

## *Application – Postfix Expression*

## *Application – Postfix Expression*

Infix

Postfix

## *Application – Postfix Expression*

Infix

Postfix

$A + B$

## *Application – Postfix Expression*

Infix

$A + B$

Postfix

$AB+$

## *Application – Postfix Expression*

Infix

Postfix

operator

$A + B$

$AB+$

## *Application – Postfix Expression*

Infix

Postfix

operand operator operand

A + B

AB+

## *Application – Postfix Expression*

Infix

operand operator operand

$A + B$

Postfix

operand operand

$AB+$

## *Application – Postfix Expression*

Infix

operand operator operand

$A + B$

Postfix

operand operand operator

$AB+$



## *Application – Postfix Expression*

Infix

operand operator operand

$A + B$

$A - B$

Postfix

operand operand operator

$AB+$

## *Application – Postfix Expression*

Infix

operand operator operand

$A + B$

$A - B$

Postfix

operand operand operator

$AB+$

$AB-$

## *Application – Postfix Expression*

Infix

operand operator operand

$A + B$

$A - B$

$A + B + C$

Postfix

operand operand operator

$AB+$

$AB-$

## *Application – Postfix Expression*

Infix

operand operator operand

$A + B$

$A - B$

$A + B + C$

Postfix

operand operand operator

$AB+$

$AB-$

$A + B + C$

## *Application – Postfix Expression*

Infix

operand operator operand

$A + B$

$A - B$

$A + B + C$

Postfix

operand operand operator

$AB+$

$AB-$

$A + B + C$

## *Application – Postfix Expression*

Infix

operand operator operand

$A + B$

$A - B$

$A + B + C$

Postfix

operand operand operator

$AB+$

$AB-$

$AB++C$

## *Application – Postfix Expression*

Infix

operand operator operand

$A + B$

$A - B$

$A + B + C$

Postfix

operand operand operator

$AB+$

$AB-$

$AB + C+$

## *Application – Postfix Expression*

Infix

operand operator operand

$$A + B$$

$$A - B$$

$$A + B + C$$

$$(A + B) * (C + D)$$

Postfix

operand operand operator

$$AB+$$

$$AB-$$

$$AB + C +$$



## *Application – Postfix Expression*

Infix

operand operator operand

$$A + B$$

$$A - B$$

$$A + B + C$$

$$(A + B) * (C + D)$$

Postfix

operand operand operator

$$AB +$$

$$AB -$$

$$AB + C +$$

$$(A + B) * (C + D)$$

## *Application – Postfix Expression*

Infix

operand operator operand

$$A + B$$

$$A - B$$

$$A + B + C$$

$$(A + B) * (C + D)$$

Postfix

operand operand operator

$$AB +$$

$$AB -$$

$$AB + C +$$

$$(AB+) * (C + D)$$

## *Application – Postfix Expression*

### Infix

operand operator operand

$$A + B$$

$$A - B$$

$$A + B + C$$

$$(A + B) * (C + D)$$

### Postfix

operand operand operator

$$AB+$$

$$AB-$$

$$AB + C +$$

$$(AB+) * (CD+)$$

## *Application – Postfix Expression*

### Infix

operand operator operand

$$A + B$$

$$A - B$$

$$A + B + C$$

$$(A + B) * (C + D)$$

### Postfix

operand operand operator

$$AB+$$

$$AB-$$

$$AB + C +$$

$$(AB+)(CD+)*$$

## *Application – Postfix Expression*

Infix

operand operator operand

