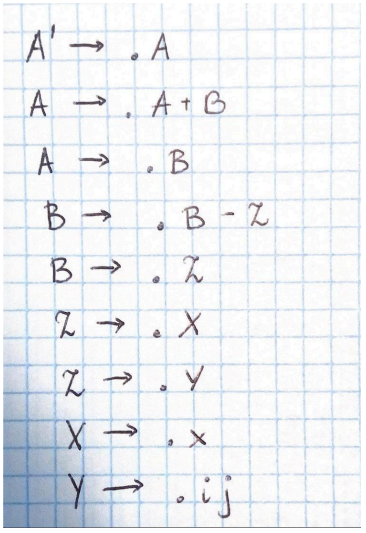
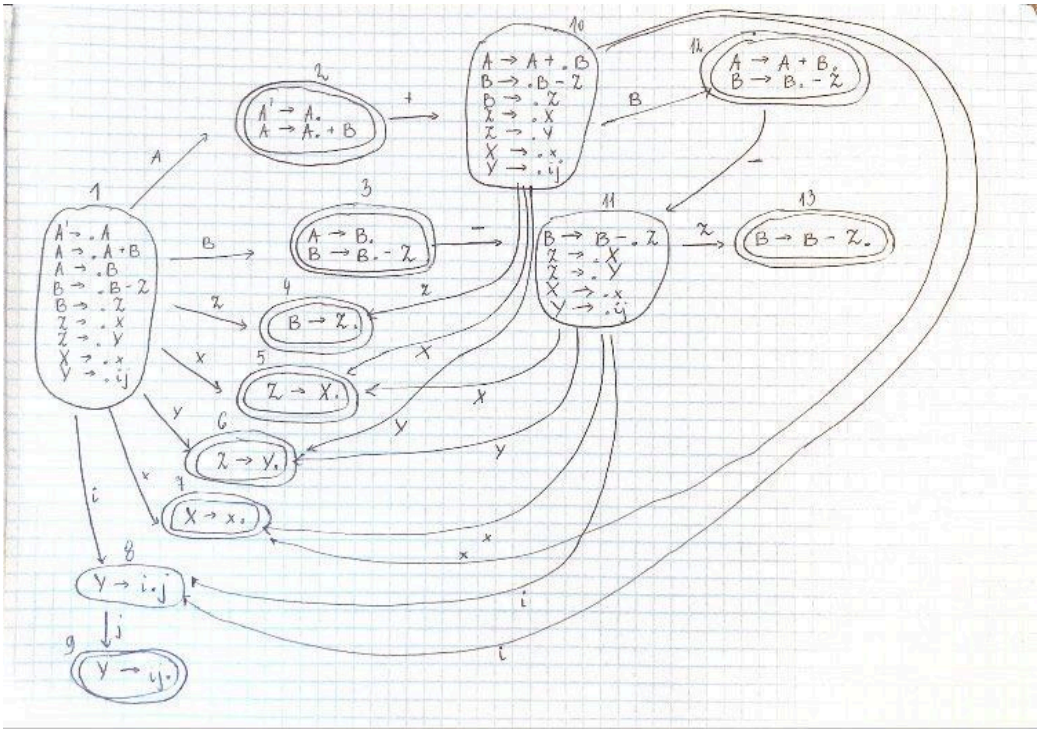
Compiler Construction
Problem Set 3
Alicja Jonczyk

The given grammar:
A → A + B | B
B → B − Z | Z
Z → X | Y
X → x
Y → i j

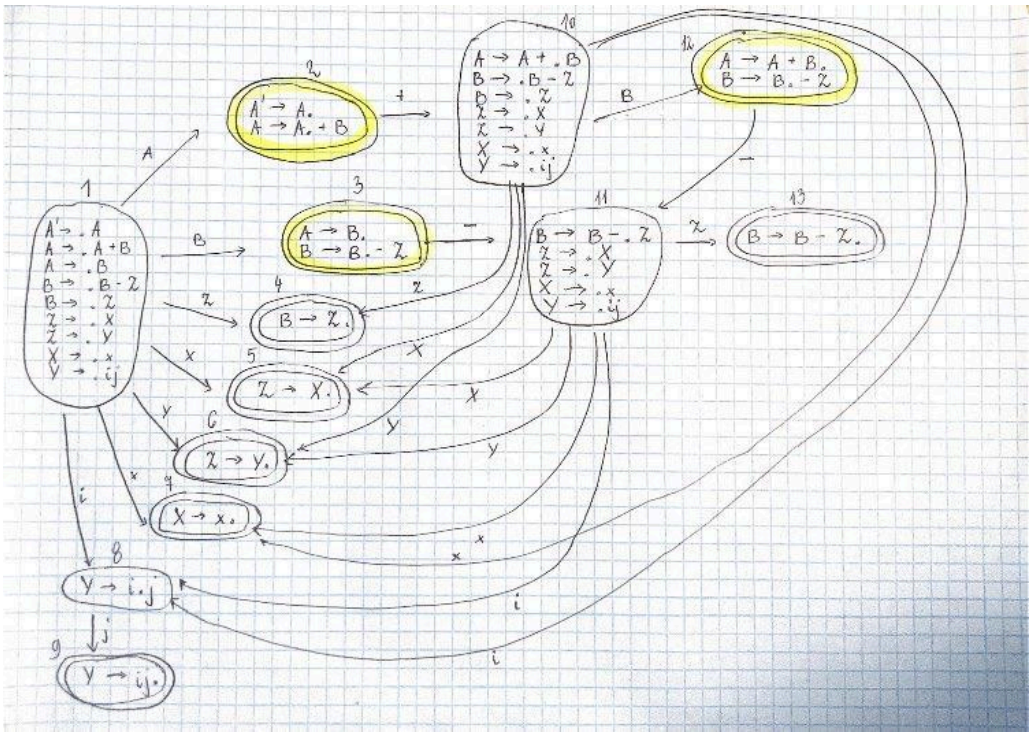
1.1
Augment the grammar, and construct its LR(0) automaton





