Here is you answer

Please read carefully:

This tool was built to help students test their ability answering the problems, and thus I totally disclaim my responsibility for any unethical use of it.

Question:

The production is known as follows:

A -> B | A - C

B -> C | bDe

C ->f | A + C

D ->(A)

a) Draw a Go To transition diagram

b) Make the SLR table

c) Perform stack implementation for the string: b(f)e-f

Answer:

SLR closure table

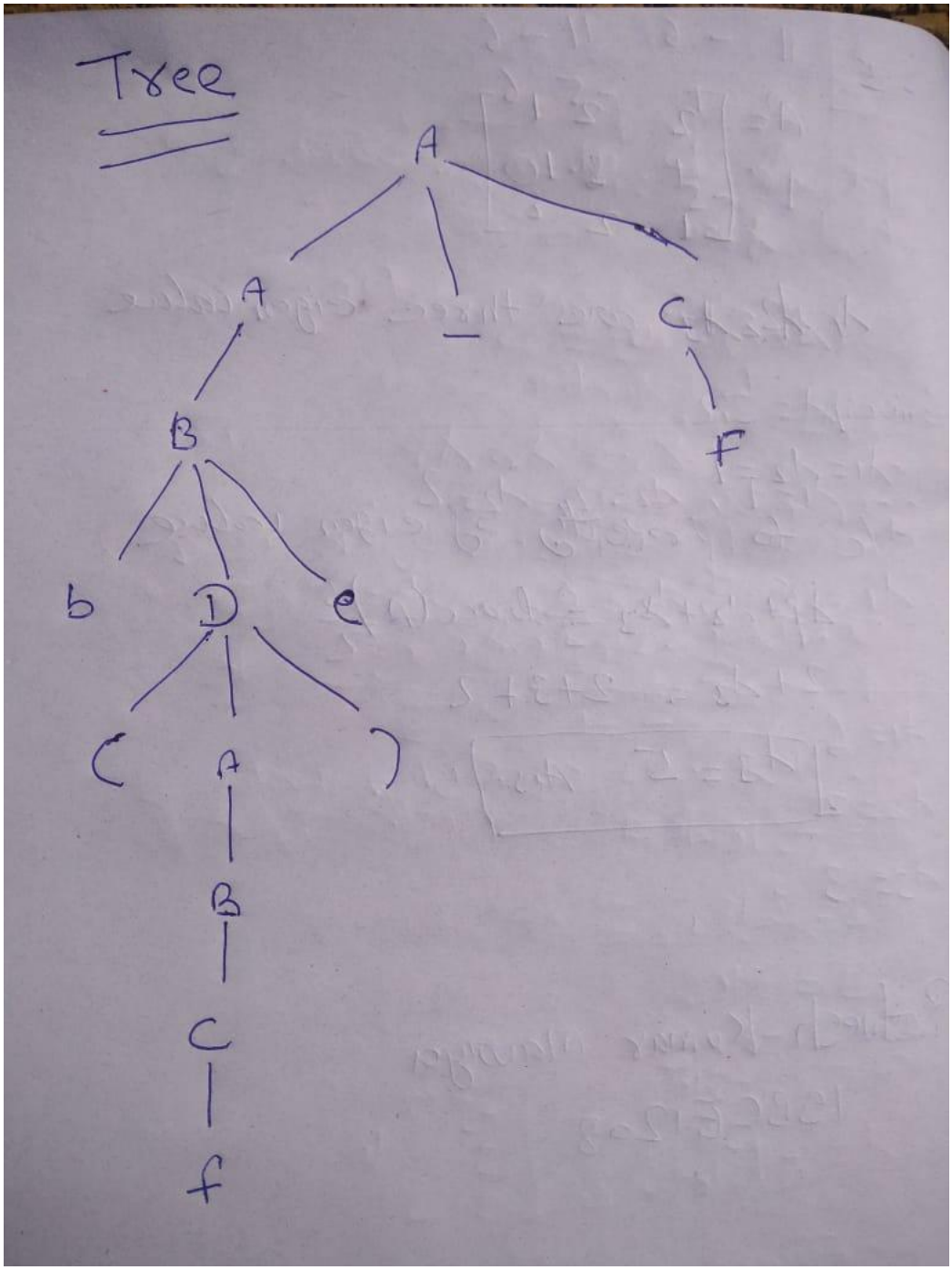
Goto	Kernel	State	Closure
	{A' -> .A}	0	{A' -> .A; A -> .B; A -> .A - C; B -> .C; B -> .b D e; C -> .f; C -> .A + C}
goto(0, A)	{A' -> A.; A -> A.- C; C -> A.+ C}	1	{A' -> A.; A -> A.- C; C -> A.+ C}
goto(0, B)	{A -> B.}	2	{A -> B.}
goto(0, C)	{B -> C.}	3	{B -> C.}
goto(0, b)	{B -> b.D e}	4	{B -> b.D e; D -> .(A)}
goto(0, f)	{C -> f.}	5	{C -> f.}
goto(1, -)	{A -> A -.C}	6	{A -> A -.C; C -> .f; C -> .A + C; A -> .B; A -> .A - C; B -> .C; B -> .b D e
goto(1, +)	{C -> A +.C}	7	{C -> A +.C; C -> .f; C -> .A + C; A -> .B; A -> .A - C; B -> .C; B -> .b D e
goto(4, D)	{B -> b D.e}	8	{B -> b D.e}
goto(4, ()	{D -> (.A)}	9	{D -> (.A); A -> .B; A -> .A - C; B -> .C; B -> .b D e; C -> .f; C -> .A + C
goto(6, C)	{A -> A - C.; B -> C.}	10	{A -> A - C.; B -> C.}
goto(6, f)	{C -> f.}	5	
goto(6, A)	{C -> A.+ C; A -> A.- C}	11	{C -> A.+ C; A -> A.- C}
goto(6, B)	{A -> B.}	2	
goto(6, b)	{B -> b.D e}	4	
goto(7, C)	{C -> A + C.; B -> C.}	12	{C -> A + C.; B -> C.}
goto(7, f)	{C -> f.}	5	
goto(7, A)	{C -> A.+ C; A -> A.- C}	11	
goto(7, B)	{A -> B.}	2	
goto(7, b)	{B -> b.D e}	4	
goto(8, e)	{B -> b D e.}	13	{B -> b D e.}
goto(9, A)	{D -> (A.); A -> A.- C; C -> A.+ C}	14	{D -> (A.); A -> A.- C; C -> A.+ C}
goto(9, B)	{A -> B.}	2	
goto(9, C)	{B -> C.}	3	
goto(9, b)	{B -> b.D e}	4	
goto(9, f)	{C -> f.}	5	
goto(11, +)	{C -> A +.C}	7	
goto(11, -)	{A -> A -.C}	6	
goto(14,))	{D -> (A).}	15	{D -> (A).}
goto(14, -)	{A -> A -.C}	6	
goto(14, +)	{C -> A +.C}	7	

