

## PROGRAMMING LANGUAGES

### PROBLEM SET - ASSIGNMENT 1 (CHAPTER 3)

Q3. Rewrite the BNF of Example 3.4 to give + precedence over \* and force + to be right associative.

Ans →  $\langle \text{assign} \rangle \rightarrow \langle \text{id} \rangle = \langle \text{expr} \rangle$

$\langle \text{id} \rangle \rightarrow A | B | C$

$\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle^* \langle \text{term} \rangle \langle \text{term} \rangle$

$\langle \text{term} \rangle \rightarrow \langle \text{factor} \rangle + \langle \text{term} \rangle | \langle \text{factor} \rangle$

$\langle \text{factor} \rangle \rightarrow (\langle \text{expr} \rangle) | \langle \text{id} \rangle$

Q4. Rewrite the BNF of example 3.4 to add ++ and -- unary operators of Java.

Ans →  $\langle \text{assign} \rangle \rightarrow \langle \text{id} \rangle = \langle \text{expr} \rangle$

$\langle \text{id} \rangle \rightarrow A | B | C$

$\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle + \langle \text{term} \rangle | \langle \text{term} \rangle$

$\langle \text{term} \rangle \rightarrow \langle \text{term} \rangle^* \langle \text{factor} \rangle | \langle \text{factor} \rangle$

$\langle \text{factor} \rangle \rightarrow (\langle \text{expr} \rangle) | \langle \text{id} \rangle | \langle \text{id} \rangle ++ | \langle \text{id} \rangle --$

Q5. Using the grammars in example 3.2 show a parse tree and a leftmost derivation for each of the following statements.

a)  $A = A^* (B + (C^* A))$

$\langle \text{assign} \rangle \Rightarrow \langle \text{id} \rangle = \langle \text{expr} \rangle$

$\Rightarrow A = \langle \text{expr} \rangle$

$\Rightarrow A = A^* \langle \text{id} \rangle \langle \text{expr} \rangle$

$\Rightarrow A = A^* A \langle \text{expr} \rangle$

$\Rightarrow A = A^* (A^* \langle \text{id} \rangle + \langle \text{expr} \rangle)$

$\Rightarrow A = A^* (B + \langle \text{expr} \rangle)$

$\langle \text{assign} \rangle \Rightarrow A = A^* (B + \langle \text{expr} \rangle)$

$\Rightarrow A = A^* (B + (A^* \langle \text{id} \rangle \langle \text{expr} \rangle))$

$\Rightarrow A = A^* (B + (A^* (A^* \langle \text{id} \rangle)))$

$\Rightarrow A = A^* (B + (A^* A \langle \text{id} \rangle))$

$\Rightarrow A = A^* (B + (A^* A))$

&lt;Assign&gt;

&lt;id&gt; = &lt;expr&gt;

A

&lt;id&gt;

&lt;expr&gt;

A

&lt;expr&gt;

&lt;id&gt;

+

&lt;expr&gt;

B

&lt;expr&gt;

&lt;id&gt;

&lt;expr&gt;

C

&lt;id&gt;

1

(16)  $B = C * (A * B + C + B)$ 

&lt;Assign&gt;

&lt;id&gt;

= &lt;expr&gt;

B

&lt;id&gt;

&lt;expr&gt;

&lt;expr&gt;

&lt;expr&gt;

&lt;expr&gt;

&lt;expr&gt;

C

&lt;expr&gt;

&lt;expr&gt;

A

&lt;expr&gt;

C

&lt;expr&gt;

