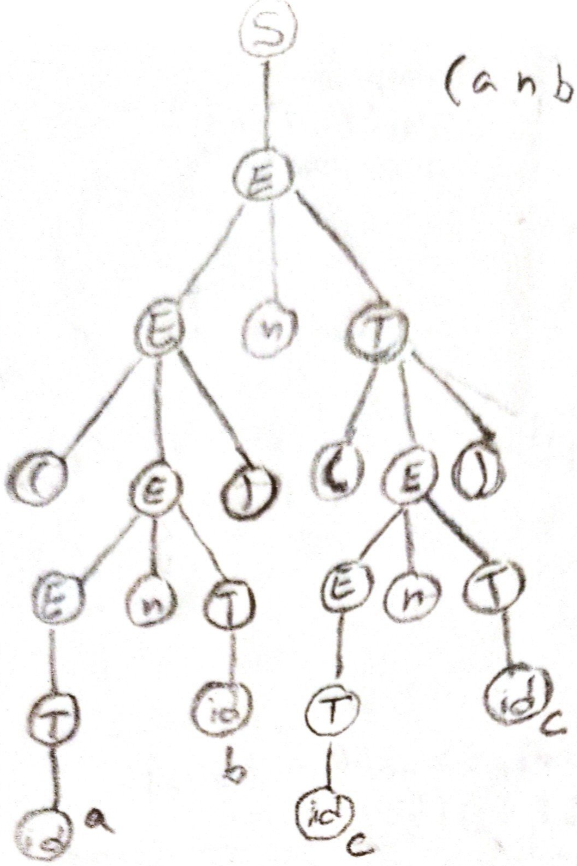
TP 4

1 a)



(a n b) n (c n c)

b) Left-recursion em $E \rightarrow E\,n\,T$

c)

$S \rightarrow E\$$

$E \rightarrow TE'$

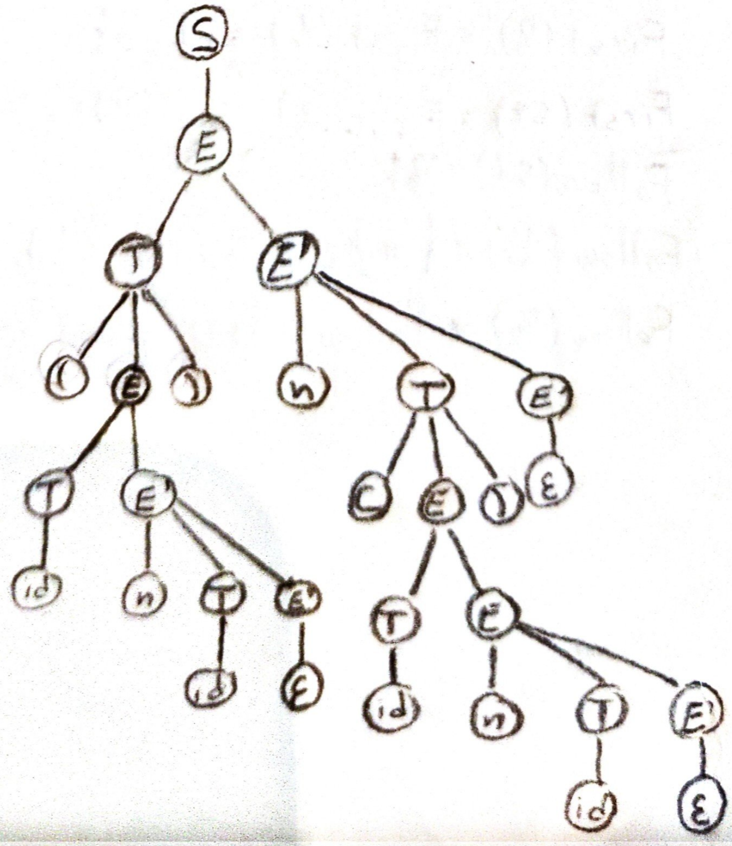
$E' \rightarrow nTE'$

$\quad | \; \varepsilon$

$T \rightarrow (E)$

$\quad | \; id$

d)



e)

$S \to E\$$

$E \to TE'$

$E' \to nTE'$ {print(n)}

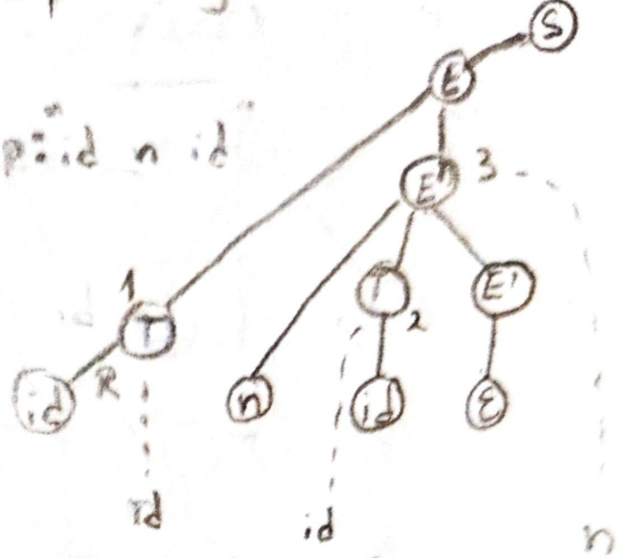
$E' \to \varepsilon$

$T \to (E)$

$T \to id$ {print(id)}

! Assumindo shift-reduce parsing

inp: "id n id"



output: id id n

(Pseudo-código)

f) $S \to E\$$ { \$\$ = new(Program); \$\$.add(new(\$1)) }

$E \to TE'$ { \$\$.add(new(\$1, \$2)) }

$E' \to nTE'$ { \$\$.add(new(\$1, \$2, \$3)) }

$E' \to \varepsilon$ { }

$T \to (E)$ { \$\$.add(new(\$2)) }

$T \to id$ { \$\$.add(new(\$1, \$1.token)) }

(Também dava para fazer sem nodes do tipo Term e Expression)

in: id n id

out: Program

    Expression
        Nand
        Term
            id
                token
        Term
            id
                token

3

0	$S \to St\$$
1	$St \to L = R$
2	$St \to R$
3	$L \to !R$
4	$L \to id$
5	$R \to L$

a) First(L) = {!, id}

First(R) = First(L) = {!, id}

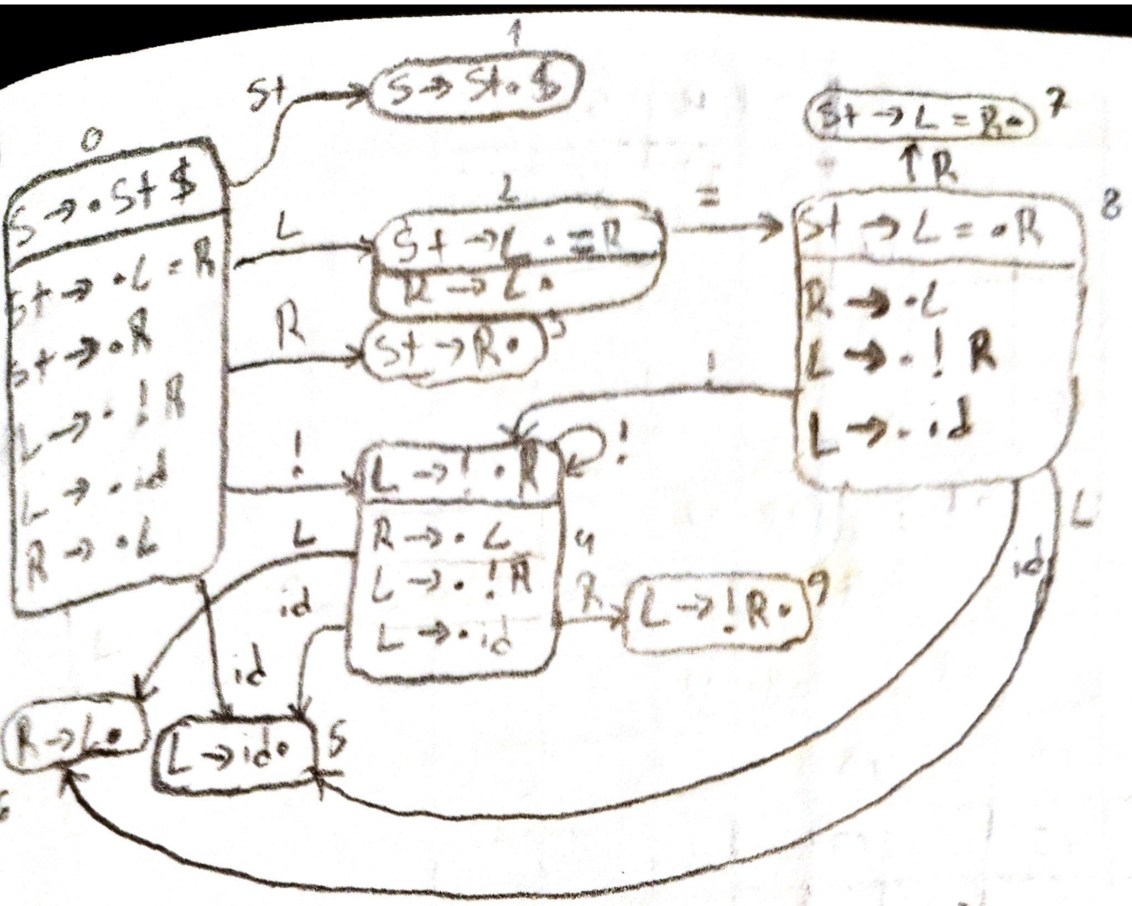
First(St) = First(R) ∪ First(R) = {!, id}

