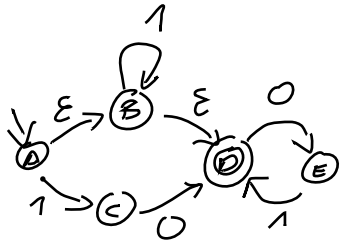
Formal Languages and Compilers Theory test July 19, 2022

1. (9 points) Find the minimum-state DFA that accepts the language denoted by the following regular expression:
(1* | 10) (01)*

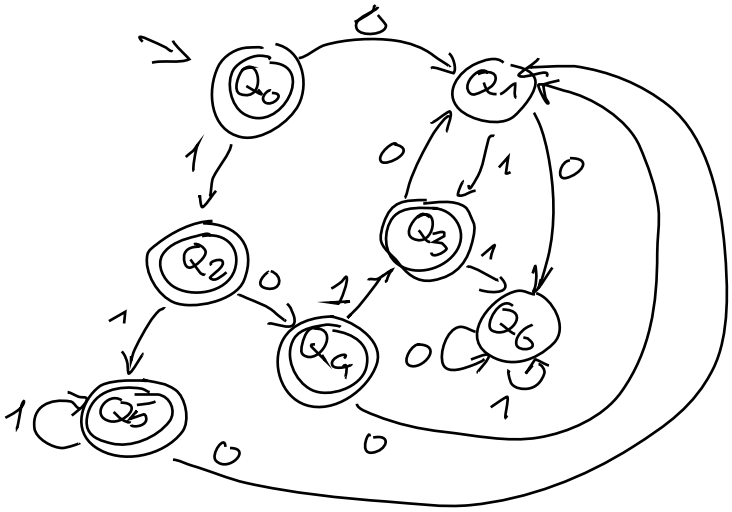
NFA construction:



This is an ε-NFA

Now we build an equivalent DFA:

		0	1
Q_0	* {A,B,D}	{E}	{B,C,D}
Q_1	{E}	{}	{D}
Q_2	* {B,C,D}	{D,E}	{B,D}
Q_3	* {D}	{E}	{}
Q_4	* {D,E}	{E}	{D}
Q_5	* {B,D}	{E}	{B,D}
Q_6	{}	{}	{}



DFA Minimization

The DFA is already fully specified

Π_0 : {Q_0, Q_2, Q_3, Q_4, Q_5}, {Q_1, Q_6}

Π_1 : {Q_0, Q_4, Q_5}, {Q_2}, {Q_3}, {Q_1}, {Q_6}

Π_2 : {Q_0}, {Q_4}, {Q_5}, {Q_2}, {Q_3}, {Q_1}, {Q_6}

The DFA is already minimum-state

2. (8 points) Write a grammar equivalent to the grammar G=({ S , X , Y , W , Z } , { a , b , ; , (,) , *} , P , S} (where the set of productions P is shown below), but without useless symbols.

S → (X) | (Y) | a
X → X ; S | S
Y → Z ; X | (* Z ; W *)
W → (a ; W) | b
Z → a Y | b Z

symbols that generate a non-empty language:

a, b, ;, (,), *, S, W, X

⇒ Y and Z generate an empty language

⇒ S → (X) | a
X → X ; S | S
W → (a ; W) | b

symbols that are reachable from S:

S, X

⇒ W is not reachable

⇒ S → (X) | a
X → X ; S | S

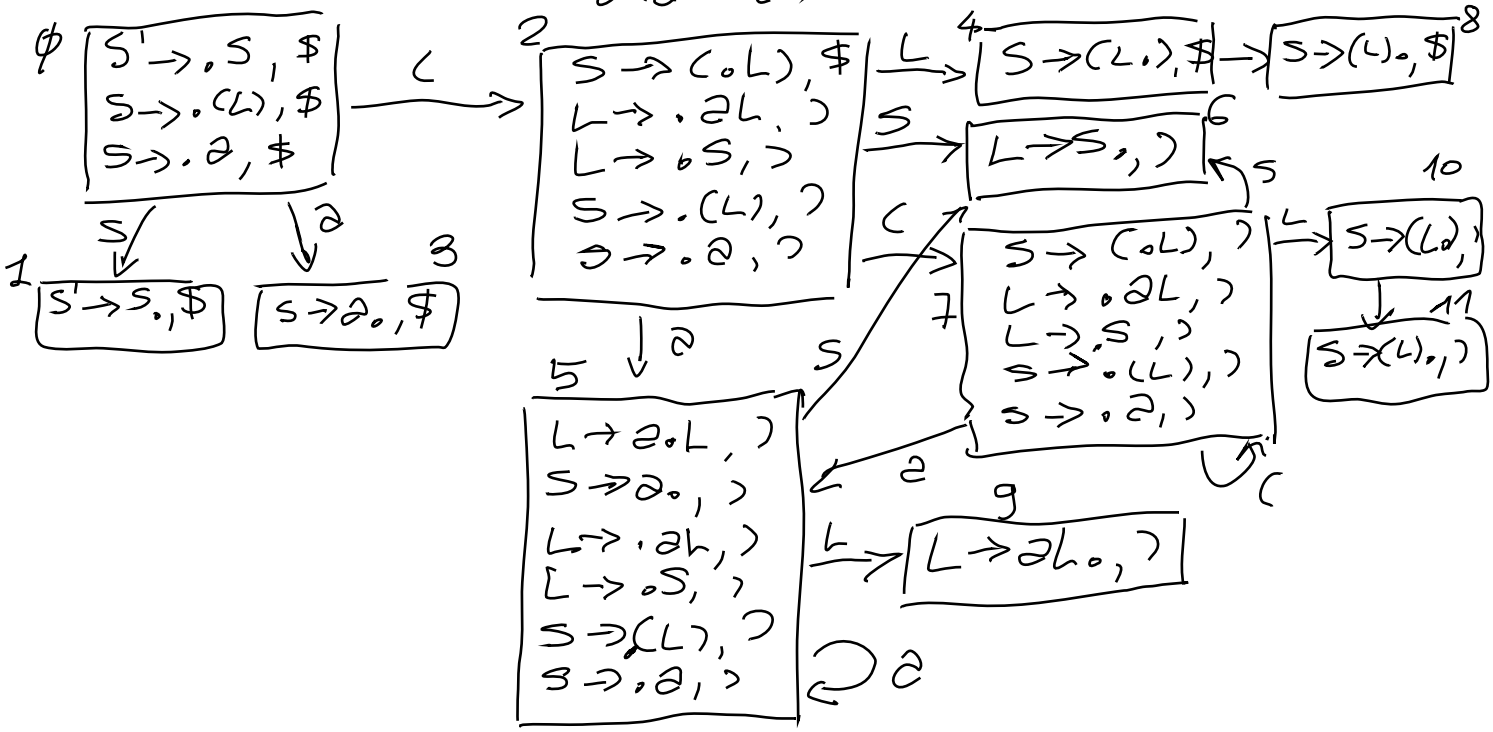
3. (9 points) Given the following grammar, whose set of terminal symbols is { a , (,) } and whose start symbol is S, find the LR(1) parsing table for it.

S → (L) | a

L → a L | S

Is the grammar LR(1)? Looking at the LR(1) parsing table, can we say if the grammar is also LR(0)? Motivate your answers.

DFA construction: we add the rule S'->S (r0) and we number the other rules 1-4



LR(1) parsing table:

	a	(	)	$	S	L
0	s3	s2			1	
1				acc		
2	s5	s7			6	4
3				r2		
4			s8			
5	s5	s7	r2		6	9
6			r4			
7	s5	s7			6	10
8				r1		
9			r3			
10			s11			
11			r1			

The grammar is LR(1) because the LR(1) parsing table has no conflict

The grammar is not LR(0) because in the LR(0) parsing table there would be conflicts, as reduction would be extended on the entire row

4. (6 points) Is it possible that a language generated by a type-2 grammar is also generated by another type-1 grammar? Explain why.

Yes, it is possible because any type-2 language, generated by a type-2 grammar, is also a type-1 language, hence it can be generated by a type-1 grammar.