COMP 520: Compilers
Sample Solutions – Written Assignment 2

(Note: the rules applied in these solutions can be found in lecture 4, slides 6 – 19)

1. For the following grammars, find the Nullable, Starters, and Followers sets for the nonterminals and justify whether or not the grammar meets the LL(1) condition.

Grammar (a)  
\[ S ::= A \; B \; c \]

\[ A ::= a \; B \mid \varepsilon \]

\[ B ::= b \mid \varepsilon \]

First determine the grammar properties by finding the fixpoint of the relevant equations.

1. Nullable
\[ N_0(S) = N_0(A) = N_0(B) = \text{false} \]
\[ N_{i+1}(S) = N_i(A) \land N_i(B) \land N(c) = N_i(A) \land N_i(B) \land \text{false} = \text{false} \]
\[ N_{i+1}(A) = (N(a) \land N_i(B)) \lor N(\varepsilon) = (\text{false} \land N_i(B)) \lor \text{true} = \text{true} \]
\[ N_{i+1}(B) = N(b) \lor N(\varepsilon) = \text{false} \lor \text{true} = \text{true} \]

No iteration is needed since the equations are at the fixpoint solution.

2. Starters
\[ ST_0(S) = \{ \}, \quad ST_0(A) = ST_0(B) = \{ \varepsilon \} \]
\[ ST_{i+1}(S) = ST_i(A) \bigoplus ST_i(B) \bigoplus ST(c) = (ST_i(A) - \{ \varepsilon \}) \cup (ST_i(B) - \{ \varepsilon \}) \cup \{ c \} \]
\[ ST_{i+1}(A) = (ST(a) \bigoplus ST_i(B)) \cup ST(\varepsilon) = \{ a \} \cup \{ \varepsilon \} = \{ a, \varepsilon \} \quad \text{fixpoint} \]
\[ ST_{i+1}(B) = ST(b) \cup ST(\varepsilon) = \{ b \} \cup \{ \varepsilon \} = \{ b, \varepsilon \} \quad \text{fixpoint} \]

Iterate: \[ ST_1(S) = \{ c \}, \quad ST_2(S) = \{ a, b, c \} \quad \text{fixpoint} \]

3. Followers
\[ FL_0(S) = \{ \} \quad \text{S not on RHS of any rule} \]
\[ FL_0(A) = ST(B \; c) - \{ \varepsilon \} = (\{ b, \varepsilon \} \bigoplus ST(c)) = \{ b, c \} \quad \text{A on RHS rule 1} \]
\[ FL_0(B) = (ST(c) \cup ST(\varepsilon)) - \{ \varepsilon \} = \{ c \} \quad \text{B on RHS rule 1,2} \]

Since \( A ::= a \; B \) in rule 2 is of the form \( A ::= a \; By \), where \( y \) is nullable, we must add the followers of \( A \) into the followers of \( B \):

\[ FL_{i+1}(B) = FL_i(B) \cup FL_i(A) = \{ c \} \cup \{ b, c \} = \{ b, c \} \]

As \( FL_i(B) \) is fully defined, no further iteration is needed and \( FL(B) = FL_1(B) = \{ b, c \} \).

Finally we need to add \( \varepsilon \) to \( FL(N) \) for any nonterminal \( N \) that can appear as the last symbol in a derivation starting from \( S \). As this is an augmented grammar in which (a) the start symbol \( S \) does not appear on the RHS of any rule and (b) the rule for \( S \) ends in a terminal (\( c \) in this case), we can be sure only \( FL(S) \) includes \( \{ \varepsilon \} \). So \( FL(S) = \{ \varepsilon \} \).
In summary we have

<table>
<thead>
<tr>
<th></th>
<th>Nullable</th>
<th>Starters</th>
<th>Followers</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>F</td>
<td>{a, b, c}</td>
<td>{c}</td>
</tr>
<tr>
<td>A</td>
<td>T</td>
<td>{a, c}</td>
<td>{b, c}</td>
</tr>
<tr>
<td>B</td>
<td>T</td>
<td>{b, c}</td>
<td>{b, c}</td>
</tr>
</tbody>
</table>

To determine whether the LL(1) condition is met for this grammar, inspect all choice points to determine if they define the predict function unambiguously. There are only two choice points: the alternatives in the second and third rules.