B

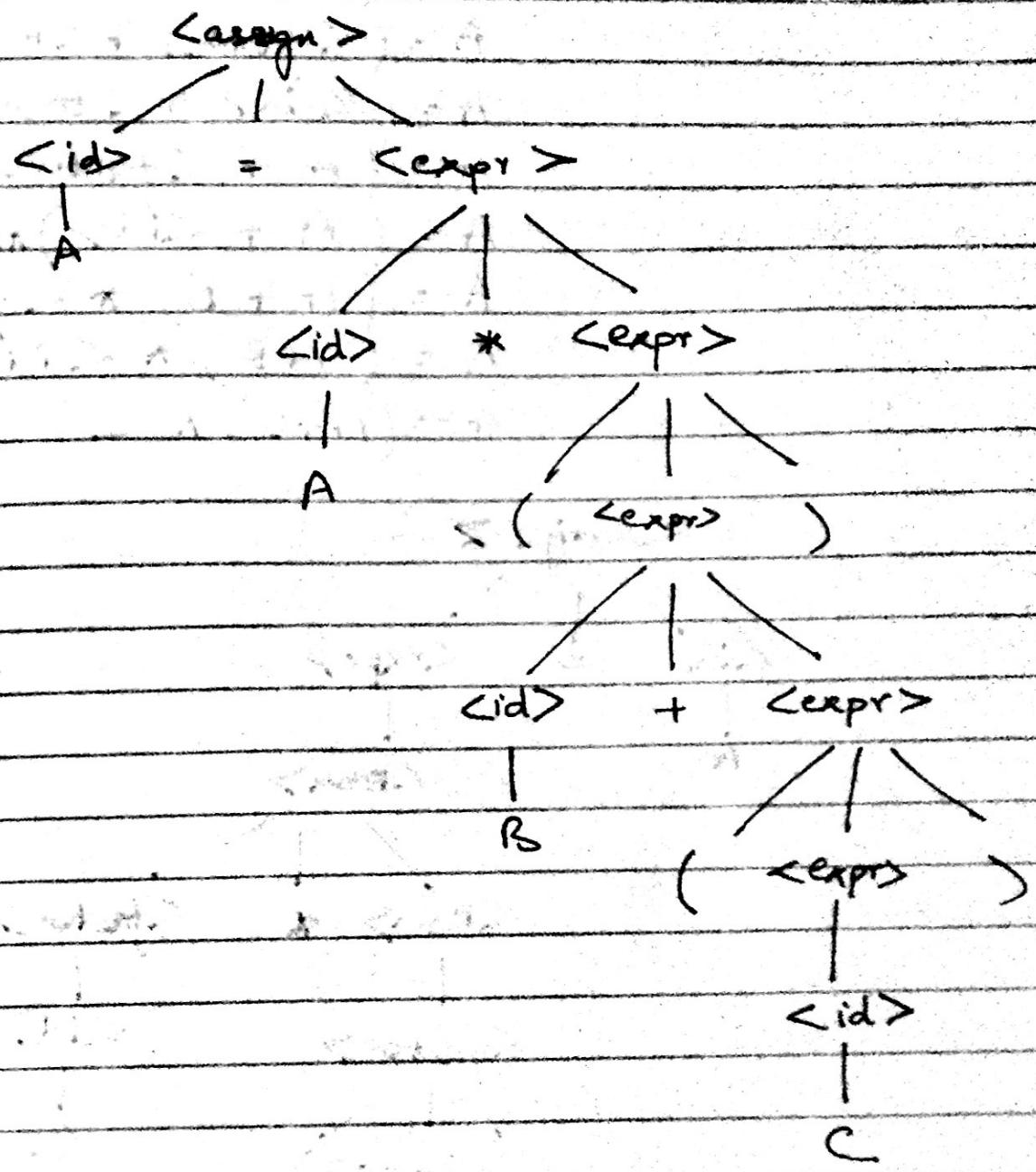
$$(b) B = C * (A * C + S)$$

$$\begin{aligned}\langle \text{Assign} \rangle &\Rightarrow \langle \text{id} \rangle = \langle \text{expr} \rangle \\&\Rightarrow B = \langle \text{expr} \rangle \\&\Rightarrow B = \langle \text{id} \rangle * \langle \text{expr} \rangle \\&\Rightarrow B = C * \langle \text{expr} \rangle \\&\Rightarrow B = C * (\langle \text{id} \rangle * \langle \text{expr} \rangle) \\&\Rightarrow B = C * (A * \langle \text{expr} \rangle) \\&\Rightarrow B = C * (A * \langle \text{id} \rangle + \langle \text{expr} \rangle) \\&\Rightarrow B = C * (A * C + \langle \text{expr} \rangle) \\&\Rightarrow B = C * (A * C + \langle \text{id} \rangle) \\&\Rightarrow B = C * (A * C + B)\end{aligned}$$

$$(c) A = A * (B + (C))$$

$$\begin{aligned}\langle \text{Assign} \rangle &\Rightarrow \langle \text{id} \rangle = \langle \text{expr} \rangle \\&\Rightarrow A = \langle \text{expr} \rangle \\&\Rightarrow A = \langle \text{id} \rangle * \langle \text{expr} \rangle \\&\Rightarrow A = A * \langle \text{expr} \rangle \\&\Rightarrow A = A * (\langle \text{expr} \rangle) \\&\Rightarrow A = A * (\langle \text{id} \rangle + \langle \text{expr} \rangle) \\&= A = A * (B + \langle \text{expr} \rangle) \\&= A = A * (B + (\langle \text{expr} \rangle)) \\&= A = A * (B + (\langle \text{id} \rangle)) \\&= A = A * (B + (C))\end{aligned}$$

