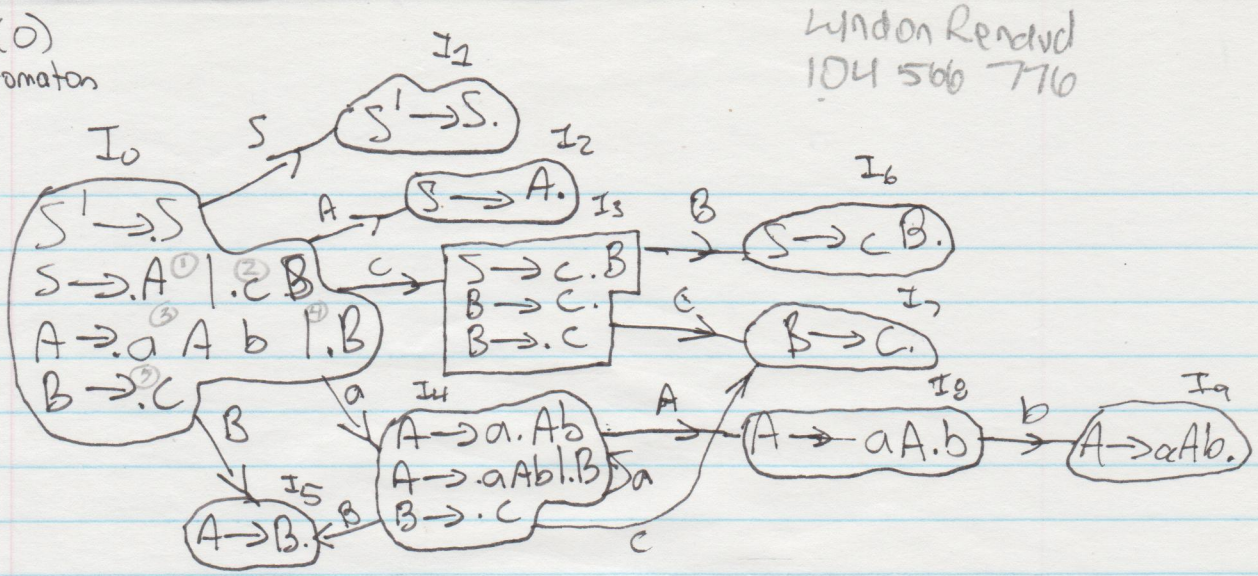
LR(0)
Automaton

Lyndon Rendvd
104 566 7716

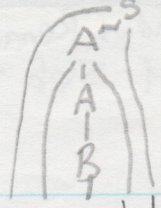


LR(0) Table

State	ACTION				GOTO		
	a	b	c	$	S	A	B
0	s_4		s_3		1	2	5
1				acc			
2	r_1	r_1	r_1	r_1			
3	r_5	r_5	s_7	r_5			6
4	s_4		s_7			8	5
5	r_4	r_4	r_4	r_4			
6	r_2	r_2	r_2	r_2			
7	r_5	r_5	r_5	r_5			
8		s_4					
9	r_3	r_3	r_3	r_3			

Hilroy

Lyndon Rendod
10456676



LR(0) Parse Trace

adcbb

1)

Stack	input	action
0	adcbb$	S4
0a4	acbb$	S4
0a4a4	cbb$	S7
0a4a4c7	bb$	r5
0a4a4B	bb$	5
0a4a4B5	bb$	r4
0a4a4A	bb$	8
0a4a4A8	bb$	S9
0a4a4A8b9	b$	r3
0a4A	b$	8
0a4A8	b$	S9
0a4A8b9	$	r3
0A	$	2
0A2	$	r1
0S	$	1
0S1	$	acc

