Question 1:

Let G: < *stmt* > → < assignment >/ < *if-stmt* >

< *if-stmt* > → *if* (<*cond.*>) <*stmt*> / *if* (<*cond.*>) <*stmt*> *else* <*stmt*>

<*cond.*> → <*arth.E*> <*relop*> <*arth.E*>

< assignment > → <*id*> = <*arth.E*>

<*arth.E*> →<*id*> / <*const.*>

<*relop*> → < / > / = =

<*id*> → x / y

<*const.*> → -1/ 1 / 2/ 0

Construct the parse tree and left most derivation of the following string:

if (x >2) y=1

else if (x= =2) y=0

else y = -1

Answer

< *stmt* > ⇒ < *if-stmt* >

⇒ *if* (<*cond.*>) <*stmt*> *else* <*stmt*>

⇒ *if* (<*arth.E*> <*relop*> <*arth.E*>) <*stmt*> *else* <*stmt*>

⇒ *if* (<*id*> <*relop*> <*arth.E*>) <*stmt*> *else* <*stmt*>

⇒ *if* (*x* <*relop*> <*arth.E*>) <*stmt*> *else* <*stmt*>

⇒ *if* (*x* > <*arth.E*>) <*stmt*> *else* <*stmt*>

⇒ *if* (*x* ><*const.*>) <*stmt*> *else* <*stmt*>

⇒ *if* (*x* >2)<*stmt*> *else* <*stmt*>

⇒ *if (x >2) < assignment > else <stmt>*

⇒ *if (x >2) <id> = <arth.E>else <stmt>*

⇒ *if (x >2) y = <arth.E>else <stmt>*

⇒ *if (x >2) y = <const.> else <stmt>*

⇒ *if (x >2) y = 1 else <stmt>*

⇒ *if (x >2) y = 1 else < if-stmt >*

⇒ *if (x >2) y = 1 else if (<cond.>) <stmt> else <stmt>*

⇒ *if (x >2) y = 1 else if (<arth.E> <relop> <arth.E>)<stmt> else <stmt>*

⇒ *if (x >2) y = 1 else if (<id> <relop> <arth.E>)<stmt> else <stmt>*

⇒ *if (x >2) y = 1 else if (x <relop> <arth.E>)<stmt> else <stmt>*

⇒ *if (x >2) y = 1 else if (x == <arth.E>)<stmt> else <stmt>*

⇒ *if (x >2) y = 1 else if (x== <const.>)<stmt> else <stmt>*

⇒ *if (x >2) y = 1 else if (x== 2)<stmt> else <stmt>*

⇒ *if (x >2) y = 1 else if (x== 2) < assignment > else <stmt>*

⇒ *if (x >2) y = 1 else if (x== 2) <id> = <arth.E>else <stmt>*

⇒ *if (x >2) y = 1 else if (x== 2) y = <arth.E>else <stmt>*

⇒ *if (x >2) y = 1 else if (x== 2) y = <const.>else <stmt>*

⇒ *if (x >2) y = 1 else if (x== 2) y = 0 else <stmt>*

⇒ *if (x >2) y = 1 else if (x== 2) y = 0 else* **< assignment >**

⇒ *if (x >2) y = 1 else if (x== 2) y = 0 else <id> = <arth.E>*

⇒ *if (x >2) y = 1 else if (x== 2) y = 0 else y = <arth.E>*

⇒ *if (x >2) y = 1 else if (x== 2) y = 0 else y = <const.>*

⇒ *if (x >2) y = 1 else if (x== 2) y = 0 else y = -1*