Follow(St) = {\$}

Follow(L) = {=} ∪ Follow(R) = {\$, =}

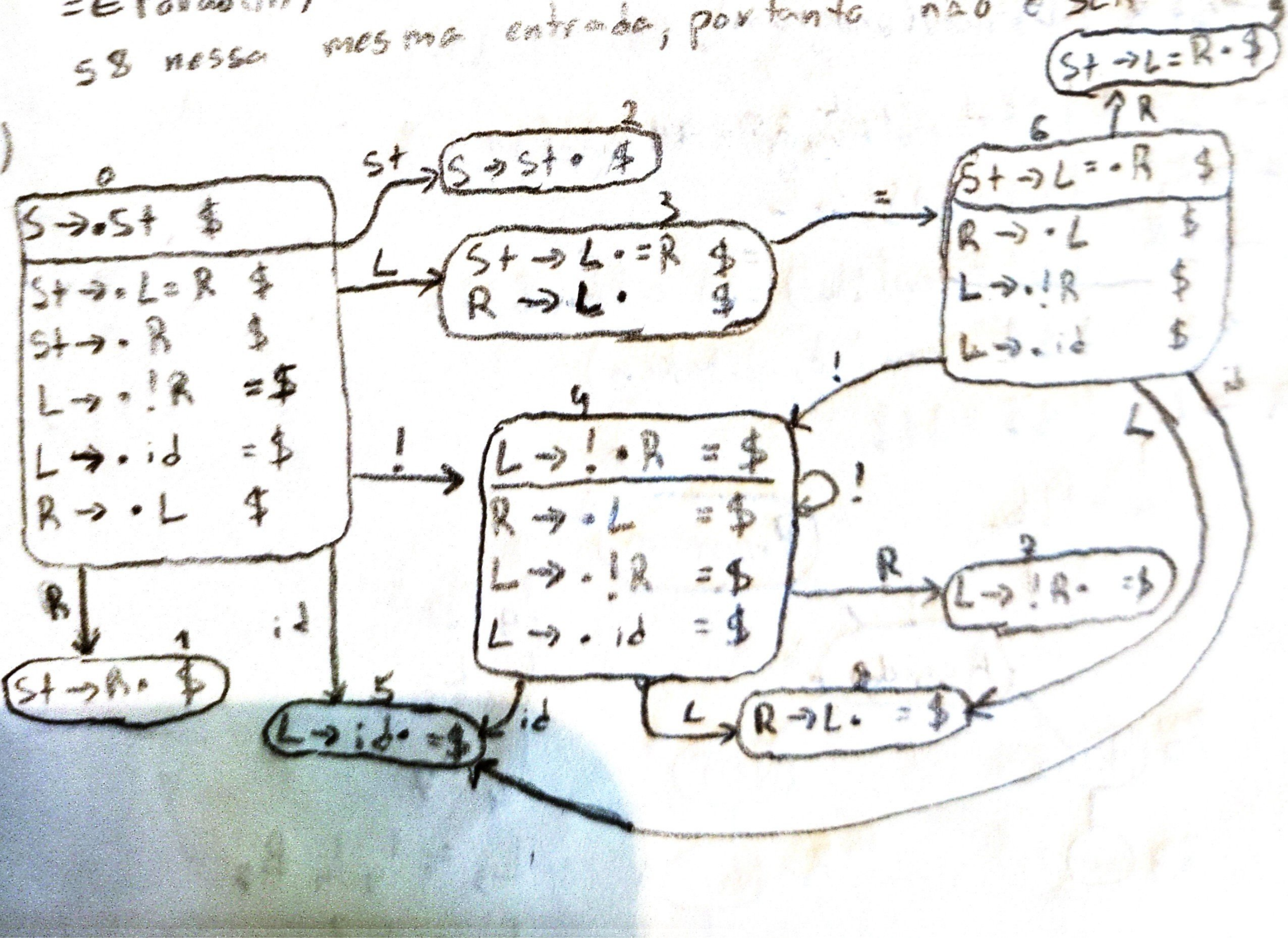
Follow(R) = Follow(St) ∪ Follow(L) = {\$, =}