- A ::= aB | \(\varepsilon\)
  
  Predict\((aB)\) = Starters\((aB)\) \(\oplus\) Followers\((A)\) = \{a\}
  Predict\((\varepsilon)\) = Starters\((\varepsilon)\) \(\oplus\) Followers\((A)\) = \{b, c\}
  
  The predict sets are disjoint, so the LL(1) condition is met.

- B ::= b | \(\varepsilon\)
  
  Predict\((b)\) = Starters\((b)\) \(\oplus\) Followers\((B)\) = \{b\}
  Predict\((\varepsilon)\) = Starters\((\varepsilon)\) \(\oplus\) Followers\((B)\) = \{b, c\}
  
  The predict sets are not disjoint, so the LL(1) condition is not met.

Since the LL(1) condition is not met by at least one choice point in the grammar, this grammar is not LL(1).

Grammar (b)  
S ::= A s
A ::= B c A | \(\varepsilon\)
B ::= b B | a A

1. Nullable. By inspection it is easy to determine that only A is nullable.

2. Starters.
   
   \(ST_0(S) = ST_0(B) = \{\}\), \(ST_0(A) = \{\varepsilon\}\)
   
   \(ST_{i+1}(S) = ST_i(A) \oplus ST(s) = (ST_i(A) - \{\varepsilon\}) \cup \{s\}\)
   
   \(ST_{i+1}(A) = (ST_i(B) \oplus ST(cA)) \cup ST(\varepsilon) = ST_i(B) \cup \{\varepsilon\}\)
   
   \(ST_{i+1}(B) = ST(bbB) \cup ST(aA) = \{b\} \cup \{a\} = \{a, b\}\) (fixpoint)
   
   Iterate:
   
   \(ST_1(S) = \{s\}\)  \(ST_2(S) = \{a, b, s\}\) (fixpoint)
   
   \(ST_1(A) = \{a, b, \varepsilon\}\)  \(ST_2(A) = \{a, b, \varepsilon\}\) (fixpoint)

3. To compute followers, note A occurs on the right in all three rules, B occurs on the right in two rules, AND S does not occur on the right in any rule. Thus our initial definitions are

   FL_0(S) = \{\}\n   FL_0(A) = (starters(s) \cup starters(c) \cup starters(\varepsilon)) - \{\varepsilon\} = \{s\}\n   FL_0(B) = (starters(cA) \cup starters(\varepsilon)) - \{\varepsilon\} = \{c\}\n

Now we need to account for any occurrence of a nonterminal \( X \) in a rule \( C \Rightarrow \alpha X \beta \) where \( \beta \) is nullable, because in this case we need to also include \( FL(C) \) in the \( FL(X) \).

When \( X = A \) this happens when \( B \Rightarrow aA \) in the third rule, which means we must include \( FL(B) \) and also happens when \( A \Rightarrow BcA \) in the second rule, which means we must also include \( FL(A) \). Thus we have

\[
FL_{i+1}(A) = FL_i(A) \cup (FL_i(B) \cup FL_i(A)) = FL_i(A) \cup FL_i(B)
\]

When \( X = B \) this happens when \( B \Rightarrow bB \) in the third rule, which means we must include \( FL(B) \) in \( FL(B) \). Thus,

\[
FL_{i+1}(B) = FL_i(B) \cup FL_i(B) = FL_i(B).
\]

Finally, since \( S \) does not have an occurrence in any right hand side, nothing needs so be added so

\[
FL_{i+1}(S) = FL_i(S)
\]

Iterating these rules we find the fixpoint solution at \( FL_2 \).

