a) Inorder traversal is: x / y + 3 \* b / c

To construct the expression tree, we start with the expression "x / y + 3 \* b / c".

Step 1: Identify the operators and operands in the expression: Operators: /, + Operands: x, y, 3, b, c

Step 2: Convert the expression into postfix notation (Reverse Polish Notation) to help build the expression tree:

Original Expression: x / y + 3 \* b / c Postfix Expression: x y / 3 b \* c / +

Step 3: Build the expression tree using the postfix expression:

+

/ \

/ \

/ \

/ \

x \*

/ \

/ \

/ \

/ \

/ \

/ \

/ \

/ \

/ \

/ \

/ \

/ \

/ \

y 3

b) Postorder traversal is: x y z + a b - c \* / -

Step 1: Identify the operators and operands in the expression: Operators: +, -, \*, / Operands: x, y, z, a, b, c

Step 2: Build the expression tree using the postorder traversal:

-

/ \

/ \

/ \

+ /

/ \ / \

/ \ / \

x y z \*

/ \

/ \

a b

/

/

C

c) Preorder traversal is: \* + a - x y / c d

Step 1: Identify the operators and operands in the expression: Operators: \*, +, -, / Operands: a, x, y, c, d

Step 2: Build the expression tree using the preorder traversal:

\*

/ \

+ /

/ \ / \

a - c d

/ \

x y