LR table													
State	ACTION								GOTO				
	-	b	e	f	+	(	)	$	A'	A	B	C	D
0		s4		s5						1	2	3	
1	s6				s7			acc					
2	r_1				r_1		r_1	r_1					
3	r_3				r_3		r_3	r_3					
4						s9							8
5	r_5				r_5		r_5	r_5					
6		s4		s5						11	2	10	
7		s4		s5						11	2	12	
8			s13										
9		s4		s5						14	2	3	
10	◉r_2 / ○r_3				◉r_2 / ○r_3		◉r_2 / ○r_3	◉r_2 / ○r_3					
11	s6				s7								
12	◉r_6 / ○r_3				◉r_6 / ○r_3		◉r_6 / ○r_3	◉r_6 / ○r_3					
13	r_4				r_4		r_4	r_4					
14	s6				s7		s15						
15			r_7										

FIRST / FOLLOW table		
Nonterminal	FIRST	FOLLOW
A'	{b,f}	{$}
A	{b,f}	{$,-,+,)}
B	{b,f}	{$,-,+,)}
C	{f,b}	{$,-,+,)}
D	{(}	{e}

Input string: b(f) e - f

Step	Stack	Input	Action
1	0	b (f) e - f $	s4
2	0 b 4	(f) e - f $	s9
3	0 b 4 (9	f) e - f $	s5
4	0 b 4 (9 f 5	) e - f $	r_5
5	0 b 4 (9 C	) e - f $	3
6	0 b 4 (9 C 3	) e - f $	r_3
7	0 b 4 (9 B	) e - f $	2
8	0 b 4 (9 B 2	) e - f $	r_1
9	0 b 4 (9 A	) e - f $	14
10	0 b 4 (9 A 14	) e - f $	s15
11	0 b 4 (9 A 14) 15	e - f $	r_7
12	0 b 4 D	e - f $	8
13	0 b 4 D 8	e - f $	s13
14	0 b 4 D 8 e 13	- f $	r_4
15	0 B	- f $	2
16	0 B 2	- f $	r_1
17	0 A	- f $	1
18	0 A 1	- f $	s6
19	0 A 1 - 6	f $	s5
20	0 A 1 - 6 f 5	$	r_5
21	0 A 1 - 6 C	$	10
22	0 A 1 - 6 C 10	$	r_2
23	0 A	$	1
24	0 A 1	$	acc


Tree
A
A
C
B
f
b
D
e
(
A
)
B
C
f