

CMSC430 Spring 2011 Midterm 1

Name _____

Instructions

- You have 50 minutes for to take this exam.
- This exam has a total of 100 points. An average of 30 seconds per point.
- This is a closed book exam. No notes or other aids are allowed.
- If you have a question, please raise your hand and wait for the instructor.
- Answer essay questions concisely using 1-2 sentences. Longer answers are not necessary and a penalty may be applied.
- In order to be eligible for partial credit, show all of your work and clearly indicate your answers.
- Write neatly. Credit cannot be given for illegible answers.

	Problem	Score	Max Score
1	Compiler front ends		6
2	LL(1) parsing		26
3	Shift-reduce parsing		12
4	LR(1) canonical		20
5	LR(1) parse table construction		12
6	LR(1) conflicts		12
7	SLR(1) & LALR(1) parsers		6
	Total		100

1. (6 pts) Compiler front ends
 - a. (3 pts) Explain how bottom-up parsing is different from top-down parsing.
 - b. (3 pts) Explain how a LR parser can generate a right-most derivation even though it scans input from left to right
2. (32 pts) LL(1) parsing
 - a. (12 pts) Compute FIRST and FOLLOW for S, A, B, and C for the following grammar, where ϵ is the empty string:

S	→	AB		Cd
A	→	a		Bf
B	→	b		ϵ
C	→	c		ϵ

- b. (10 pts) Using the FIRST and FOLLOW information provided, construct the LL(1) parse table for the following grammar.

$S \rightarrow aA \mid A$
$A \rightarrow bA \mid \epsilon$

	FIRST	FOLLOW
S	a, b, ϵ	\$
A	b, ϵ	\$

- c. (2 pts) Is the grammar in part 2(b) LL(1)? Explain.

- d. (8 pts) Using the following LL(1) parse table, parse the input “ba”. You only need to show the stack and remaining input at each stage of the parse.

	a	b	\$
S	$S \rightarrow ba$	$S \rightarrow A$	$S \rightarrow C$
A	$A \rightarrow ba$	$A \rightarrow bBa$	$A \rightarrow \epsilon$
B	$B \rightarrow \epsilon$	$B \rightarrow a$	$B \rightarrow \epsilon$
C	$C \rightarrow ab$	$C \rightarrow ba$	$C \rightarrow \epsilon$

3. (12 pts) Shift-reduce parsing

Given the following ACTION/GOTO table, show the parse if the current stack contents are 0 b 4 B 1 (i.e., current state is 1) and the remaining input is “d\$”.

State	ACTION			GOTO	
	d	e	\$	S	B
0	Shift 2	Reduce $S \rightarrow Bb$	Reduce $S \rightarrow bB$	2	1
1	Shift 3	Reduce $B \rightarrow d$	Accept	3	4
2	Shift 1	Shift 4	Accept	2	4
3	Reduce $S \rightarrow Bd$	Reduce $B \rightarrow d$	Reduce $S \rightarrow bBd$	1	2
4	Reduce $B \rightarrow e$	Shift 3	Reduce $S \rightarrow bSe$	4	3

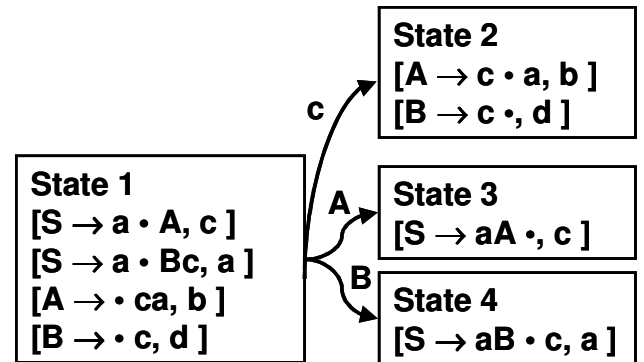
4. (20 pts) LR(1) canonical sets of items

Given the following augmented grammar, derive the following two states in its canonical sets of LR(1) items: the starting state, and the state reached from the starting state after shifting “a”.

$S' \rightarrow$	S
$S \rightarrow$	$Aha \mid ab$
$A \rightarrow$	$aAc \mid d$
$B \rightarrow$	b

5. (12 pts) LR(1) parse table construction

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



6. (12 pts) LR(1) conflicts

- a. (8 pts) 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.

[$A \rightarrow a \cdot bAb$, d]
[$B \rightarrow a \cdot bc$, c]
[$C \rightarrow da \cdot da$, a]
[$D \rightarrow aa \cdot d$, c]
[$E \rightarrow ca \cdot c$, a]
[$F \rightarrow aAa \cdot$, a]
[$G \rightarrow ba \cdot$, a]
[$H \rightarrow Ha \cdot$, c]

- b. (4 pts) Describe how compilers may intentionally use shift-reduce parsers containing conflicts.

7. (6 pts) SLR(1) & LALR(1) parsers

a. (3 pts) What is the main advantage of SLR(1) and LALR(1) parsers, as compared to LR(1) parsers?

b. (3 pts) What is the main disadvantage of SLR(1) and LALR(1) parsers, as compared to LR(1) parsers?