COMP-3220 | Midas Oden | Book Assignment

1) Grammar:

<S> → <A>a<B>b
<A> → <A>b | b
<B> → a<B> | a

a) baab

<S> → <A>a<B>b
→ ba<B>b
→ baab

<A> → b
<B> → a

∴ baab is valid!

b) bbbab

<S> → <A>a<B>b
→ <A>ba<B>b
→ <A>bba<B>b
→ bbba<B>b

<A> → <A>b
<A> → <A>b
<A> → b

∴ bbbab is not generated by the grammar, so is invalid!

c) bbaaaaaa

<S> → <A>a<B>b
→ <A>ba<B>b ..... <A> → <A>b
→ bba<B>b ..... <A> → b
→ bbaa<B>b ..... <B> → a<B>
→ bbaaa<B>b ..... <B> → a<B>
→ bbaaaa<B>b ..... <B> → a<B>
→ bbaaaaa<B>b ..... <B> → a<B>
→ bbaaaaaa<B>b ..... <B> → a

∴ sentence is invalid

d) bbaab

<S> → <A>a<B>b
→ <A>ba<B>b
→ bba<B>b
→ bbaab

<A> → <A>b
<A> → b
<B> → b

∴ bbaab is derived from the grammar and is therefore valid!

2.) Grammar

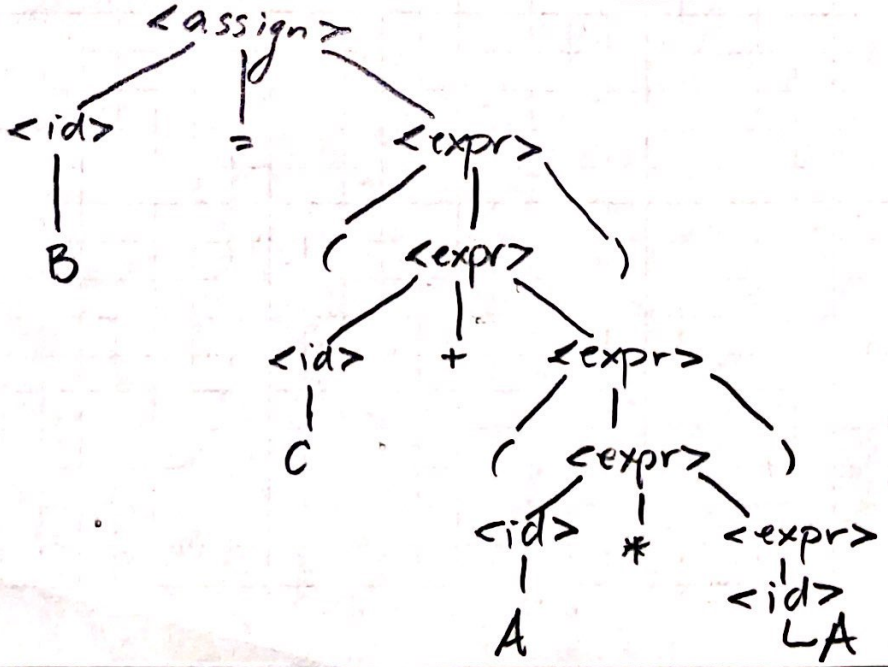
<assign> → <id> = <expr>
<id> → A | B | C
<expr> → <id> + <expr>
| <id> * <expr>
| (<expr>)
| <id>

Tokens	Lexemes
variables	A, B, C
Arith. ops	+, *
Assign. ops	=
Special chars	(,)

3.) Left-most derivation for: B = B + (C + (A * A))

<assign> → <id> = <expr>
→ B = <expr>
→ B = <id> + <expr>
→ B = B + <expr>
→ B = B + (<expr>)
→ B = B + (<id> + <expr>)
→ B = B + (C + <expr>)
→ B = B + (C + (<expr>))
→ B = B + (C + (<id> * <expr>))
→ B = B + (C + (A * <expr>))
→ B = B + (C + (A * <id>))
→ B = B + (C + (A * A)) ✓

Parse Tree



4.) Grammar

$S \to Aa \mid Bb$

$A \to Aa \mid AbC \mid C$

$B \to S \mid bb$

$C \to c$

By removing the left recursion:

$A = A\alpha / B \Rightarrow A = BA'$

$A' = \alpha A' / \varepsilon$

Grammar after removing the recursion:

$S \to Aa \mid Bb$

$A \to CA'$

$A' \to aA' \mid bCA' \mid \varepsilon$

$B \to S \mid bb$

$C \to c$

5.) Grammar

$A \to aBc \mid ac \mid a$

$B \to b \mid aB$

∴ 'a' is common thru out grammar...