- \( FL_1(A) = FL_0(A) \cup FL_0(B) = \{c, s\} \)
- \( FL_1(B) = FL_0(B) = \{c\} \)
- \( FL_1(S) = FL_0(S) = \{\} \)

Since this grammar is in augmented form, we only need to add \( \epsilon \) to \( FL(S) \). So \( FL(S) = \{\epsilon\} \). In summary,

<table>
<thead>
<tr>
<th></th>
<th>Nullable</th>
<th>Starters</th>
<th>Followers</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>F</td>
<td>{a, b, s}</td>
<td>{\epsilon}</td>
</tr>
<tr>
<td>A</td>
<td>T</td>
<td>{a, b, \epsilon}</td>
<td>{c, s}</td>
</tr>
<tr>
<td>B</td>
<td>F</td>
<td>{a, b}</td>
<td>{c}</td>
</tr>
</tbody>
</table>

**LL(1) Condition:** There are two choice points in the rules.

- A ::= B c A | ε
  - \( Predict(BcA) = Starters(BcA) \oplus Followers(A) = Starters(B) = \{a, b\} \)
  - \( Predict(\epsilon) = Starters(\epsilon) \oplus Followers(A) = Followers(A) = \{c, s\} \)
  - The predict sets are disjoint.

- B ::= b B | a A
  - \( Predict(bB) = Starters(bB) \oplus Followers(B) = \{b\} \)
  - \( Predict(aA) = Starters(aA) \oplus Followers(B) = \{a\} \)
  - The predict sets are disjoint.

This predict sets are disjoint for each choice point, hence this grammar is LL(1).
Grammar (c)  \[ S ::= A \, c \]
\[ A ::= a A^* b \mid b \]

The Nullable and Starters functions are straightforward to compute. In this grammar \( A \) appears on the RHS in rule 1 where it is followed by \( c \), and it appears on the RHS in rule 2 where it is either followed by \( b \) or by \( ST(A) \) (since \( A^* \) can yield consecutive occurrences of \( A \)). So we have

\[
FL_0(S) = \{ \}
\]
\[
FL_0(A) = (\{c\} \cup ST(A^* b)) - \{\epsilon\} = \{c\} \cup \{a, b\} \cup \{b\} = \{a, b, c\}
\]

Since every occurrence of \( A \) in a rule is followed by a terminal, there are no other followers to include. Since this grammar is in augmented form, \( FL(S) = \{\epsilon\} \). In summary:

<table>
<thead>
<tr>
<th></th>
<th>Nullable</th>
<th>Starters</th>
<th>Followers</th>
</tr>
</thead>
<tbody>
<tr>
<td>( S )</td>
<td>F</td>
<td>{a, b}</td>
<td>{c}</td>
</tr>
<tr>
<td>( A )</td>
<td>F</td>
<td>{a, b}</td>
<td>{a, b, c}</td>
</tr>
</tbody>
</table>

**LL(1) Condition:** The first rule has no choice points. There are two choice points in the second rule. Working from the inside out:

- \( a A^* b \mid b \)
  
  \[
  \text{Predict}\left( a A^* b \right) = \{a\} \text{ and Predict}(b) = \{b\}
  \]
  The predict sets are disjoint, so the LL(1) condition is met at this choice point.

- \( A^* b \)
  
  First we need that \( A \) is not nullable - this is the case from the table above. Second we need to be able to predict repetition or termination of the Kleene star. For repetition: \( \text{Predict}(A) = \text{Starters}(A) = \{a, b\} \). For predict end of the repetition: \( \text{Predict}(b) = \{b\} \). The predict sets are not disjoint, so the LL(1) condition is not met at this choice point.

Since the LL(1) condition is not met by at least one choice point in the grammar, this grammar is not LL(1).