$$A + B$$

$$A - B$$

$$A + B + C$$

$$(A + B) * (C + D)$$

Postfix

operand operand operator

$$AB +$$

$$AB -$$

$$AB + C +$$

$$AB + CD + *$$

## *Infix to Postfix*

Infix

Postfix

## *Infix to Postfix*

Infix

A/B/C

Postfix

## *Infix to Postfix*

Infix

A/B/C

Postfix

AB/C/



## *Infix to Postfix*

Infix

$A/B/C$

$A + B * C$

Postfix

$AB/C/$

## *Infix to Postfix*

Infix

$A/B/C$

$A + B * C$

Postfix

$AB/C/$

$ABC * +$

## *Infix to Postfix*

Infix

$A/B/C$

$A + B * C$

$A * B + C$

Postfix

$AB/C/$

$ABC * +$

## *Infix to Postfix*

Infix

$A/B/C$

$A + B * C$

$A * B + C$

Postfix

$AB/C/$

$ABC * +$

$AB * C +$

## *Infix to Postfix*

Infix

$A/B/C$

$A + B * C$

$A * B + C$

$A + B * C + D$

Postfix

$AB/C/$

$ABC * +$

$AB * C +$

## *Infix to Postfix*

Infix

$A/B/C$

$A + B * C$

$A * B + C$

$A + B * C + D$

Postfix

$AB/C/$

$ABC * +$

$AB * C +$

$ABC * + D +$

## *Infix to Postfix*

Infix

$$A/B/C$$

$$A + B * C$$

$$A * B + C$$

$$A + B * C + D$$

$$(A + B) * (C + D)$$

Postfix

$$AB/C/$$

$$ABC * +$$

$$AB * C +$$

$$ABC * + D +$$

## *Infix to Postfix*

**Infix**

$A/B/C$

$A + B * C$

$A * B + C$

$A + B * C + D$

$(A + B) * (C + D)$

**Postfix**

$AB/C/$

$ABC * +$

$AB * C +$

$ABC * + D +$

$AB + CD + *$



# *Postfix Expressions*

- *Evaluation*

# *Postfix Expressions*

## *- Evaluation*

$$A + B$$

# *Postfix Expressions*

## *- Evaluation*

$$A + B \implies AB +$$

# *Postfix Expressions*

## *- Evaluation*



$$A + B \implies AB +$$

$$2 + 3 \implies 2\ 3\ +$$

## *Postfix Expressions*

### *- Evaluation*



$$A + B \implies AB +$$

$$2 + 3 \implies \textcolor{red}{2} 3 +$$

## *Postfix Expressions - Evaluation*

3
2

$$A + B \implies AB +$$

$$2 + 3 \implies 2 \text{ } 3 +$$

# *Postfix Expressions*

## *- Evaluation*



$$A + B \implies AB +$$

$$2 + 3 \implies 2 \ 3 \ +$$

## *Postfix Expressions*

### *- Evaluation*



$$A + B \implies AB +$$

$$2 \ + \ 3 \implies 2 \ 3 \ + \ = \ 5$$



# *Postfix Expressions*

## *- Evaluation*



$$A + B + C$$

# *Postfix Expressions*

## *- Evaluation*



$$A + B + C \implies AB + C +$$

# *Postfix Expressions*

## *- Evaluation*

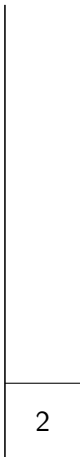


$$A + B + C \implies AB + C +$$

$$2 + 3 + 4 \implies 2\ 3 + 4 +$$

## *Postfix Expressions*

### *- Evaluation*



$$A + B + C \implies AB + C +$$

$$2 + 3 + 4 \implies \textcolor{red}{2} 3 + 4 +$$

## *Postfix Expressions*

### *- Evaluation*

3
2

$$A + B + C \implies AB + C +$$

$$2 + 3 + 4 \implies 2 \text{ } 3 + 4 +$$

# *Postfix Expressions*

## *- Evaluation*



$$A + B + C \implies AB + C +$$

$$2 + 3 + 4 \implies 2 \ 3 \text{ + } 4 \ +$$

## *Postfix Expressions*

### *- Evaluation*

4
5

$$A + B + C \implies AB + C +$$

$$2 + 3 + 4 \implies 2 \ 3 + \textcolor{red}{4} +$$

# *Postfix Expressions*

## *- Evaluation*

9

$$A + B + C \implies AB + C +$$

