CS323 Assignment 3 Sample Answers

Exercise 1 (Grammar Basics)

Consider the following Grammar G:

$$S \to aSbS \mid bSaS \mid \epsilon$$

1. Yes. [3 points]

2. A sample leftmost derivation for the string *aabbab* is as follows.

$$\begin{aligned} S &\Rightarrow aSbS \\ &\Rightarrow aaSbSbS \\ &\Rightarrow aabSbS \\ &\Rightarrow aabbSaSbS \\ &\Rightarrow aabbab \end{aligned}$$

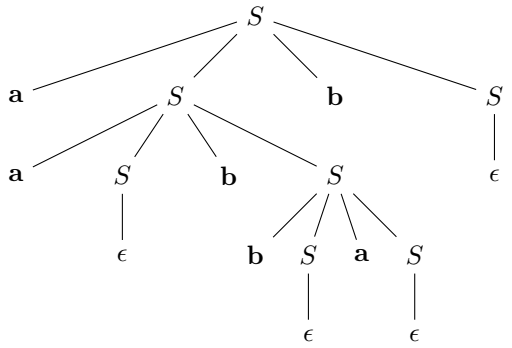
[8 points]

3. A sample rightmost derivation for the string *aabbab* is as follows.

$$\begin{aligned} S &\Rightarrow aSbS \\ &\Rightarrow aSbaSbS \\ &\Rightarrow aSbab \\ &\Rightarrow aaSbSbab \\ &\Rightarrow aabbab \end{aligned}$$

[8 points]

4. A possible parse tree is as follows.



[6 points]

Exercise 2 (Top-Down Parsing)

Consider the following Grammar G:

$$S \to aSbS \mid bSaS \mid \epsilon$$

1. Compute the FIRST and FOLLOW sets for G.

Computing FIRST set for G:
Terminal symbols:
FIRST$(a) = \{a\}$, FIRST$(b) = \{b\}$.
Nonterminal symbol S:
For $S \to aSbS$, add a to FIRST(S): FIRST$(S) = \{a\}$.
For $S \to bSaS$, add b to FIRST(S): FIRST$(S) = \{a, b\}$.
For $S \to \epsilon$, add ϵ to FIRST(S): FIRST$(S) = \{a, b, \epsilon\}$.

Computing FOLLOW set for nonterminal S:
Add \$ in FOLLOW(S): FOLLOW$(S) = \{\$\}$
For production $S \to aSbS$: FOLLOW(S) = FOLLOW$(S) \cup$ FIRST$(b) = \{\$\} \cup \{b\} = \{\$, b\}$
For production $S \to bSaS$: FOLLOW(S) = FOLLOW$(S) \cup$ FIRST$(a) = \{\$, b\} \cup \{a\} = \{\$, a, b\}$
[15 points]

2. Construct the predictive parsing table for G.

Constructing the parsing table M:
For $S \to aSbS$: FIRST$(aSbS)$ = FIRST$(a) = \{a\}$, add $S \to aSbS$ to $M[S, a]$.
For $S \to bSaS$: FIRST$(bSaS)$ = FIRST$(b) = \{b\}$, add $S \to bSaS$ to $M[S, b]$.
For $S \to \epsilon$: FOLLOW$(S) = \{\$, a, b\}$, add $S \to \epsilon$ to $M[S, a]$, $M[S, b]$, and $M[S, \$]$.

Nonterminal	Input symbol		
	a	b	\$
S	$S \to aSbS$ $S \to \epsilon$	$S \to bSaS$ $S \to \epsilon$	$S \to \epsilon$

[15 points]

3. This grammar is not LL(1) since there are conflicts in the parsing table.
[5 points]

4. Resolving conflicts in the parsing table:

Nonterminal	Input symbol		
	a	b	\$
S	$S \to aSbS$	$S \to \epsilon$	$S \to \epsilon$

Parsing the string "ababab":

Matched	**Stack**	**Input**	**Action**
	$S\$$	$ababab\$$	
	$aSbS\$$	$ababab\$$	output $S \to aSbS$
a	$SbS\$$	$babab\$$	match a
a	$bS\$$	$babab\$$	output $S \to \epsilon$
ab	$S\$$	$abab\$$	match b
ab	$aSbS\$$	$abab\$$	output $S \to aSbS$
aba	$SbS\$$	$bab\$$	match a
aba	$bS\$$	$bab\$$	output $S \to \epsilon$
$abab$	$S\$$	$ab\$$	match b
$abab$	$aSbS\$$	$ab\$$	output $S \to aSbS$
$ababa$	$SbS\$$	$b\$$	match a
$ababa$	$bS\$$	$b\$$	output $S \to \epsilon$
$ababab$	$S\$$	\$	match b
$ababab$	\$	\$	output $S \to \epsilon$

[10 points]

Exercise 3 (Grammar Rewrite and Parsing)

Consider the following context-free grammar G:

$$S \to SS+ \mid SS- \mid a$$

Is it possible, by modifying the grammar in any way, to construct a predictive parser for the language L(G)? If yes, please modify the grammar and provide the predictive parsing table. Otherwise, please state the reason.

First, rewriting the grammar by eliminating the left-recursion:

$$S \to aS'$$
$$S' \to S + S' \mid S - S' \mid \epsilon$$

Second, applying the left factorization:

$$S \to aS'$$
$$S' \to ST \mid \epsilon$$
$$T \to +S' \mid -S'$$

Then, computing the FIRST and FOLLOW sets:
FIRST$(S) = \{a\}$
FIRST$(S') = \{a, \epsilon\}$
FIRST$(T) = \{+, -\}$
FOLLOW$(S) = \{+, -, \$\}$
FOLLOW$(S') = \{+, -, \$\}$
FOLLOW$(T) = \{+, -, \$\}$
Finally, Constructing the parsing table:

Nonterminal	Input symbol			
	a	+	-	$
S	$S \to aS'$			
S'	$S' \to ST$	$S' \to \epsilon$	$S' \to \epsilon$	$S' \to \epsilon$
T		$T \to +S'$	$T \to -S'$	

[30 points]