b)



c) Se for construir a tabela, obviamente vou ter um shift-reduce conflict em T[2, =] porque =∈ Follow(R), mas também há uma instrução de s8 nessa mesma entrada, portanto não é SLR

d)



	id	=	!	$	L	R	St	S
0	s5		s4		g3	g1	g2	
1				r2				
2				A				
3		s6		r5				
4	s5		s4		g8	g7		
5		r4		r4				
6	s5		s4		g8	g9		
7		r3		r3				
8		r5		r5				
9				r1				

e) YACC / BISON

0 S → St $ { $$=new(program); $$.add($1)}

1 St → L = R { $$=new(Assign); $$.add($1, $3)}

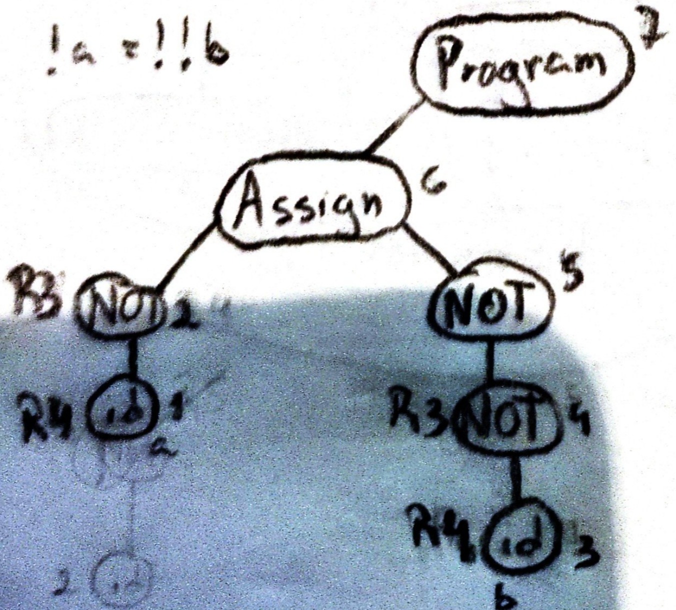
2 St → R { $$ = $1 }

3 L → ! R { $$ = new(Not); $$.add ($2)}

4 L → id { $$ = new(id, id.token)

5 R → L { $$ = $1 }

f) !a = !!b



$!_4 \; a_5$

$!_4 \; L_8$

$!_4 \; R_7$

$L_3 =_6 !_4 !_4 R_7$

$L_3 =_6 !_4 R_7$

4) 0 $S \to St\,\$$

1 $St \to E\ ;$

2 $St \to \{ StL \}$

3 $StL \to StL\ St$

4 $StL \to \varepsilon$

5 $E \to id = E$

6 $E \to id$

$First(E) = \{id\}$

$First(StL) = First(St) \cup \{\varepsilon\} = \{id, \{, \varepsilon\}$

$First(St) = First(E) \cup \{\{\} = \{id, \{\}$

$Follow(St) = \{\$\}$

$Follow(StL) = \{\}\} \cup First(St) = \{\$, \}\}$

$Follow(E) = \{;\}$

$Nullable = \{StL\}$

b, c, d) Mesmo de sempre

e) 0 $S \to St\,\$$

1 $St \to E\ ;$

2 $St \to \{StL\}$

3 $StL \to St\ StL$

4 $StL \to \varepsilon$

5 $E \to id\ E'$

6 $E' \to = E$

7 $E' \to \varepsilon$

$First(E') = \{\varepsilon, =\}$

$First(E) = \{id\}$

$First(St) = \{\{\} \cup First(E) = \{\{, id\}$

$First(StL) = \{\varepsilon\} \cup First(St) = \{\varepsilon, \{, id\}$

$Follow(St) = \{\$\} \cup First(StL) \setminus \varepsilon \cup Follow(StL) = \{\$, \{, id, \}\}$

$Follow(StL) = \{\}\}$

$Follow(E) = \{;\} \cup Follow(E') = \{;\}$

$Follow(E') = Follow(E) = \{;\}$

	;	=	id	{	}	$
S			0	0		
St			1	2		
StL			3	3	4	
E			5			
E'	7	6				