

Problem 2

Original Grammar:

<stmt> → if <expr> then <stmt>
 → if <expr> then <stmt> else <stmt>

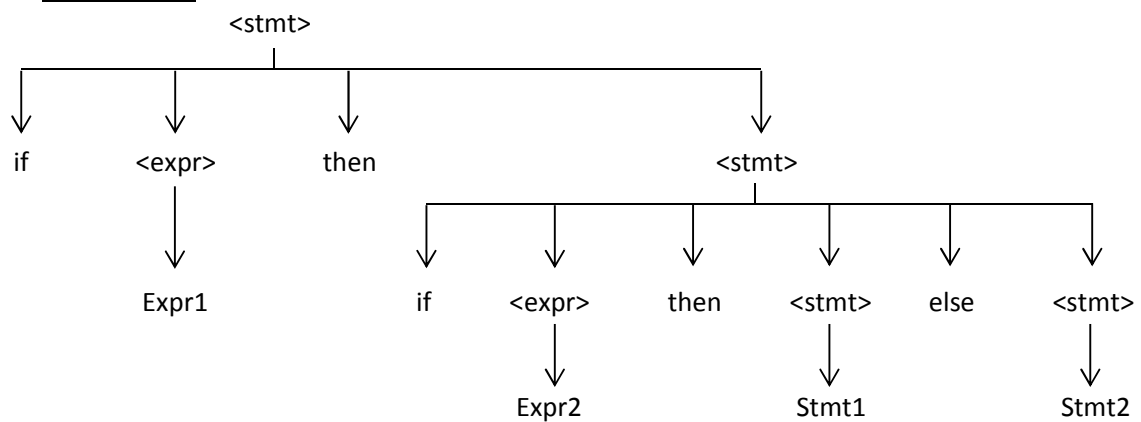
a) **Original Statement:**

if Expr1 then if Expr2 then Stmt1 else Stmt2

Leftmost Derivation 1:

<stmt> → if <expr> then <stmt>
 → if Expr1 then <stmt>
 → if Expr1 then if <expr> then <stmt> else <stmt>
 → if Expr1 then if Expr2 then <stmt> else <stmt>
 → if Expr1 then if Expr2 then Stmt1 else <stmt>
 → **if Expr1 then if Expr2 then Stmt1 else Stmt2**

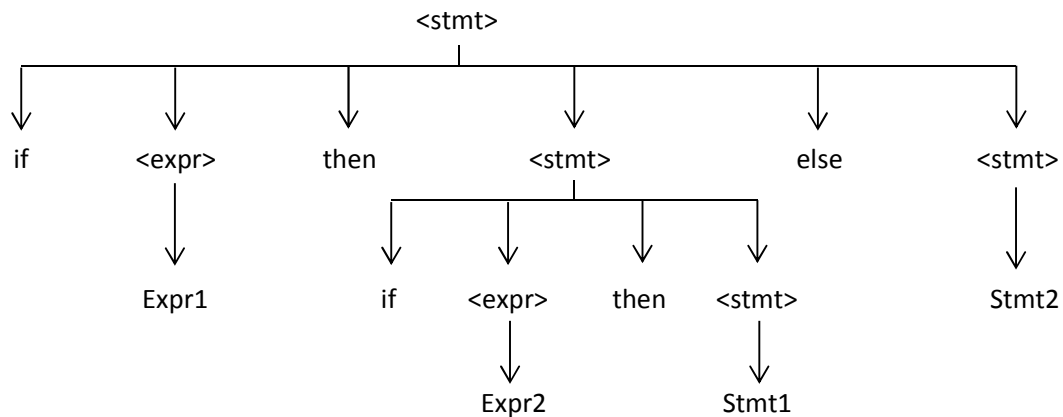
Parse Tree 1:



Leftmost Derivation 2:

<stmt> → if <expr> then <stmt> else <stmt>
 → if Expr1 then <stmt> else <stmt>
 → if Expr1 then if <expr> then <stmt> else <stmt>
 → if Expr1 then if Expr2 then <stmt> else <stmt>
 → if Expr1 then if Expr2 then Stmt1 else <stmt>
 → **if Expr1 then if Expr2 then Stmt1 else Stmt2**

Parse Tree 2:



b) **Modified Grammar:**

<stmt> → <closed>
 → <open>
<closed> → if <expr> then <closed> else <closed>
<open> → if <expr> then <stmt>
 → if <expr> then <closed> else <open>

Leftmost Derivation:

<stmt> → <open>
 → if <expr> then <stmt>
 → if Expr1 then <stmt>
 → if Expr1 then <closed>
 → if Expr1 then if <expr> then <closed> else <closed>
 → if Expr1 then if Expr2 then <closed> else <closed>
 → if Expr1 then if Expr2 then Stmt1 else <closed>
 → **if Expr1 then if Expr2 then Stmt1 else Stmt2**

Parse Tree 2:

