

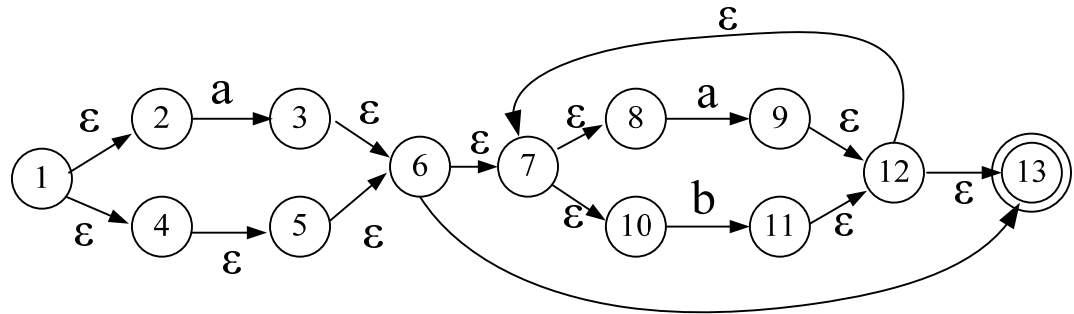
CS 352 Midterm One (2/12/09)

name _____

1. For the regular expression of $(a|\epsilon)(a|b)^*$, do the following.
 - a. (15 p) Translate the expression to a NFA .
 - b. (20 p) Translate the NFA to a DFA.

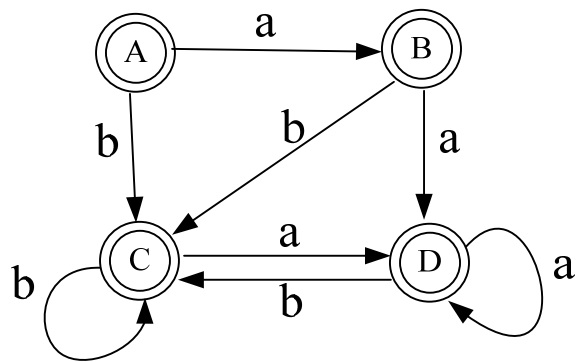
Note: Showing intermediate steps will help gaining partial credits.

(a)



(b)

	a	b
A={1,2,4,5,6,7,8,10,13}	{3,6,7,8,9,10,12,13}	{7,8,10,11,12,13}
B={3,6,7,8,9,10,12,13}	{7,8,9,10,12,13}	{7,8,10,11,12,13}
C={7,8,10,11,12,13}	{7,8,9,10,12,13}	{7,8,10,11,12,13}
D={7,8,9,10,12,13}	{7,8,9,10,12,13}	{7,8,10,11,12,13}



2. Given the following grammar

$S \rightarrow aS'$

$S' \rightarrow bS' \mid \epsilon$

a. (10 p) Are $aS'S'$, abS' , bS' , $aS'b$, bb sentential forms (**Correction: a sentence is also a sentential form**)?

Only abS' is.

b. (10 p) Compute the FIRST and FOLLOW sets for the non-terminals.

$FIRST(S) = \{a\}$; $FIRST(S') = \{b, \epsilon\}$

$FOLLOW(S) = \{\$ \}$ $FOLLOW(S') = \{\$ \}$

c. (10 p) Fill in the following LL(1) parse table.

	a	B	\$
S	$S \rightarrow aS'$	-	-
S'	-	$S' \rightarrow bS'$	$S' \rightarrow \epsilon$

d. (10 p) Show the steps of parsing “abb” using the LL(1) parsing algorithm in a table similar to the following.

Stack	Input
S\$	abb\$
aS'\$	abb\$
S'\$	bb\$
bS'\$	bb\$
S'\$	b\$
bS'\$	b\$
S'\$	\$
\$	\$

e. (15 p) If the rule of S changes to $S \rightarrow aS'b$, what are the new FIRST and FOLLOW sets (5 p)? Can you briefly explain why the grammar is not LL(1) (3 p)? In what situation does the parser have difficulty predicting the production rule (3 p)? Is the grammar ambiguous? (4 p).

FIRST sets are the same; $FOLLOW(S) = \{\$ \}$ $FOLLOW\{S'\} = \{b\}$

$FOLLOW(bS')$ has overlap with $FOLLOW(S')$.

The parser runs into trouble if it tries to use S' to parse (part of) the input string and it sees 'b', it cannot decide to use $S' \rightarrow bS'$ or $S \rightarrow \epsilon$

- f. (Extra Credit 5 p) Rewrite the grammar in (e) to make it LL(1) (Hint: first derive the sentences described by the grammar and then rewrite based on the sentence pattern).

$S \rightarrow abS'$

$S' \rightarrow bS' \mid \epsilon$

3. (10 p) Regular Expressions.

- a. (4 p) Strings over the alphabet $\{a, b\}$ with only one b.
- b. (3 p) Strings over the alphabet $\{a, b\}$ with an even number of b's.
- c. (3 p) Strings over the alphabet $\{a, b\}$ with an odd number of b's.

a. a^*ba^*

b. $a^*(ba^*ba^*)^*$

c. $a^*ba^*(ba^*ba^*)^*$