(c)  $A = A * (B + C)$

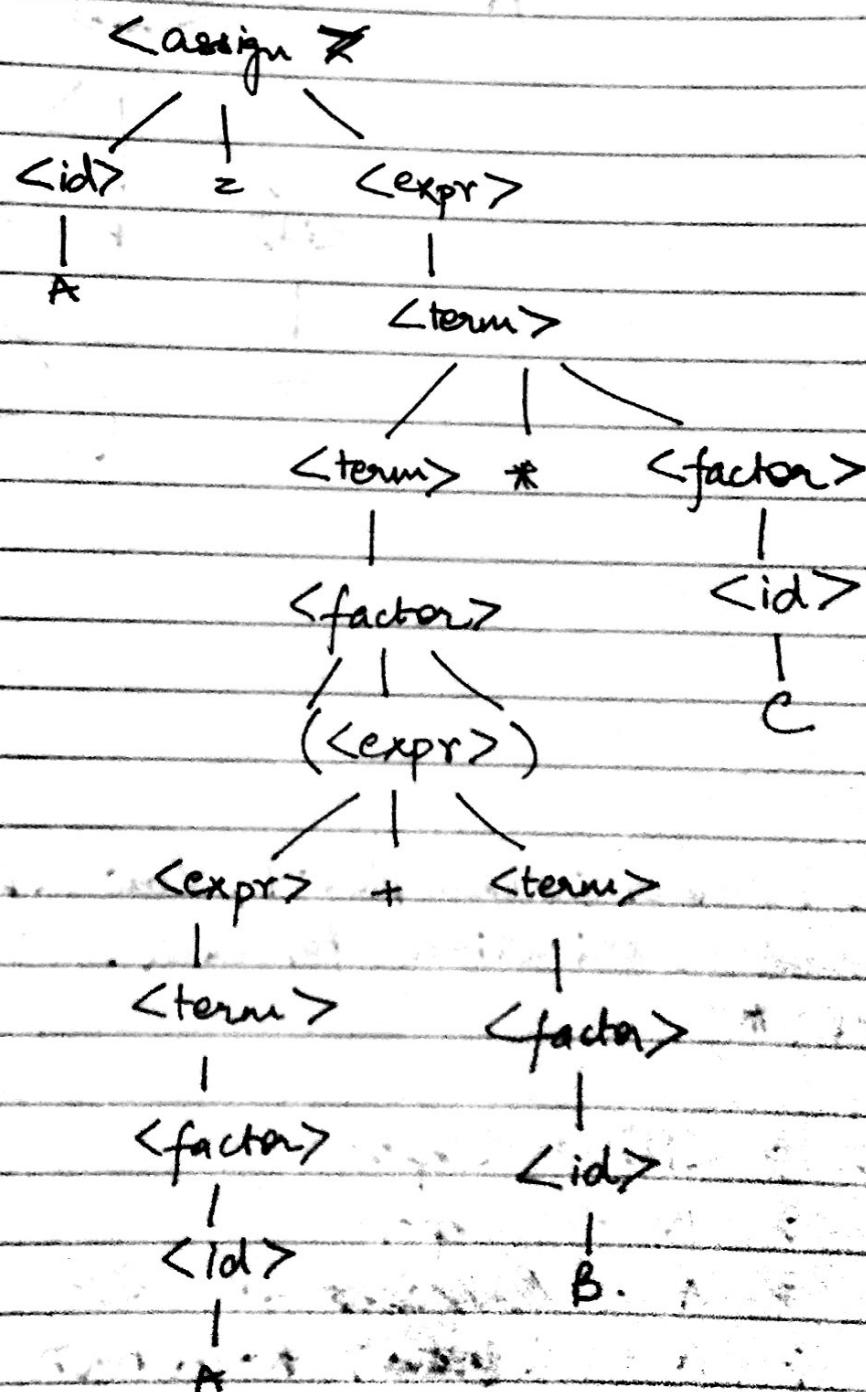


Q7. Using the grammar in example 3.4, show a parse tree and a left most derivation for each.