1.2

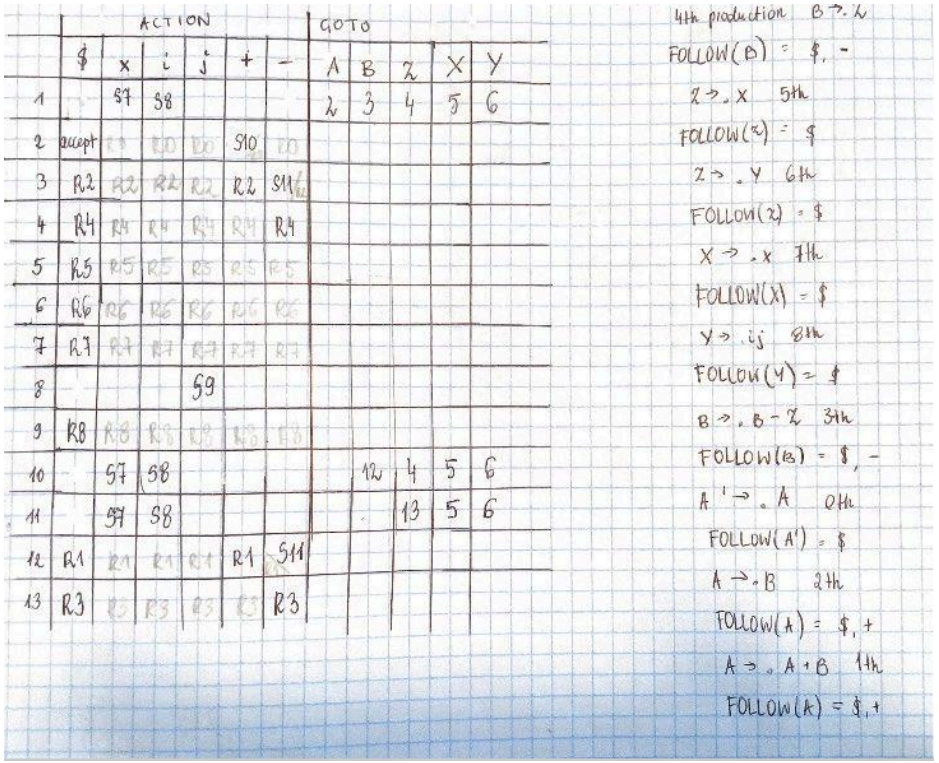
Identify any states with shift/reduce conflicts in your automaton.



1.3

Is the grammar SLR parseable? Justify your answer by constructing the SLR parsing table, and show any potential conflicts.

Before removing the shift - reduce conflicts the grammar was not parseable, but it became parsable after removing the conflicts.



	ACTION						GOTO				
	$	x	i	j	+	-	A	B	Z	X	Y
1		S7	S8				2	3	4	5	6
2	accept				S10						
3	R2	R2	R2	R2	R2	S11					
4	R4	R4	R4	R4	R4	R4					
5	R5	R5	R5	R5	R5	R5					
6	R6	R6	R6	R6	R6	R6					
7	R7	R7	R7	R7	R7	R7					
8				S9							
9	R8	R8	R8	R8	R8	R8					
10		S7	S8					12	4	5	6
11		S7	S8						13	5	6
12	R1	R1	R1	R1	R1	S11					
13	R3	R3	R3	R3	R3	R3					

4th production B → . Z

FOLLOW(B) = $, -

Z → . X 5th

FOLLOW(Z) = $

Z → . Y 6th

FOLLOW(Z) = $

X → . x 7th

FOLLOW(X) = $

Y → . ij 8th

FOLLOW(Y) = $

B → . B - Z 3th

FOLLOW(B) = $, -

A' → . A 0th

FOLLOW(A') = $

A → . B 2th

FOLLOW(A) = $, +

A → . A + B 1th

FOLLOW(A) = $, +

2.

```
jonczyk@alicja:~/COMPILER_CONSTRUCTION/problem_set_3/vsl_programs$ make ps3-check
../build/vslc -T < ps3-simplify/arrays.vsl > ps3-simplify/arrays.ast
../build/vslc -T < ps3-simplify/constant_folding.vsl > ps3-simplify/constant_folding.ast
../build/vslc -T < ps3-simplify/peephole.vsl > ps3-simplify/peephole.ast
cd ps3-simplify; \
find * -wholename "suggested/*.ast" | awk -F/ '{print $0 " " $2}' | xargs -L 1 diff -s --unified=0
Files suggested/arrays.ast and arrays.ast are identical
Files suggested/constant_folding.ast and constant_folding.ast are identical
Files suggested/peephole.ast and peephole.ast are identical
No differences found in PS3!
jonczyk@alicja:~/COMPILER_CONSTRUCTION/problem_set_3/vsl_programs$
```