$$2 + 3 + 4 \implies 2 \ 3 + 4 \text{ +}$$



## *Postfix Expressions*

### *- Evaluation*



$$A + B + C \implies AB + C +$$

$$2 + 3 + 4 \implies 2\ 3 + 4 + = 9$$

# *Postfix Expressions*

## *- Evaluation*



$$(A + B) * (C + D)$$

## *Postfix Expressions* *- Evaluation*



$$(A + B) * (C + D) \implies AB + CD + *$$

# *Postfix Expressions*

## *- Evaluation*



$$(A + B) * (C + D) \implies AB + CD + *$$

$$(2 + 3) * (4 + 5) \implies 2\ 3 + 4\ 5 + *$$

## *Postfix Expressions*

### *- Evaluation*



$$(A + B) * (C + D) \implies AB + CD + *$$

$$(2 + 3) * (4 + 5) \implies 2\ 3 + 4\ 5 + *$$

## *Postfix Expressions*

### *- Evaluation*

3
2

$$(A + B) * (C + D) \implies AB + CD + *$$

$$(2 + 3) * (4 + 5) \implies 2 \textcolor{red}{3} + 4 \ 5 + *$$

# *Postfix Expressions*

## *- Evaluation*



$$(A + B) * (C + D) \implies AB + CD + *$$

$$(2 + 3) * (4 + 5) \implies 2\ 3\ +\ 4\ 5\ +\ *$$

## *Postfix Expressions*

### *- Evaluation*

4
5

$$(A + B) * (C + D) \implies AB + CD + *$$

$$(2 + 3) * (4 + 5) \implies 2\ 3 + \textcolor{red}{4}\ 5 + *$$



## *Postfix Expressions*

### *- Evaluation*

5
4
5

$$(A + B) * (C + D) \implies AB + CD + *$$

$$(2 + 3) * (4 + 5) \implies 2\ 3 + 4\ 5 + *$$

# *Postfix Expressions*

## *- Evaluation*

9
5

$$(A + B) * (C + D) \implies AB + CD + *$$

$$(2 + 3) * (4 + 5) \implies 2\ 3 + 4\ 5\ +\ *$$

## *Postfix Expressions*

### *- Evaluation*

45

$$(A + B) * (C + D) \implies AB + CD + *$$

$$(2 + 3) * (4 + 5) \implies 2\ 3 + 4\ 5 + *$$

# *Postfix Expressions*

## *- Evaluation*



$$(A + B) * (C + D) \implies AB + CD + *$$

$$(2 + 3) * (4 + 5) \implies 2\ 3 + 4\ 5 + * = 45$$

*Why Postfix?*

## *Why Postfix?*

( 2 + 3 ) \* ( 4 + 5 )

## *Why Postfix?*

$$(2 + 3) * (4 + 5)$$


### *Tasks*

- Among all the operators, find the operator with highest precedence.

## Why Postfix?

$$(2 + 3) * (4 + 5)$$


### Tasks

- Among all the operators, find the operator with highest precedence.
- Find all paranthesis, and look at expressions within paranthesis.



## *Why Postfix?*

$$\underline{( 2 + 3 )} * \underline{( 4 + 5 )}$$

### *Tasks*

- Among all the operators, find the operator with highest precedence.
- Find all paranthesis, and look at expressions within paranthesis.

## *Why Postfix?*

$$\underline{(2 + 3)} * \underline{(4 + 5)}$$

### *Tasks*

- Among all the operators, find the operator with highest precedence.
- Find all parenthesis, and look at expressions within parenthesis.
- etc.

## *Why Postfix?*

2 3 + 4 5 + \*



## *Why Postfix?*

2 3 + 4 5 + \*



2

## *Why Postfix?*

2 3 + 4 5 + \*



## *Why Postfix?*

2 3 + 4 5 + \*



## *Why Postfix?*

2 3 + 4 5 + \*



## *Why Postfix?*

2 3 + 4 5 + \*

5
4
5



## *Why Postfix?*

2 3 + 4 5 + \*

9
5

## *Why Postfix?*

2 3 + 4 5 + \*



45

## *Why Postfix?*

$$2\ 3\ +\ 4\ 5\ +\ * = 45$$



## *Why Postfix?*

### *Postfix Expression*

- Evaluate from left to right.