$A \to aA'$

$A' \to BC \mid C \mid \varepsilon$

So... using left factoring for grammar results in:

$A \to aA'$

$A' \to Bc \mid C \mid \varepsilon$

$B \to b \mid aB$

6.) Grammar

$E \to E+T \mid E*T \mid T$

$T \to (E) \mid id$

① Closure: $E' \to E$

$E \to E+T$

$E \to E*T$

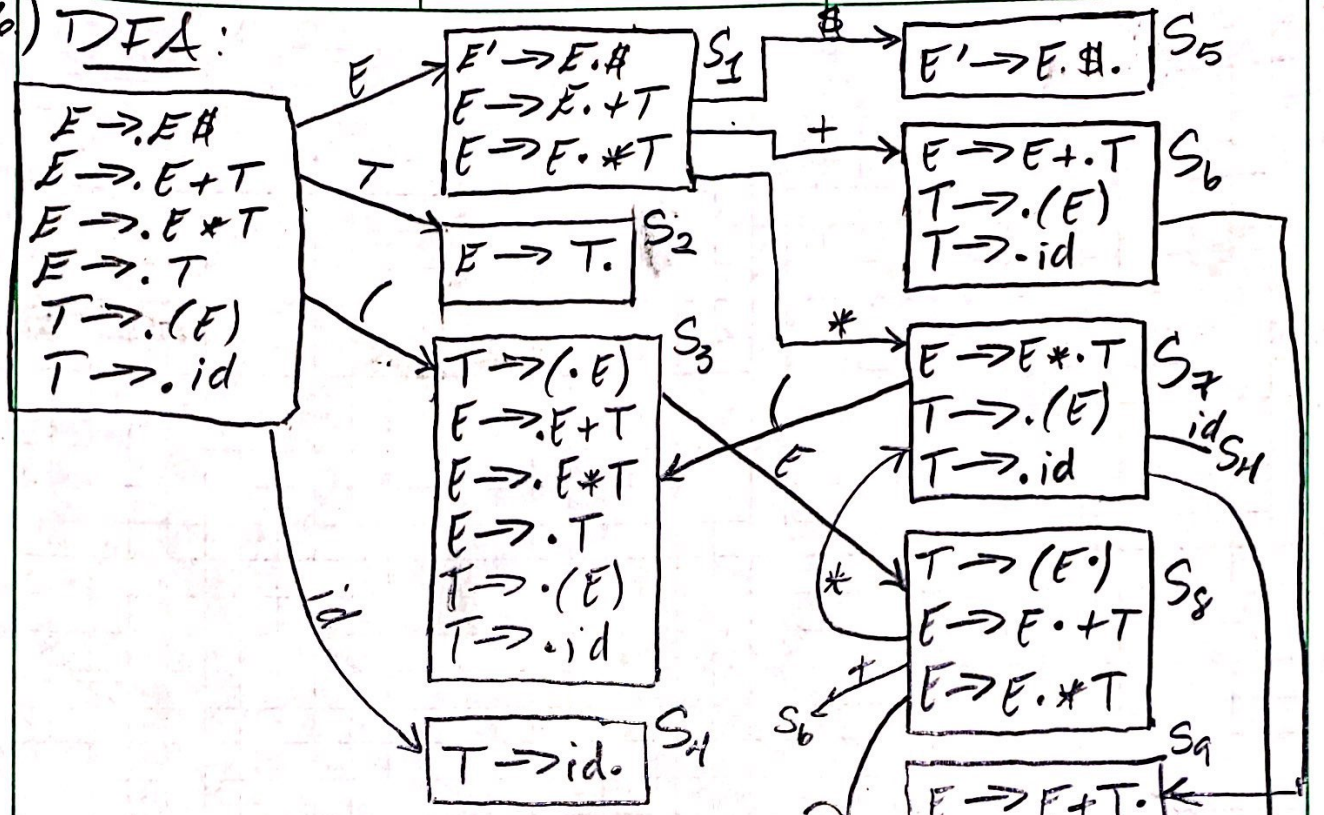
$E \to T$

$T \to (E)$

$T \to id$

(continued on next page)

⇒ 6) DFA:



State	Action						Goto	
	+	*	(	)	id	$	E	T
0			S_3		S_4		1	2
1	S_6	S_7				S_5		
2	R_3	R_3	R_3	R_3	R_3	R_3		
3					S_4			
4	R_5	R_5	R_5	R_5	R_5	R_5		
5						Acc		
6			S_3		S_4			
7			S_3		S_4			
8	S_6	S_7		S_{10}				
9	R_1	R_1	R_1	R_1	R_1	R_1		
10	R_4	R_4	R_4	R_4	R_4	R_4		
11	R_2	R_2	R_2	R_2	R_2	R_2		