Lyndon Renaud
104 566 7776

First and Follow

$S' \to A \mid cB$ $\quad first(S') = \{a, c\}$

① $S \to A$ $\quad first(S) = first(A) \cup \{c\} = \{a, c\}$

② $S \to cB$ $\quad first(A) = \{a\} \cup first(B) = \{a, c\}$

③ $A \to aAb$ $\quad first(B) = \{c\}$

④ $A \to B$ $\quad follow(S') = \{\$\}$

⑤ $B \to c$ $\quad follow(S) = \{\$\}$

$follow(A) = \{b\} \cup follow(S) = \{\$, b\}$

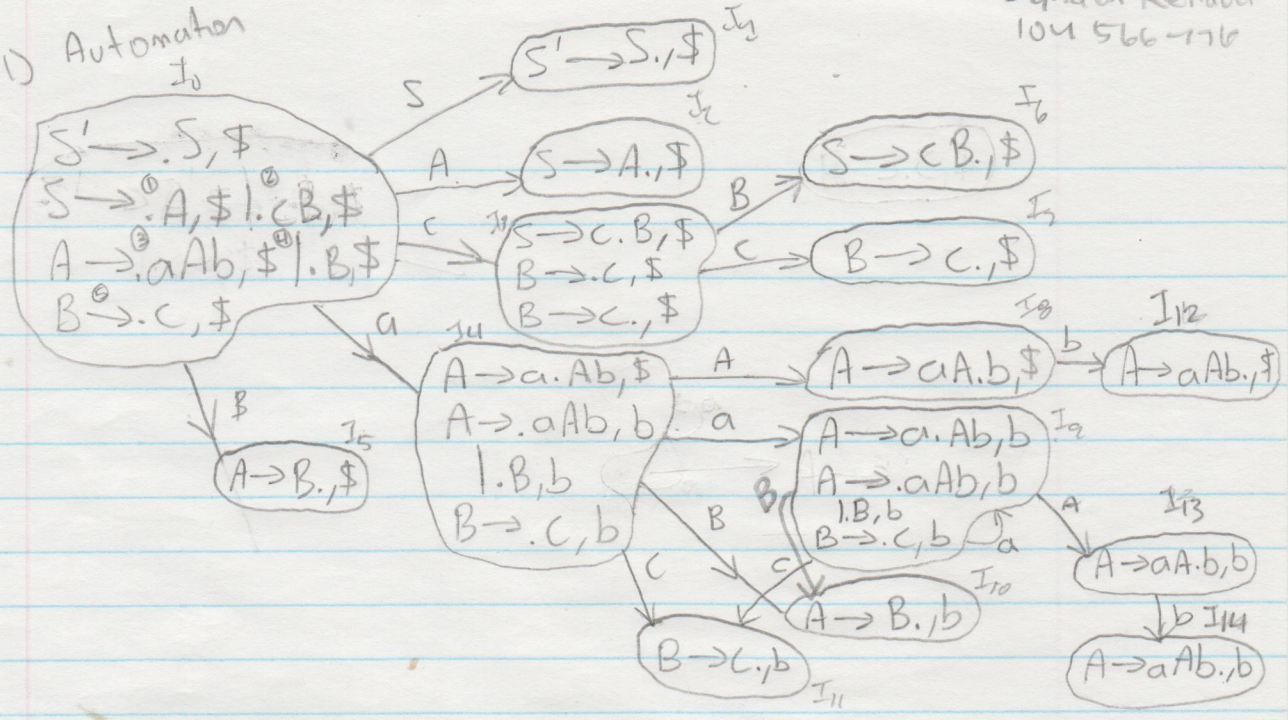
$follow(B) = follow(S) \cup \{b\} = \{\$, b\}$

SLR(1) Table

	ACTION				GOTO		
State	a	b	c	$	S	A	B
0	S_4		S_3		1	2	5
1				acc			
2				r_1			
3		r_5	S_7	r_5			6
4	S_4		S_7			8	5
5		r_4		r_4			
6				r_2			
7		r_5		r_5			
8		S_9					
9		r_3		r_3			

Lyndon Renaud
104 566-176

LR(1) Automaton



CLR(1) Table

State	ACTION a	b	c	$	GOTO S	A	B
0	S_4		S_3		1	2	5
1				acc			
2				r_1			
3			S_7	r_5		6	
4	S_9		S_{11}			8	10
5				r_4			
6				r_2			
7				r_5			
8		S_{12}					
9	S_9		S_{11}			13	10
10		r_4					
11		r_5					
12				r_3			
13		S_{14}					
14		r_3					

12,14
5,10
7,11
4,9
8,13

LALR(1) Table

State	ACTION				GOTO		
	a	b	c	$	S	A	B
0	S_{49}		S_3		1	2	510
1				acc			
2				r_1			
3			S_{711}				6
49	S_{49}		S_{711}			813	510
510		r_4		r_4			
6				r_2			
711		r_5		r_5			
813		S_{1214}					
1214		r_3		r_3			

2) The grammar is not LR(0) because of shift reduce conflict in state 3

I_3: S → C.B Goto
B → C. reduce
B → .C shift

The grammar is SLR(1) because the state 3 conflict is resolved by only reducing when follow is b or $

I_3: S → C.B, $ Goto
B → C., $/b reduce
B → .C, C shift

The grammar is CLR(1) because there are no shift reduce or reduce reduce conflicts

the grammar is LALR(1) because there are no reduce reduce conflicts after merging