## *Why Postfix?*

### *Postfix Expression*

- Evaluate from left to right.

### *Infix Expression*

- Among all the operators, find the operator with highest precedence.
- Find all paranthesis, and look at expressions within paranthesis.
- etc.

## *Why Postfix?*

### *Postfix Expression*

- Evaluate from left to right.

### *Infix Expression*

- Among all the operators, find the operator with highest precedence.
- Find all paranthesis, and look at expressions within paranthesis.
- etc.

It is easier for computers to evaluate postfix expressions.



## *Infix to Postfix*

$$A + B \implies AB +$$

*Input*

$$A + B$$

*Output*





## *Infix to Postfix*

$$A + B \implies AB +$$

*Input*

$$A + B$$

*Output*



## *Rules*

Input Operand  $\Rightarrow$  Print

## *Infix to Postfix*

$$A + B \implies AB +$$

*Input*

$$A + B$$

*Output*

A

## *Infix to Postfix*

$$A + B \implies AB+$$

*Input*

$$A + B$$

*Output*

A

## *Rules*

Input Operand  $\Rightarrow$  Print

Input Operator + Empty Stack

## *Rules*

Input Operand  $\implies$  Print

Input Operator + Empty Stack  $\implies$  Push Operator

## *Infix to Postfix*



$$A + B \implies AB +$$

*Input*

$$A + B$$

*Output*

A

## *Infix to Postfix*



$$A + B \implies AB +$$

*Input*

$$A + B$$

*Output*

$$AB$$



## *Rules*

Input Operand  $\implies$  Print

Input Operator + Empty Stack  $\implies$  Push Operator

End of Input  $\implies$  Empty Stack

## *Infix to Postfix*



$$A + B \implies AB +$$

*Input*

$$A + B$$

*Output*

$$AB$$

## *Infix to Postfix*

$$A + B \implies AB +$$

*Input*

$$A + B$$

*Output*

$$AB +$$

## *Infix to Postfix*

$$A + B + C \implies AB + C +$$

*Input*

$$A + B + C$$

*Output*

## *Infix to Postfix*

$$A + B + C \implies AB + C +$$

*Input*

$$A + B + C$$

*Output*

A

## *Infix to Postfix*



$$A + B + C \implies AB + C +$$

*Input*

$$A + B + C$$

*Output*

A

## *Infix to Postfix*

$$A + B + C \implies AB + C +$$

*Input*

$$A + \textcolor{red}{B} + C$$

*Output*

AB

+

## *Infix to Postfix*

$$A + B + C \implies AB + C +$$

*Input*

$$A + B + C$$

*Output*

AB

+



## *Rules*

Input Operand  $\implies$  Print

Input Operator + Empty Stack  $\implies$  Push Operator

End of Input  $\implies$  Empty Stack

Input Operator  $\leq$  Stack Operator

## *Rules*

Input Operand  $\implies$  Print

Input Operator + Empty Stack  $\implies$  Push Operator

End of Input  $\implies$  Empty Stack

Input Operator  $\leq$  Stack Operator  $\implies$  Pop and Print

## *Rules*

Input Operand  $\implies$  Print

Input Operator + Empty Stack  $\implies$  Push Operator

End of Input  $\implies$  Empty Stack

Input Operator  $\leq$  Stack Operator  $\implies$  Pop and Print

Input Operator  $>$  Stack Operator

## *Rules*

Input Operand  $\implies$  Print

Input Operator + Empty Stack  $\implies$  Push Operator

End of Input  $\implies$  Empty Stack

Input Operator  $\leq$  Stack Operator  $\implies$  Pop and Print

Input Operator  $>$  Stack Operator  $\implies$  Push Operator

## *Infix to Postfix*

$$A + B + C \implies AB + C +$$

*Input*

$$A + B + C$$

*Output*

AB



## *Infix to Postfix*

$$A + B + C \implies AB + C +$$

*Input*

$$A + B + C$$

*Output*

$$AB +$$

## *Infix to Postfix*



$$A + B + C \implies AB + C +$$

*Input*

$$A + B + C$$

*Output*

$$AB +$$

## *Infix to Postfix*



$$A + B + C \implies AB + C +$$

*Input*

