrow * F T1
8. T1 \rightarrow / F T1
9. T1 \rightarrow ϵ
10. F \rightarrow (E)
11. F \rightarrow num
12. F \rightarrow name

Follow(Goal) = eof

Follow(E) = eof,)

Follow(E1) = eof,)

Follow(T) = +, -, eof,)

Follow(T1) = +, -, eof,)

Follow(F) = *, /, +, -, eof,)

Build the Follow Sets for the Following Grammar

1. Goal \rightarrow E
2. E \rightarrow T E1
3. E1 \rightarrow + T E1
4. E1 \rightarrow - T E1
5. E1 \rightarrow ϵ
6. T \rightarrow F T1
7. T1 \rightarrow * F T1
8. T1 \rightarrow / F T1
9. T1 \rightarrow ϵ
10. F \rightarrow (E)
11. F \rightarrow num
12. F \rightarrow name

Follow(Goal) = eof
Follow(E) = Follow(Goal),)
Follow(E1) = Follow(E)
Follow(T) = First(E1)
Follow(T1) = Follow(T)
Follow(F) = First(T1)

Build the Follow Sets for the Following Grammar

1. Goal \rightarrow E
2. E \rightarrow T E1
3. E1 \rightarrow + T E1
4. E1 \rightarrow - T E1
5. E1 \rightarrow ϵ
6. T \rightarrow F T1
7. T1 \rightarrow * F T1
8. T1 \rightarrow / F T1
9. T1 \rightarrow ϵ
10. F \rightarrow (E)
11. F \rightarrow num
12. F \rightarrow name

Follow(Goal) = eof

Follow(E) = Follow(Goal),)
= eof,)

Follow(E1) = Follow(E)
= eof,)

Follow(T) = First(E1)
= +, -, Follow(E1),
Follow(E)

Follow(T1) = Follow(T)
= +, -, Follow(E1),
Follow(E)

Follow(F) = First(T1)
= *, /, Follow(T),
Follow(T1)

Build the Follow Sets for the Following Grammar

1. Goal \rightarrow E
2. E \rightarrow T E1
3. E1 \rightarrow + T E1
4. E1 \rightarrow - T E1
5. E1 \rightarrow ϵ
6. T \rightarrow F T1
7. T1 \rightarrow * F T1
8. T1 \rightarrow / F T1
9. T1 \rightarrow ϵ
10. F \rightarrow (E)
11. F \rightarrow num
12. F \rightarrow name

Follow(Goal) = eof
Follow(E) = Follow(Goal),)
= eof,)
Follow(E1) = Follow(E)
= eof,)
Follow(T) = First(E1)
= +, -, Follow(E1),
Follow(E)
= +, -, eof,)
Follow(T1) = Follow(T)
= +, -, Follow(E1),
Follow(E)
= +, -, eof,)
Follow(F) = First(T1)
= *, /, Follow(T),
Follow(T1)
= *, /, +, -, eof,)

Build LL(1) Parse Table

```
build FIRST, FOLLOW, and FIRST+ sets;

for each nonterminal A do;
  for each terminal w do;
    Table[A,w] ← error;
  end;

  for each production p of the form A → β do;
    for each terminal w ∈ FIRST+(A → β) do;
      Table[A,w] ← p;
    end;

    if eof ∈ FIRST+(A → β)
      then Table[A,eof] ← p;
    end;
  end;
end;
```


Build the LL(1) Parse Table for the Following Grammar

1. Goal $\rightarrow E$
2. $E \rightarrow T E_1$
3. $E_1 \rightarrow + T E_1$
4. $E_1 \rightarrow - T E_1$
5. $E_1 \rightarrow \epsilon$
6. $T \rightarrow F T_1$
7. $T_1 \rightarrow * F T_1$
8. $T_1 \rightarrow / F T_1$
9. $T_1 \rightarrow \epsilon$
10. $F \rightarrow (E)$
11. $F \rightarrow \text{num}$
12. $F \rightarrow \text{name}$

	eof	+	-	*	/	()	num	name
G						1		1	1
E						2		2	2
E ₁	5	3	4				5		
T						6		6	6
T ₁	9	9	9	7	8		9		
F						10		10	10

Build the First+ Sets for the Following Grammar

1. Goal \rightarrow E	First(E) = (, num, name
2. E \rightarrow T E1	First(T) = (, num, name
3. E1 \rightarrow + T E1	+
4. E1 \rightarrow - T E1	-
5. E1 \rightarrow ϵ	Follow(E1) = eof,)
6. T \rightarrow F T1	First(F) = (, num, name
7. T1 \rightarrow * F T1	*
8. T1 \rightarrow / F T1	/
9. T1 \rightarrow ϵ	Follow(T1) = +, -, eof,)
10. F \rightarrow (E)	(
11. F \rightarrow num	num
12. F \rightarrow name	name

Build the LL(1) Parse Table for the Following Grammar

1. Goal $\rightarrow E$
2. $E \rightarrow T E_1$
3. $E_1 \rightarrow + T E_1$
4. $E_1 \rightarrow - T E_1$
5. $E_1 \rightarrow \epsilon$
6. $T \rightarrow F T_1$
7. $T_1 \rightarrow * F T_1$
8. $T_1 \rightarrow / F T_1$
9. $T_1 \rightarrow \epsilon$
10. $F \rightarrow (E)$
11. $F \rightarrow \text{num}$
12. $F \rightarrow \text{name}$

	eof	+	-	*	/	()	num	name
G						1		1	1
E						2		2	2
E ₁	5	3	4				5		
T						6		6	6
T ₁	9	9	9	7	8		9		
F						10		10	10

Build the First, Follow and First+ Sets and the LL(1) Parse Table for the Following Grammar

1. $S \rightarrow B$
2. $B \rightarrow B1 B2$
3. $B2 \rightarrow OR B1 B2$
4. $B2 \rightarrow \epsilon$
5. $B1 \rightarrow B3 B4$
6. $B4 \rightarrow AND B3 B4$
7. $B4 \rightarrow \epsilon$
8. $B3 \rightarrow VAR$