(a)  $A = (A + B) * C$

$\begin{aligned} <\text{assign}> &\Rightarrow <\text{id}> = <\text{expr}> \\ &\Rightarrow A = <\text{expr}> \\ &\Rightarrow A = <\cancel{\text{term}}> <\text{term}> \\ &\Rightarrow A = <\cancel{\text{term}}> * <\text{factor}> \\ &\Rightarrow A = <\cancel{A}> <\text{expr}> \\ &\Rightarrow A = <\text{factor}> * <\text{factor}> \\ &\Rightarrow A = <\text{expr}> * <\text{factor}> \end{aligned}$

$\langle \text{assign} \rangle \Rightarrow A = (\langle \text{expr} \rangle + \langle \text{term} \rangle) * \langle \text{factor} \rangle$   
 $A = (\langle \text{term} \rangle + \langle \text{term} \rangle) * \langle \text{factor} \rangle$   
 $A = (\langle \text{factor} \rangle + \langle \text{term} \rangle) * \langle \text{factor} \rangle$   
 $A = (\langle \text{id} \rangle + \langle \text{term} \rangle) * \langle \text{factor} \rangle$   
 $A = (A + \langle \text{factor} \rangle) * \langle \text{factor} \rangle$   
 $A = (A + \langle \text{id} \rangle) * \langle \text{factor} \rangle$   
 $A = (A + B) * \langle \text{factor} \rangle$   
 $A = (A + B) * \langle \text{id} \rangle$   
 $A = (A + B) * C$



$$Q7.(b) A = B + C + A$$

`<Assign>  $\Rightarrow$  <id> = <expr>`  
 $\Rightarrow A = <expr>$

$$\Rightarrow A = \langle \exp \rangle$$

$\Rightarrow A = \langle \text{expr} \rangle + \langle \text{term} \rangle$

$\Rightarrow A = \langle \text{expr} \rangle + \langle \text{term} \rangle + \langle \text{term} \rangle$

$\Rightarrow A = \langle \text{term} \rangle + \langle \text{term} \rangle + \langle \text{term} \rangle$

$\Rightarrow A = \langle \text{factor} \rangle + \langle \text{term} \rangle + \langle \text{term} \rangle$

$\Rightarrow A = \langle \text{id} \rangle + \langle \text{term} \rangle + \langle \text{terms} \rangle$

$$\Rightarrow A = B + \langle \text{term} \rangle + \langle \text{term} \rangle$$

$$\Rightarrow A = B + \cancel{\langle \text{fix} \rangle} + \langle \text{term} \rangle$$

$$\Rightarrow A = B + \langle \text{id} \rangle + \langle \text{term} \rangle$$

$$\Rightarrow A = B + C + \langle \text{term} \rangle$$

$$\Rightarrow A = B + C + \langle \text{factor} \rangle$$

$$\Rightarrow A = B + C + \langle 1d \rangle$$

$$\Rightarrow A = B + C + A$$

Assign >

*id* = *expr*

A  $\xrightarrow{\text{lower}}$  + <term>

$\begin{array}{c} \text{<expr>} \\ | \\ \text{<expr>} \quad + \quad \text{<term>} \quad | \quad \text{<factor>} \end{array}$

1 1 <id>  
<term> <factor>

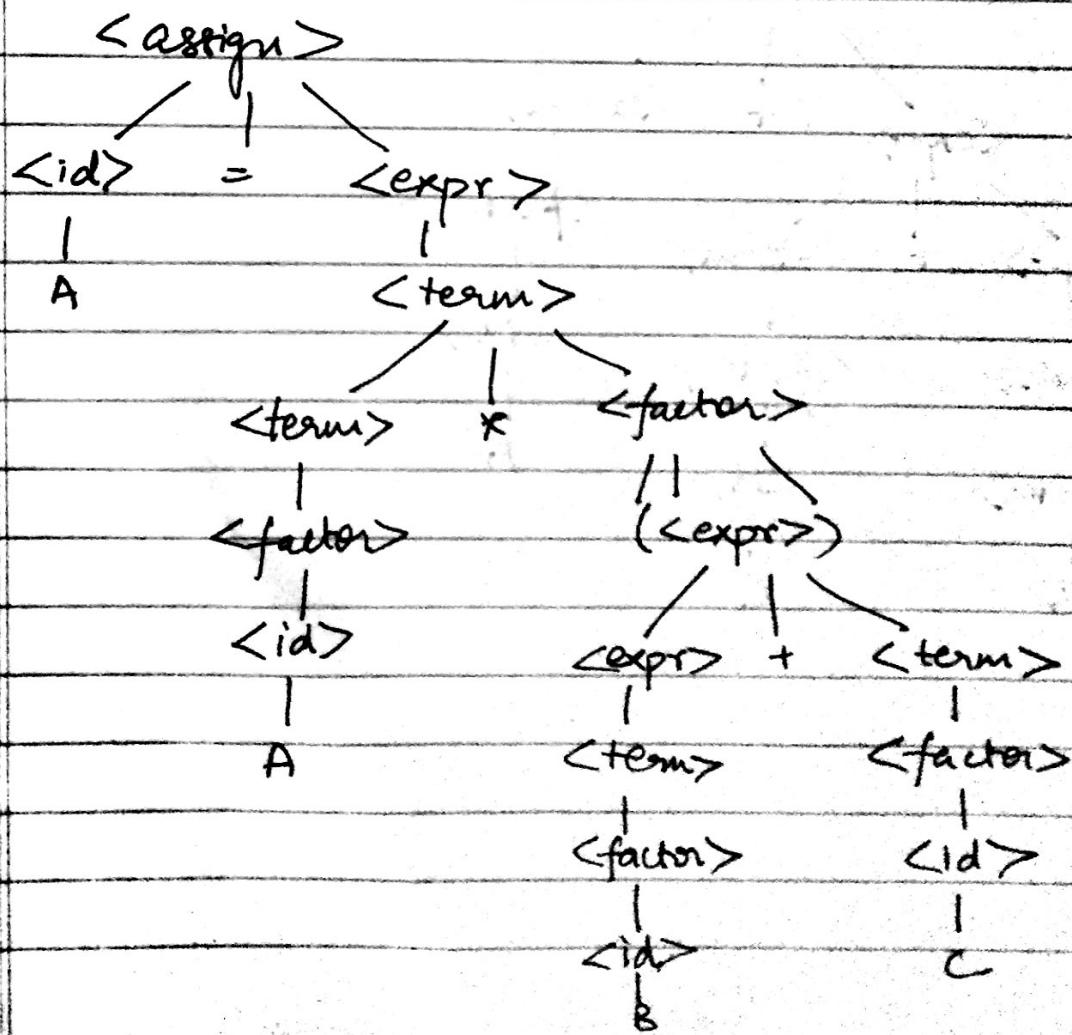
<factor> <id>

<id>      c

1  
b

$$(C) A = A * (B + C)$$

$$\begin{aligned}
 <\text{assign}> &\Rightarrow <\text{id}> = <\text{expr}> \\
 &\Rightarrow A = <\text{expr}> \\
 &\Rightarrow A = <\text{term}> \\
 &\Rightarrow A = <\text{term}> * <\text{factor}> \\
 &\Rightarrow A = <\text{factor}> * <\text{factor}> \\
 &\Rightarrow A = <\text{id}> * <\text{factor}> \\
 &\Rightarrow A = A * <\text{factor}> \\
 &\Rightarrow A = A * (<\text{expr}>) \\
 &\Rightarrow A = A * (<\text{expr}> + <\text{term}>) \\
 &\Rightarrow A = A * (<\text{term}> + <\text{term}>) \\
 &\Rightarrow A = A * (<\text{factor}> + <\text{term}>) \\
 &\Rightarrow A = A * (<\text{id}> + <\text{term}>) \\
 &\Rightarrow A = A * (B + <\text{term}>) \\
 &\Rightarrow A = A * (B + <\text{factor}>) \\
 &\Rightarrow A = A * (B + <\text{id}>) \\
 &\Rightarrow A = A * (B + C)
 \end{aligned}$$



