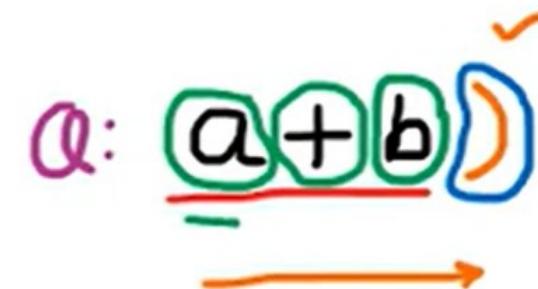


ALGORITHM: REVERSE\_POLISH (Q, P)Q.  $a+b$    P:  $ab+$ 

Suppose Q is an arithmetic expression written in infix notation. This algorithm finds the equivalent postfix expression P.



- 1 Push "(" onto STACK and add ")" to the end of Q.
- 2 Scan Q from left to right and repeat Steps 3 to 6 for each element of Q until the STACK is empty.

If an operand is encountered, add it to P. ✓ ✗ ✓

If a left parenthesis is encountered, push it onto STACK.

If an operator Θ is encountered then: ✓

a) Repeatedly pop from STACK and add to P each operator on top of STACK which has same precedence as or higher precedence than Θ. ✗

b) Add Θ to STACK.

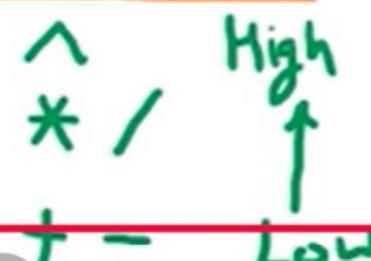
c) If a right parenthesis is encountered then:

a) Repeatedly pop from STACK and add to P each operator on top of STACK until a left parenthesis is encountered.

b) Remove the left parenthesis.

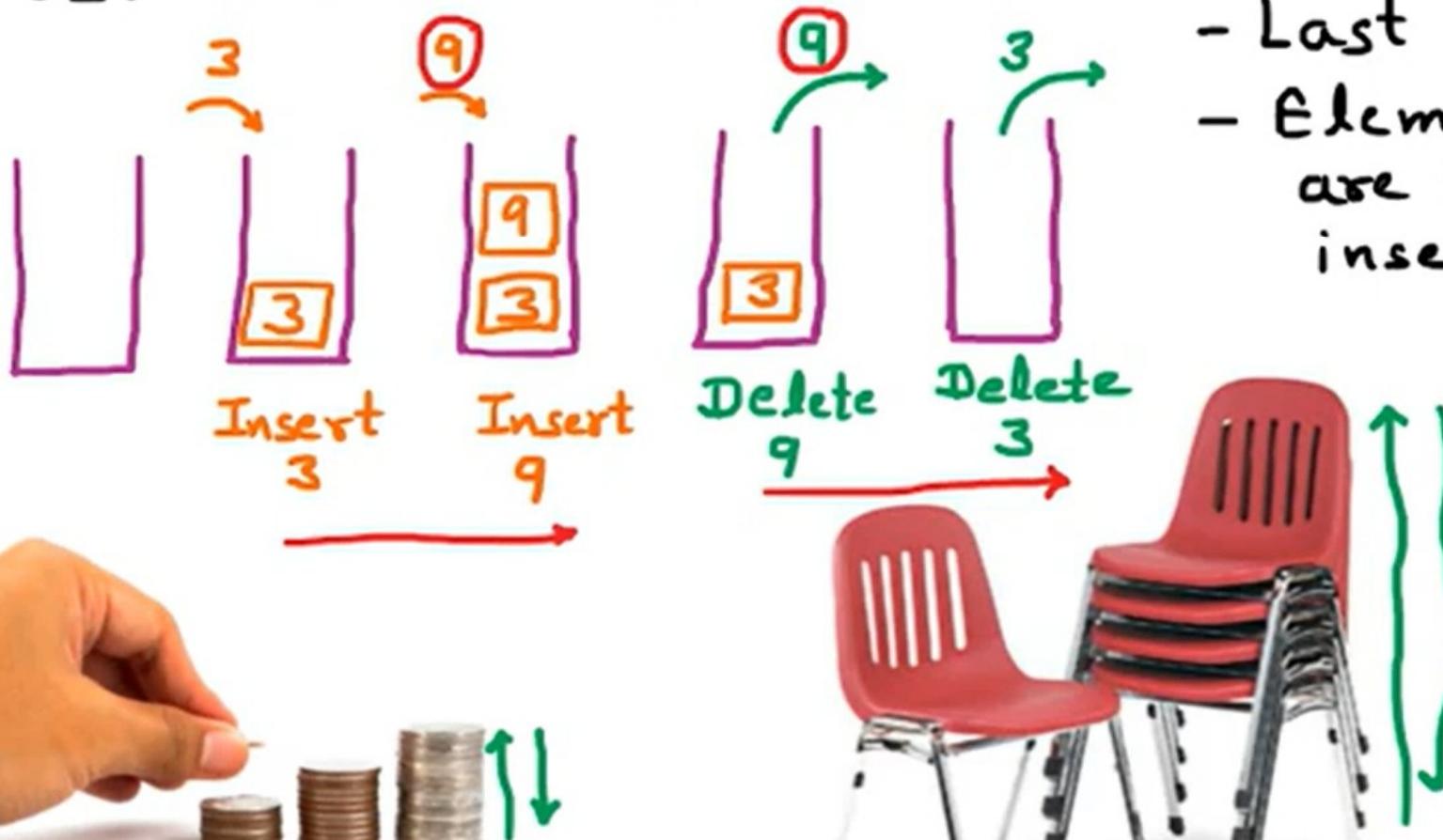
\* > +

7. Exit

P:  $ab+$ 

# Stack

- Stack is a linear structure in which items are added or deleted at one end (top of stack)
- Last element added to the stack is first item to be removed

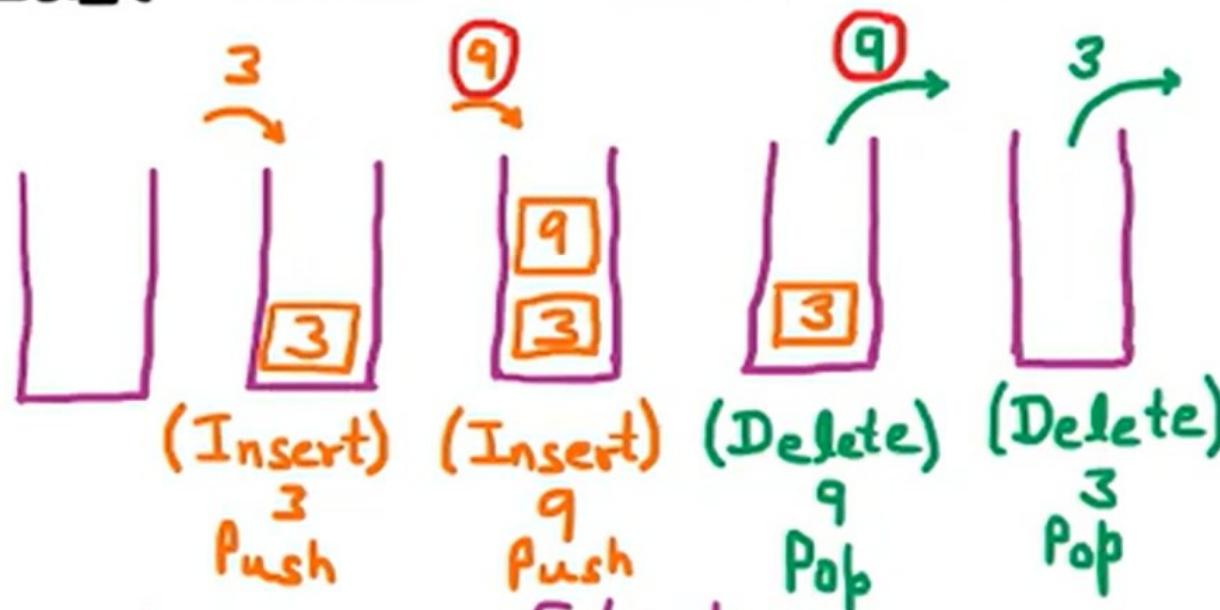


- Last In First Out (LIFO)
- Elements deleted from stack are in reverse order in which inserted



# Stack

- Stack is a linear structure in which items are added or deleted at one end (top of stack)
- Last element added to the stack is first item to be removed



- Last In First Out (LIFO)
- Elements deleted from stack are in reverse order in which inserted

## Operations on Stack

### Push

- Insert element on Stack

### Pop

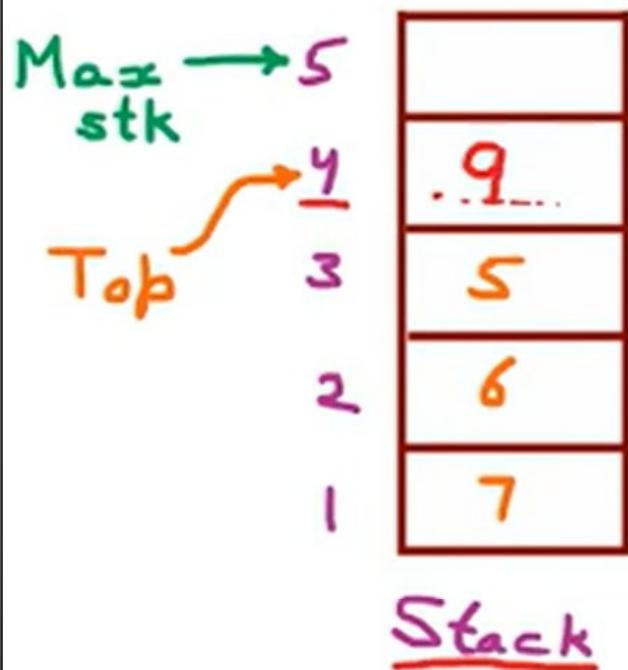
- Delete element from Stack

## ALGORITHM: PUSH (STACK, TOP, MAXSTK, ITEM)

This procedure pushes an ITEM onto a Stack.

