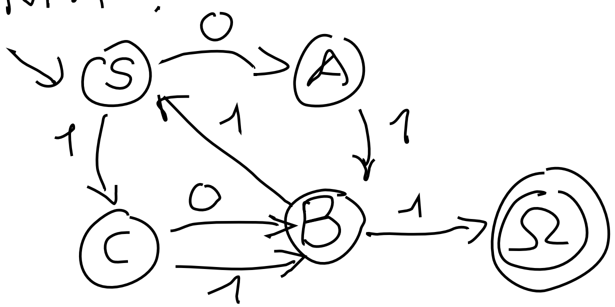
Formal Languages and Compilers Theory test February 26, 2021

1. Find the minimum-state DFA that accepts the language generated by the following grammar:

G=({A,B,C,S}, {0,1}, P, S)

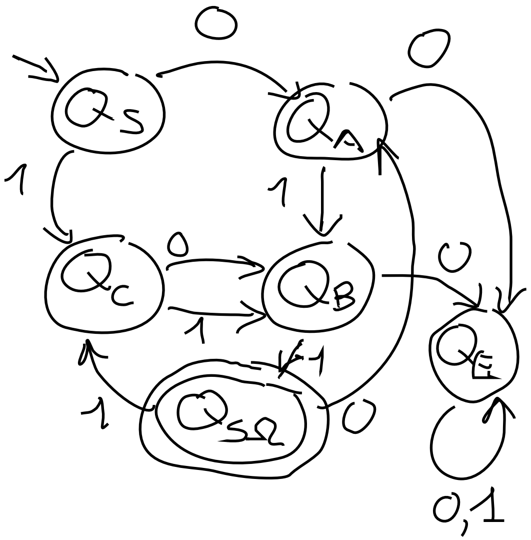
P= { $S \rightarrow 0A \mid 1C$
$A \rightarrow 1B$
$B \rightarrow 1S \mid 1$
$C \rightarrow 0B \mid 1B$
}

NFA :



Conversion To DFA :

		0	1
Q_S	$\{S\}$	$\{A\}$	$\{C\}$
Q_A	$\{A\}$	$\{\}$	$\{B\}$
Q_C	$\{C\}$	$\{B\}$	$\{B\}$
Q_B	$\{B\}$	$\{\}$	$\{S,\Omega\}$
$Q_{S\Omega}$	* $\{S,\Omega\}$	$\{A\}$	$\{C\}$
Q_E	$\{\}$	$\{\}$	$\{\}$



Minimization of DFA :

$\Pi_0 : \{Q_S, Q_A, Q_B, Q_C, Q_E\}, \{Q_{S\Omega}\}$

$\Pi_1 : \{Q_S, Q_A, Q_C, Q_E\}, \{Q_B\}, \{Q_{S\Omega}\}$

$\Pi_2 : \{Q_S, Q_E\}, \{Q_A\} \{Q_C\} \{Q_B\} \{Q_{S\Omega}\}$

$\Pi_3 : \{Q_S\}, \{Q_E\}, \{Q_A\}, \{Q_C\}, \{Q_B\}, \{Q_{S\Omega}\}$

$\Rightarrow$ the DFA is already minimum-state

Formal Languages and Compilers Theory test February 26, 2021

2. Write a CFG that generates the language $\{ a^n b^m c b^m d^{n+1} \mid n > 0, m > 0 \}$

$$a^n b^m c b^m d^{n+1} = \underbrace{\underbrace{a^n \underbrace{b^m c b^m}_{B} d^n}_{A} d}_{S}$$

$S \rightarrow A d$

$A \rightarrow a A d \mid a B d$

$B \rightarrow b B b \mid b c b$

3. Assume we want to build a top-down parser for the language generated by the following grammar

$S \rightarrow x \mid S^* \mid (S)$

Tell if it is necessary to modify the grammar and why. Then, if necessary, write the modified grammar.

Then, find the LL(1) parsing table and tell if the grammar used to build it is LL(1) or not (motivate your answer).

The grammar must be modified because it is left-recursive. Left recursion must be eliminated, otherwise top-down parsing may not terminate.

Modified grammar: $S \rightarrow xR \mid (S)R$
$R \rightarrow *R \mid \varepsilon$

Construction of the LL(1) parsing table

	nullable	FIRST	FOLLOW
S	F	x, (	\$,)
R	T	*	\$,)
x	F	x	*, \$,)
*	F	*	*, \$,)
(	F	(	x, (
)	F	)	*, \$,)

	x	*	(	)	\$
S	S->xR		S->(S)R		
R		R->*R		R->ε	R->ε

As there are no conflicts, the grammar is LL(1).