$$A + B + C$$

*Output*

$$AB + C$$



## *Infix to Postfix*

$$A + B + C \implies AB + C +$$

*Input*

$$A + B + C$$

*Output*

$$AB + C +$$

## *Infix to Postfix*

$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + B * C + D$$

*Output*

## *Infix to Postfix*

$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + B * C + D$$

*Output*

A

## *Infix to Postfix*

$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + B * C + D$$

*Output*

A



## *Infix to Postfix*

$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + \textcolor{red}{B} * C + D$$

*Output*

AB

+

## *Infix to Postfix*

$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + B * C + D$$

*Output*

AB



## *Rules*

Input Operand  $\implies$  Print

Input Operator + Empty Stack  $\implies$  Push Operator

End of Input  $\implies$  Empty Stack

Input Operator  $\leq$  Stack Operator  $\implies$  Pop and Print

Input Operator  $>$  Stack Operator  $\implies$  Push Operator

## *Infix to Postfix*

$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + B * C + D$$

*Output*

AB

+



## *Infix to Postfix*

$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + B * C + D$$

*Output*

AB

*
+

## *Infix to Postfix*

$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + B * C + D$$

*Output*

ABC

*
+

## *Infix to Postfix*

$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + B * C + D$$

*Output*

ABC

*
+

## *Rules*

Input Operand  $\implies$  Print

Input Operator + Empty Stack  $\implies$  Push Operator

End of Input  $\implies$  Empty Stack

Input Operator  $\leq$  Stack Operator  $\implies$  Pop and Print

Input Operator  $>$  Stack Operator  $\implies$  Push Operator

## *Infix to Postfix*

$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + B * C + D$$

*Output*

ABC

*
+

## *Infix to Postfix*

$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + B * C + D$$

*Output*

$$ABC *$$

+

## *Infix to Postfix*

$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + B * C + D$$

*Output*

$$ABC * +$$

## *Infix to Postfix*



$$A + B * C + D \implies ABC * +D +$$

*Input*

$$A + B * C + D$$

*Output*

$$ABC * +$$



## *Infix to Postfix*

$$A + B * C + D \implies ABC * + D +$$

*Input*

$$A + B * C + \textcolor{red}{D}$$

*Output*

$$ABC * + D$$



## *Infix to Postfix*

$$A + B * C + D \implies ABC * + D +$$


*Input*

$$A + B * C + D$$

*Output*

$$ABC * + D +$$

## *Infix to Postfix*



$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

## *Infix to Postfix*


$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

## *Rules*

Input Operand  $\implies$  Print

Input Operator + Empty Stack  $\implies$  Push Operator


End of Input  $\implies$  Empty Stack

Input Operator  $\leq$  Stack Operator  $\implies$  Pop and Print

Input Operator  $>$  Stack Operator  $\implies$  Push Operator

Open Paranthesis  $\implies$  Put in Stack

## *Infix to Postfix*


$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

## *Infix to Postfix*

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

(

## *Infix to Postfix*

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(\textcolor{red}{A} + B) * (C + D)$$

*Output*

A

(



## *Infix to Postfix*

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

A

(



## *Infix to Postfix*

+
(

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + \textcolor{red}{B}) * (C + D)$$

