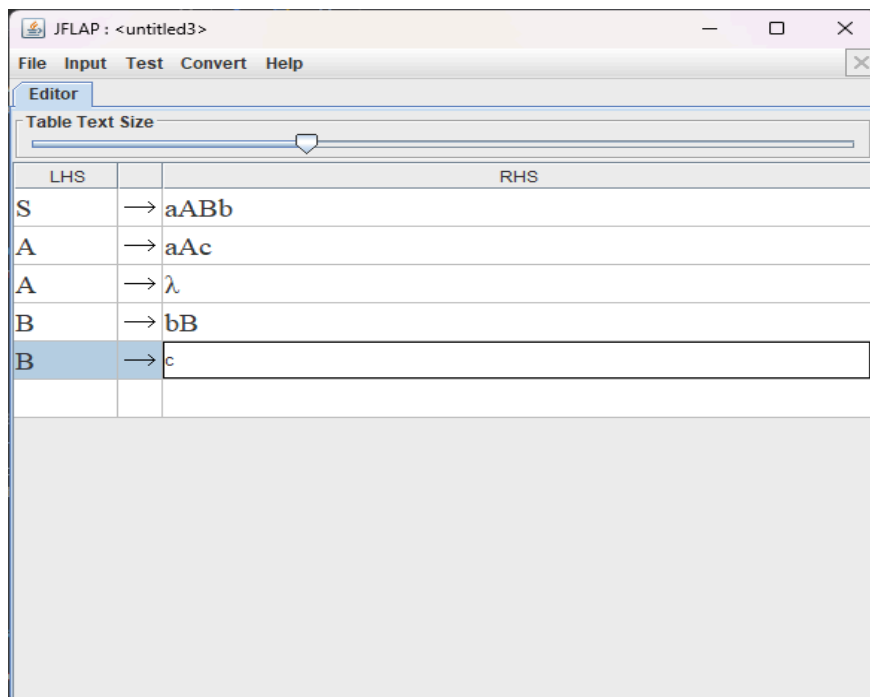**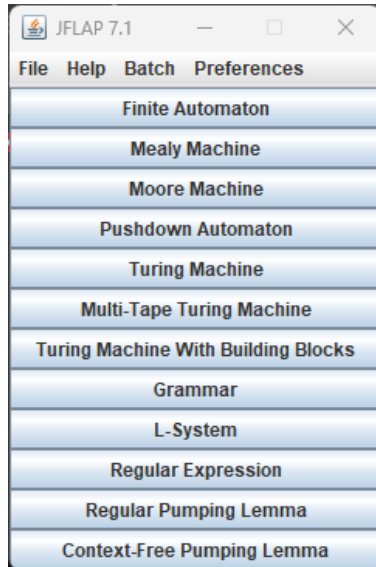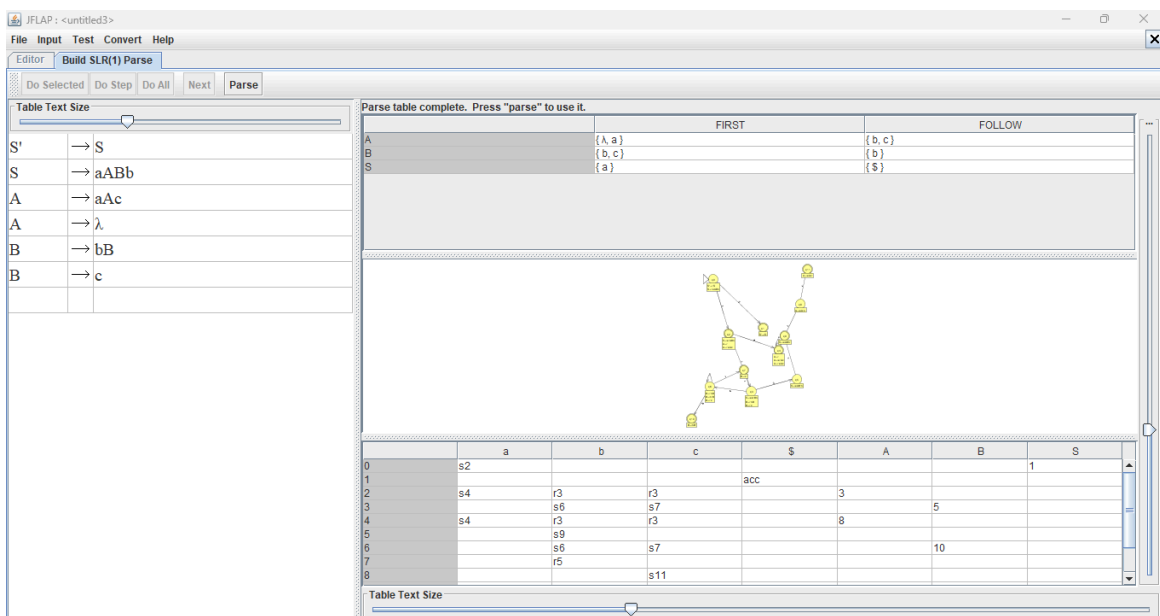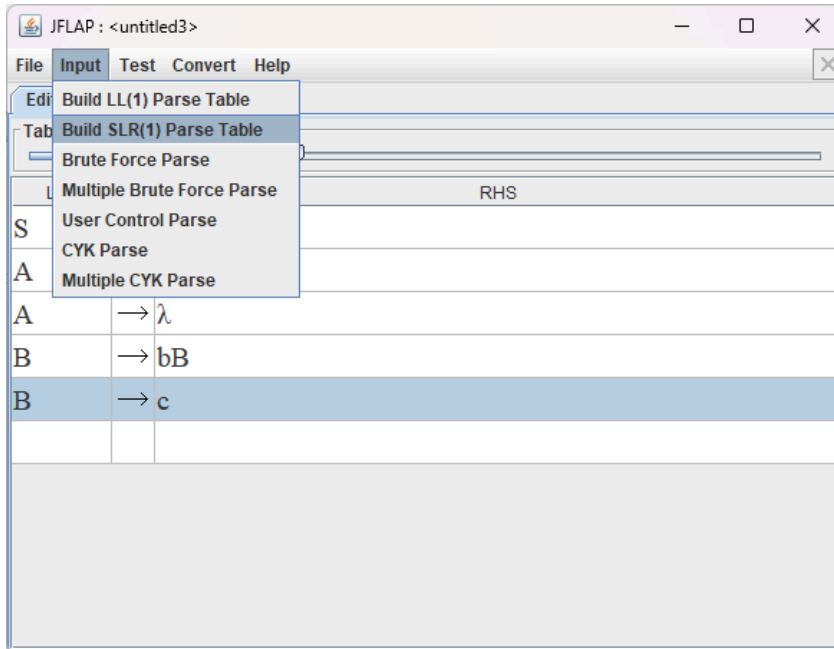Lab 14: Using JFLAP create SLR(1) parse table for given grammar. Simulate parsing and output the parse tree pepper format.**

Menu (Input):
- Build LL(1) Parse Table
- Build SLR(1) Parse Table
- Brute Force Parse
- Multiple Brute Force Parse
- User Control Parse
- CYK Parse
- Multiple CYK Parse

Grammar table:

LHS		RHS
S		
A		
A	→	λ
B	→	bB
B	→	c

Editor | Build SLR(1) Parse

Do Selected | Do Step | Do All | Next | Parse

Table Text Size

LHS		RHS
S'	→	S
S	→	aABb
A	→	aAc
A	→	λ
B	→	bB
B	→	c

Parse table complete. Press "parse" to use it.

	FIRST	FOLLOW
A	{ λ, a }	{ b, c }
B	{ b, c }	{ b }
S	{ a }	{ $ }

	a	b	c	$	A	B	S
0	s2						1
1				acc			
2	s4	r3	r3		3		
3		s6	s7			5	
4	s4	r3	r3		8		
5		s9					
6		s6	s7			10	
7		r5					
8		s11					

Table Text Size

File   Input   Test   Convert   Help

Editor | Build SLR(1) Parse | SLR(1) Parsing

Table Text Size

	a	b	c	$	A	B	S
0	s2						1
1				acc			
2	s4	r3	r3	3			
3		s6	s7		5		
4	s4	r3	r3	8			
5		s9					
6		s6	s7		10		
7		r5					

Start Step   Noninverted Tree

Input   abcb
Input Remaining $
Stack   S0

Input Field Text Size (For optimization, move one of the window size adjustors around this window after resizing the text fields)

Table Text Size

LHS		RHS
S'	⟶	S
S	⟶	aABb
A	⟶	aAc
A	⟶	λ
B	⟶	bB
B	⟶	c

String accepted