CMSC430 Spring 2014 Midterm 1 Solutions

S →	A			
A →	Aa	|	ε

1. (20 pts) LL(1) parsing
   a. (8 pts) Compute FIRST and FOLLOW for S, A for the above grammar, where ε is the empty string:

	FIRST	FOLLOW
S	**a, ε**	**$**
A	**a, ε**	**a, $**

   b. (10 pts) Construct the LL(1) parse table for the grammar.

	a	$
S	**S → A**	**S → A**
A	**A → Aa** **A → ε**	**A → ε**

   c. (2 pts) Is the grammar LL(1)? Explain.

No, since there are multiple entries for A for lookahead a.

2. (6 pts) Table-driven LL(1) parsing

Using the following LL(1) parse table, parse the input "baa". You only need to show the stack and remaining input at each stage of the parse.

	a	b	$
S	S → ba	S → AaC	S → C
A	A → ba	A → bBa	A → ε
B	B → ε	B → a	B → ε
C	C → ab	C → ba	C → ε

Stack	Input
$ S	**b a a $**
$ C a A	**B a a $**
$ C a a B b	**B a a $**
$ C a a B	**A a $**
$ C a a	**A a $**
$ C a	**A $**
$ C	**$**
$	**$**

3. (10 pts) Shift-reduce parsing

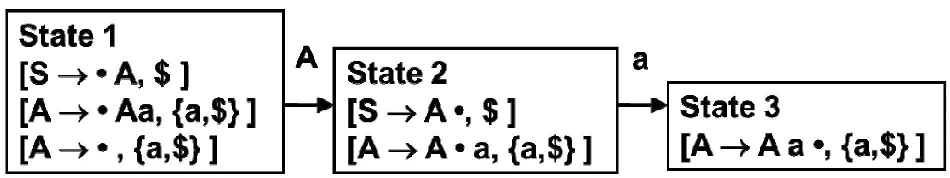
Given the ACTION/GOTO table above, show the parse if the current stack contents are 4 A 2 (i.e., current state is 2) and the remaining input is “bb”.

State	ACTION			GOTO	
	b	c	$	S	A
0	Shift 2	Reduce $S \rightarrow Bb$	Reduce $S \rightarrow cB$	1	3
1	Shift 4	Reduce $S \rightarrow c$	Reduce $S \rightarrow c$	3	2
2	Shift 3	Shift 4	Accept	2	4
3	Reduce $A \rightarrow Ab$	Reduce $A \rightarrow b$	Reduce $A \rightarrow Ab$	2	1
4	Reduce $A \rightarrow \varepsilon$	Shift 3	Accept	4	2

Stack	Input	Action
4 A 2	b b $	**shift 3**
4 A 2 b 3	**b $**	**reduce A→ Ab**
4 A	**b $**	**goto 2**
4 A 2	**b $**	**shift 3**
4 A 2 b 3	**$**	**reduce A→ Ab**
4 A	**$**	**goto 2**
4 A 2	**$**	**accept**

4. (22 pts) LR(1) parsers
   a. (16 pts) Given the following grammar, derive its canonical sets of LR(1) items.

S →	A
A →	Aa | ε



   b. (2 pts) Is the grammar LR(1)? Explain.

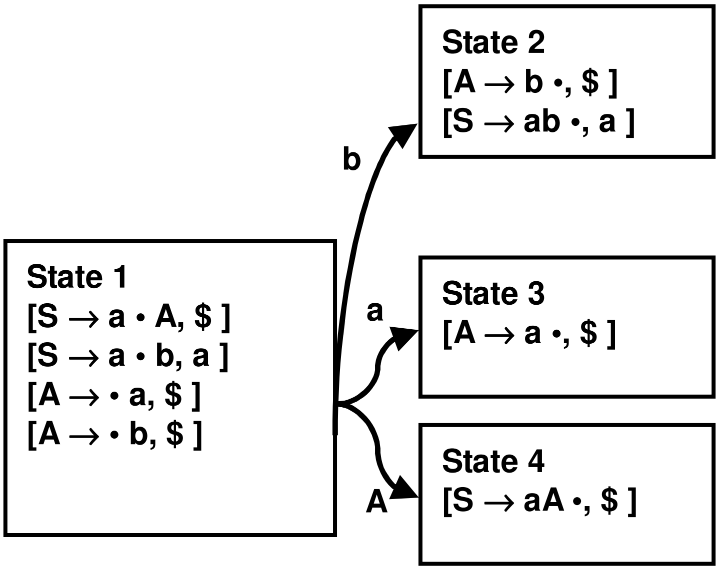
Yes, since there are no shift-reduce or reduce-reduce errors.

   c. (4 pts) How does this grammar example help explain a major difference between top-down and bottom-up parsers?

Bottom up parsers can handle left recursive grammars (with productions such as A→Aa), but top-down parsers cannot.

5. (16 pts) LR(1) parse table construction

Given the following sets of LR(1) items, construct the corresponding entries in the LR(1) parse table.



State	ACTION			GOTO	
	a	b	$	S	A
1	**Shift 3**	**Shift 2**			**4**
2	**Reduce $S \rightarrow ab$**		**Reduce $A \rightarrow b$**		
3			**Reduce $A \rightarrow a$**		
4			**Reduce $S \rightarrow aA$**		

6. (12 pts) LR(1) conflicts

Consider the following LR(1) items in a single state of a shift/reduce parser. List any conflicts that exist and describe for what lookaheads they occur.

[A → • dAd, b]
[B → dab • h, a]
[C → b • ad, c]
[D → ab • d, h]
[E → cab •, a]
[F → aAab • , c]
[G → hab • , c]
[H → aA • , b]

Shift/reduce conflict:
[C → b • ad, c] and [E → cab •, a] for lookahead a

Reduce/reduce conflict:
[F → aAab • , c] and [G → hab • , c for lookahead c

7.

(14 pts) LALR(1) parsers

a. (6 pts) Consider the following states in a LR(1) parser for a grammar G. Which states would be merged in a LALR(1) parser?

State 1	State 2	State 3	State 4
$[A \to ab \bullet, c]$	$[A \to ab \bullet, c]$	$[A \to ab \bullet, c]$	$[A \to ab \bullet, b]$
$[A \to b \bullet, a]$	$[A \to b \bullet, b]$	$[A \to \bullet\, b, c]$	$[A \to b \bullet, c]$
$[A \to \bullet\, ba, b]$	$[A \to b \bullet a, c]$	$[A \to b \bullet a, c]$	$[A \to b \bullet a, c]$

States 2 and 4.

b. (4 pts) What new state(s) would result after the merge?

$[A \to ab \bullet, b]$
$[A \to ab \bullet, c]$
$[A \to b \bullet, b]$
$[A \to b \bullet, c]$
$[A \to b \bullet a, c]$

or

$[A \to ab \bullet, \{b,c\}]$
$[A \to b \bullet, \{b,c\}]$
$[A \to b \bullet a, c]$

c. (4 pts) Is the grammar G an LALR(1) grammar? Explain.

No, since there would be a reduce/reduce conflict on b and c.