*Output*

AB

## *Rules*

Input Operand  $\implies$  Print

Input Operator + Empty Stack  $\implies$  Push Operator

End of Input  $\implies$  Empty Stack

Input Operator  $\leq$  Stack Operator  $\implies$  Pop and Print

Input Operator  $>$  Stack Operator  $\implies$  Push Operator

Closed Paranthesis  $\implies$  Pop and Print till Open Paranthesis

## *Infix to Postfix*

+
(

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

AB

## *Infix to Postfix*

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*


$$(A + B) * (C + D)$$

*Output*

$$AB +$$



## *Infix to Postfix*


$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

$$AB +$$

## *Infix to Postfix*

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

$$AB +$$

\*



## *Infix to Postfix*

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

$$AB +$$

(
*



## *Infix to Postfix*

+
(
*

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

$$AB + C$$

## *Infix to Postfix*

+
(
*

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + \textcolor{red}{D})$$

*Output*

$$AB + CD$$

## *Infix to Postfix*

+
(
*

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

$$AB + CD$$



## *Infix to Postfix*

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

$$AB + CD +$$

\*

## *Infix to Postfix*

$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$


*Output*

$$AB + CD +$$

\*



## *Infix to Postfix*


$$(A + B) * (C + D) \implies AB + CD + *$$

*Input*

$$(A + B) * (C + D)$$

*Output*

$$AB + CD + *$$


## *Infix to Postfix*


$$((A + B) * C) \wedge (D + E) / (F - G)$$

*Input*

*Output*


## *Infix to Postfix*


$$((A + B) * C) \$ (D + E) / (F - G)$$

*Input*

*Output*

## *Infix to Postfix*


$$((\textcolor{violet}{A}\textcolor{violet}{B}+) * C) \$ (D + E) / (F - G)$$

*Input*

*Output*


## *Infix to Postfix*


$$((\textcolor{violet}{A}\textcolor{violet}{B}+) * C) \$ (\textcolor{violet}{D}\textcolor{violet}{E}+) / (F - G)$$

*Input*

*Output*

## *Infix to Postfix*


$$((\textcolor{violet}{AB}+) * C) \$ (\textcolor{violet}{DE}+) / (\textcolor{violet}{FG}-)$$

*Input*

*Output*

## *Infix to Postfix*

$((AB+) C*) \$ (DE+) / (FG-)$

*Input*

*Output*

## *Infix to Postfix*

$((AB+) C*) (DE+) \$ / (FG-)$

*Input*

*Output*



## *Infix to Postfix*

$((AB+) C*) (DE+) \$ (FG-) /$

*Input*

*Output*

## *Infix to Postfix*

AB+C\*DE+\$FG- /

*Input*

*Output*

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

*Output*

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\implies AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*



## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

(
(

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((\textcolor{red}{A} + B) * C) \$ (D + E) / (F - G)$$

*Output*

A



## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

A





## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

A

+
(
(

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

AB

+
(
(

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

AB

+
(
(

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB +$$



## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB +$$

(

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB +$$

(

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB +$$



## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C$$

*
(



## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C$$

*
(

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C *$$

(

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\implies AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C *$$

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\implies AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C *$$

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C *$$

\$

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C *$$

(
\$

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * D$$

(
\$

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * D$$

+
(
\$



## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE$$

+
(
\$

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE$$

+
(
\$

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE +$$

(
\$

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE +$$

\$

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\implies AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE + \$$$

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE + \$$$

/

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE + \$$$

(
/

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE + \$F$$

(
/



## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE + \$F$$

-
(
/

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - \textcolor{red}{G})$$

*Output*

$$AB + C * DE + \$FG$$

-
(
/

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE + \$FG$$

-
(
/

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE + \$FG -$$

(
/

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\Rightarrow AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE + \$FG -$$

/

## *Infix to Postfix*

$$((A + B) * C) \$ (D + E) / (F - G)$$

$$\implies AB+C*DE+ \$FG- /$$

*Input*

$$((A + B) * C) \$ (D + E) / (F - G)$$

*Output*

$$AB + C * DE + \$FG - /$$