

CMSC 331: Principles of Programming Language Homework 1 Solutions

1. Given the grammar:

$S \rightarrow I = E$

$I \rightarrow a \mid b \mid c$

$E \rightarrow I + E \mid I * E \mid (E) \mid I$

Draw a parse tree and leftmost derivation for the following sentences:

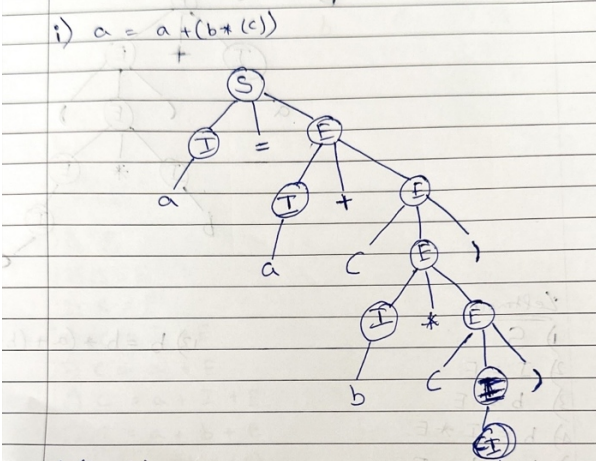
I. $a = a + (b * (c))$

II. $b = b * (a + (b * c))$

III. $c = a + b + (c)$

Sol:

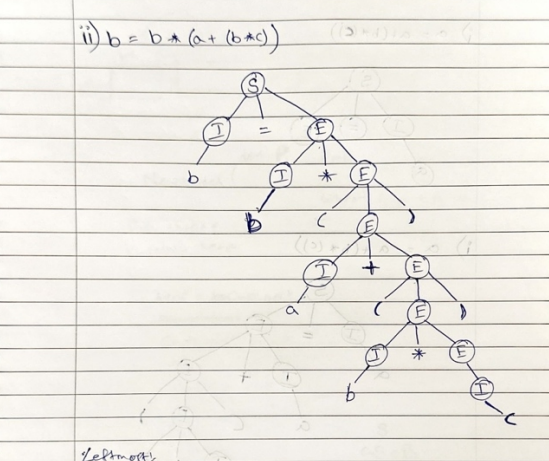
i) $a = a + (b * (c))$



Leftmost:

1) S	7) $a = a + (I * E)$
2) $I = E$	8) $a = a + (b * E)$
3) $a = E$	9) $a = a + (b * (E))$
4) $a = I + E$	10) $a = a + (b * (I))$
5) $a = a + E$	11) $a = a + (b * (c))$
6) $a = a + (E)$	

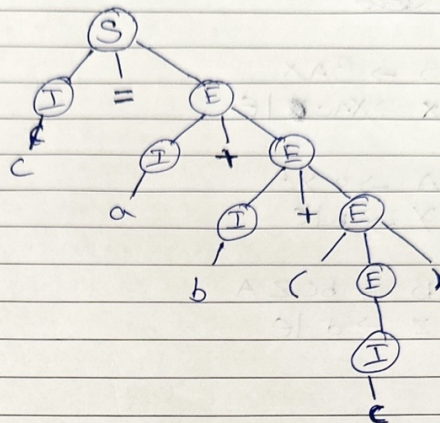
ii) $b = b * (a + (b * c))$



Leftmost:

1) S	13) $b = b * (a + (b * c))$
2) $I = E$	
3) $b = E$	
4) $b = I * E$	
5) $b = b * E$	
6) $b = b * (E)$	
7) $b = b * (I + E)$	
8) $b = b * (a + E)$	
9) $b = b * (a + (E))$	
10) $b = b * (a + (I * E))$	
11) $b = b * (a + (b * E))$	
12) $b = b * (a + (b * I))$	

iii) $C = a + b + (c)$



Leftmost:-

- 1) S
- 2) $I = E$
- 3) $C = E$
- 4) $C = I + E$
- 5) $C = a + E$
- 6) $C = a + I + E$
- 7) $C = a + b + E$
- 8) $C = a + b + (E)$
- 9) $C = a + b + (I)$
- 10) $C = a + b + (c)$

2. Demonstrate that the following grammar is ambiguous:

$S \rightarrow I$

$I \rightarrow E + E \mid (E) \mid E * E \mid I$

$E \rightarrow a-z \mid 0-9$

Here are two different left-most derivation of the same string:

$S \rightarrow I \rightarrow E * E \rightarrow E + E * E \rightarrow a + b + c$

$S \rightarrow I \rightarrow E + E \rightarrow E + E * E \rightarrow a + b + c$

3. Determine which of the sentences (a-e) are in this grammar:

$S \rightarrow aScB \mid A \mid b$

$A \rightarrow cA \mid c$

$B \rightarrow d \mid A$

a. $abcbdb$ not in grammar

S can derive a word that starts with a ($aScB$) or c (A) or b ;
must use first prod; first S must go $S \rightarrow b$, so we have $abcB$
but, " B " only goes to strings that are just b or start with c

b. $abababc$ " b " can only be produced from an S ; S is
always followed by " c "; therefore " ba " can't be there.

c. $cccc$ $S \rightarrow A \rightarrow cA \rightarrow ccA \rightarrow cccA \rightarrow cccc$

d. $accd$ $S \rightarrow aScB \rightarrow aAcB \rightarrow accB \rightarrow accd$

e. $aabdcdd$ $S \rightarrow aScB \rightarrow aaScBcB \rightarrow aabcBcB \rightarrow aabdcB$
 $\rightarrow aabdcA \rightarrow aabdcdd$

4. Convert the following basic EBNF rules to the basic BNF notation:

$S \rightarrow BA\{Ac\}$

$A \rightarrow a[b]A$

$B \rightarrow bc[d]A$

Sol: $S \rightarrow BAX$

$X \rightarrow XAc \mid Ac$

$A \rightarrow aYA$

$Y \rightarrow b \mid bY$

$B \rightarrow bcZA$

$Z \rightarrow d \mid dZ$