

Main  
preToPostFix(preFixExpr, postFixExpr)  
postFixExpr = "AB\* C+EF/-"

preToPostFix("-+\*ABC/EF", "")  
operator = "- \0"  
lenPreFix = findExprLen(+\*ABC/EF)  
temp = "+\*ABC \0"  
preToPostFix(+\*ABC, postFix1)  
postFix1 = AB\*C+  
preToPostFix(/EF, postFix2)  
postFix2 = EF/  
return

findExprLen(+\*ABC/EF)  
len1 = findExprLen(\*ABC/EF)  
len2 = findExprLen(C/EF)  
return 5

findExprLen(\*ABC/EF)  
len1 = findExprLen(ABC/EF)  
len2 = findExprLen(BC/EF)  
return 3

findExprLen(ABC/EF)  
len1 = 0  
len2 = 0  
return 1

findExprLen(BC/EF)  
len1 = 0  
len2 = 0  
return 1

findExprLen(C/EF)  
len1 = 0  
len2 = 0  
return 1

preToPostFix(/EF, postFix2)  
lenPreFix = findExprLen(EF)  
preToPostFix(E, postFix1)  
postFix1 = E  
preToPostFix(F, postFix2)  
postFix2 = F  
return

findExprLen(EF)  
len1 = 0  
len2 = 0  
return 1

preToPostFix(E, postFix1)  
return

preToPostFix(F, postFix2)  
return

preToPostFix(+\*ABC, postFix1)  
lenPreFix = findExprLen(\*ABC)  
preToPostFix(\*AB, postFix1)  
postFix1 = AB\*  
preToPostFix(C, postFix2)  
postFix2 = C  
return

findExprLen(\*ABC)  
len1 = findExprLen(ABC)  
len2 = findExprLen(BC)  
return 3

findExprLen(ABC)  
len1 = 0  
len2 = 0  
return 1

findExprLen(BC)  
len1 = 0  
len2 = 0  
return 1

preToPostFix(\*AB, postFix1)  
lenPreFix = findExprLen(AB)  
preToPostFix(A, postFix1)  
postFix1 = A  
preToPostFix(B, postFix2)  
postFix2 = B  
return

findExprLen(AB)  
len1 = 0  
len2 = 0  
return 1

preToPostFix(A, postFix1)  
return

preToPostFix(B, postFix2)  
return

preToPostFix(C, postFix2)  
return