(d)  $A = B * (C * (A+B))$

$$\langle \text{assign} \rangle \Rightarrow \langle \text{id} \rangle = \langle \text{expr} \rangle$$

$$\Rightarrow A = \langle \text{expr} \rangle$$

$$\Rightarrow A = \langle \text{term} \rangle$$

$$\Rightarrow A = \langle \text{term} \rangle * \langle \text{factor} \rangle$$

$$\Rightarrow A = \langle \text{factor} \rangle * \langle \text{factor} \rangle$$

$$\Rightarrow A = \langle \text{id} \rangle * \langle \text{factor} \rangle$$

$$\Rightarrow A = B * \langle \text{factor} \rangle$$

$$\Rightarrow A = B * (\langle \text{term} \rangle)$$

$$\Rightarrow A = B * (\langle \text{term} \rangle * \langle \text{factor} \rangle)$$

$$\Rightarrow A = B * (\langle \text{factor} \rangle * \langle \text{factor} \rangle)$$

$$\Rightarrow A = B * \{ \langle \text{id} \rangle * \langle \text{factor} \rangle \}$$

$$\Rightarrow A = B * (C * \langle \text{factor} \rangle)$$

$$\Rightarrow A = B * (C * (\langle \text{expr} \rangle))$$

$$\Rightarrow A = B * (C * (\langle \text{expr} \rangle + \langle \text{term} \rangle))$$

$$\Rightarrow A = B * (C * (\langle \text{term} \rangle + \langle \text{term} \rangle))$$

$$\Rightarrow A = B * (C * (\langle \text{factor} \rangle + \langle \text{term} \rangle))$$

$$\Rightarrow A = B * (C * (\langle \text{id} \rangle + \langle \text{term} \rangle))$$

$$\Rightarrow A = B * (C * (A + \langle \text{term} \rangle))$$

$$\Rightarrow A = B * (C * (A + \langle \text{factor} \rangle))$$

$$\Rightarrow A = B * (C * (A + \langle \text{id} \rangle))$$

$$\Rightarrow A = B * (C * (A + B))$$

$\langle \text{assign} \rangle$

$\langle \text{id} \rangle = \langle \text{expr} \rangle$

A

$\langle \text{term} \rangle$

$\langle \text{term} \rangle$

\*

$\langle \text{factor} \rangle$

$\langle \text{factor} \rangle$

(

$\langle \text{expr} \rangle$

)

$\langle \text{id} \rangle$

$\langle \text{term} \rangle$

B

$\langle \text{term} \rangle$

\*

$\langle \text{factor} \rangle$

$\langle \text{factor} \rangle$

(

$\langle \text{expr} \rangle$

)

$\langle \text{id} \rangle$

/

$\langle \text{expr} \rangle$

+

$\langle \text{term} \rangle$

C

$\langle \text{term} \rangle$

$\langle \text{factor} \rangle$

$\langle \text{factor} \rangle$

$\langle \text{id} \rangle$

$\langle \text{id} \rangle$

$\langle \text{id} \rangle$

A

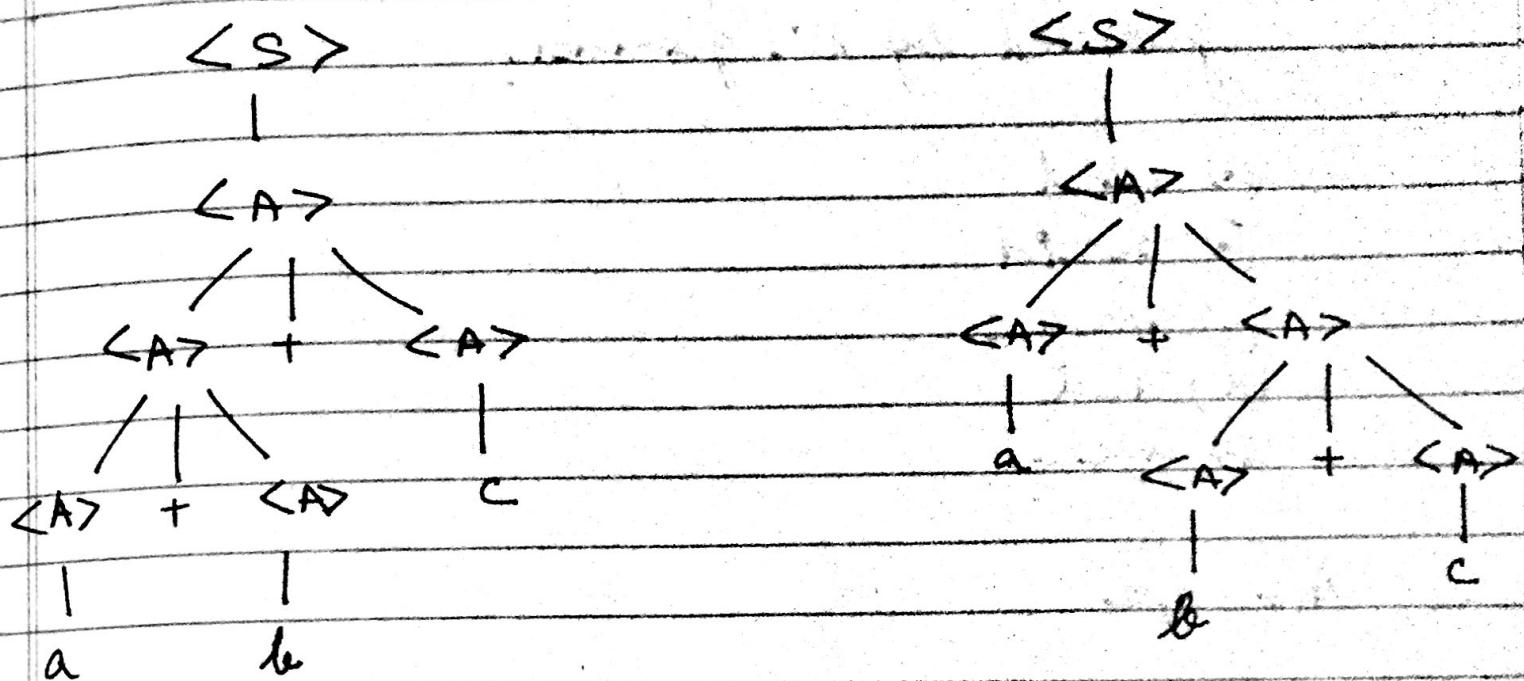
B

Q8. Prove that the following grammar is Ambiguous.

$$\langle S \rangle \rightarrow \langle A \rangle$$

$$\langle A \rangle \rightarrow \langle A \rangle + \langle A \rangle \mid \langle id \rangle$$

$$\langle id \rangle \rightarrow a \mid b \mid c$$



Here, for  $a+b+c$  there are two possible parse trees generated from the grammar.

Hence the grammar is ambiguous.

Q11. Consider the following grammar:

$$\langle S \rangle \Rightarrow \langle A \rangle a \langle B \rangle b$$

$$\langle A \rangle \Rightarrow \langle A \rangle b \mid b$$

$$\langle B \rangle \Rightarrow a \langle B \rangle \mid a$$

Which of the following sentences are in the language generated by this grammar?

- (a) babb
- (b) bbbabb
- (c) bbaaaa
- (d) bbaabb

Ans. a, d

Q12. Consider the following grammar:

$$\langle S \rangle \rightarrow a \langle S \rangle c \langle B \rangle \mid \langle A \rangle \mid b$$

$$\langle A \rangle \rightarrow c \langle A \rangle \mid c$$

$$\langle B \rangle \rightarrow d \mid \langle A \rangle$$

Which of the following sentences are in the language generated by this grammar?

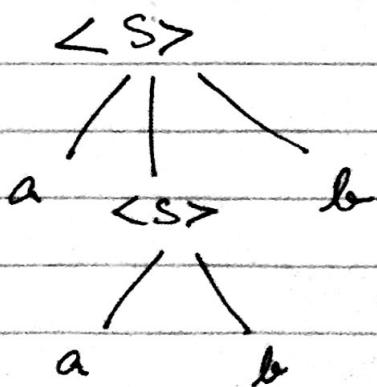
- (a) abcd
- (b) accbdc
- (c) accc bcc
- (d) acd
- (e) accc

Ans. a, c

Q13. Write a grammar for the language consisting of strings that have  $n$  copies of the letter a followed by the same number of copies of the letter b, where  $n \geq 0$ . For example, the strings ab, aabb, and aaaaaaaaaaaaaaaaaaabb are in the language but a, ab, ba, and aabb are not.

$$\langle S \rangle \rightarrow ab \mid a \langle S \rangle b$$

Q14. Draw parse trees for the sentences aabb, and aaaaaaaaaaaaaaaaaaabb, as derived from the grammar of Problem 3.



Parse tree for aabb

Parse tree for `available`