- 1. If TOP = MAXSTK, then: Print OVERFLOW, and Return.
  - 2. Set TOP := TOP + 1.
  - 3. Set STACK [TOP] := ITEM.
  - 4. Return.
- X [

Case 3

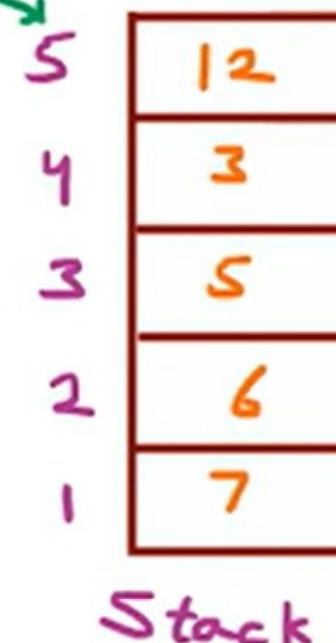


Item

9

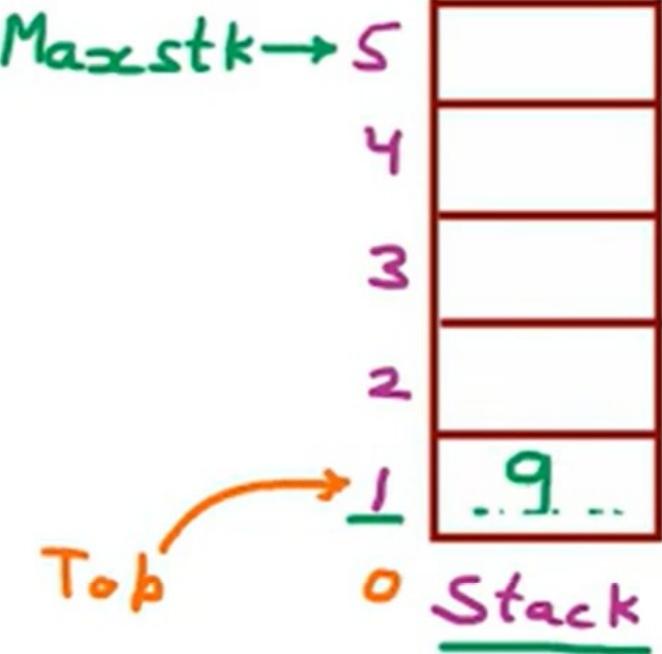
Maxstk →  
Top → 5

Case 1



Case 2

Maxstk → 5

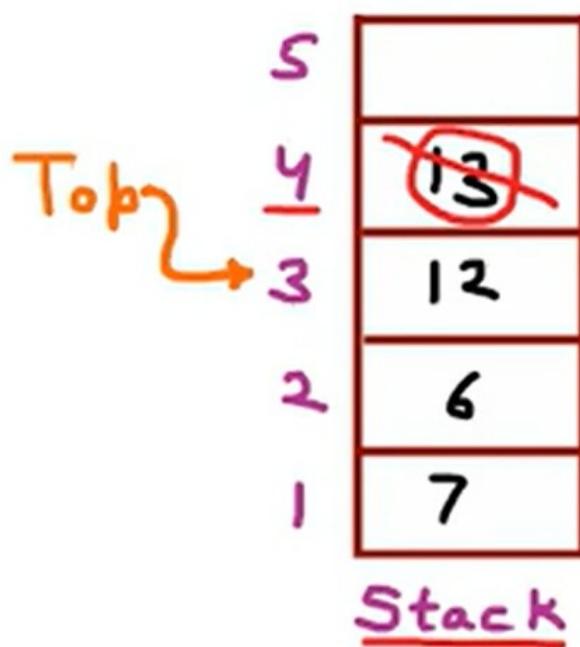


## ALGORITHM: POP (STACK, TOP, ITEM)

This procedure deletes the top element of STACK and assigns it to the variable ITEM.

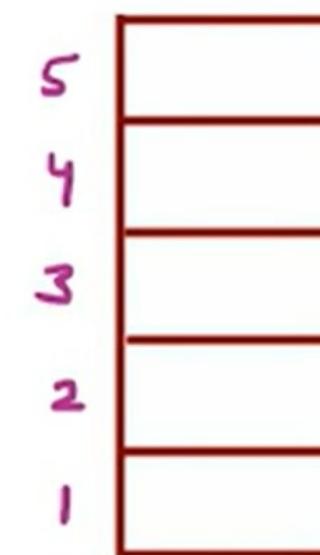
- 1. If TOP = 0, then: Print UNDERFLOW, and Return.
- 2. Set ITEM := STACK [TOP].
- 3. Set TOP := TOP - 1.
- 4. Return.

Case 3

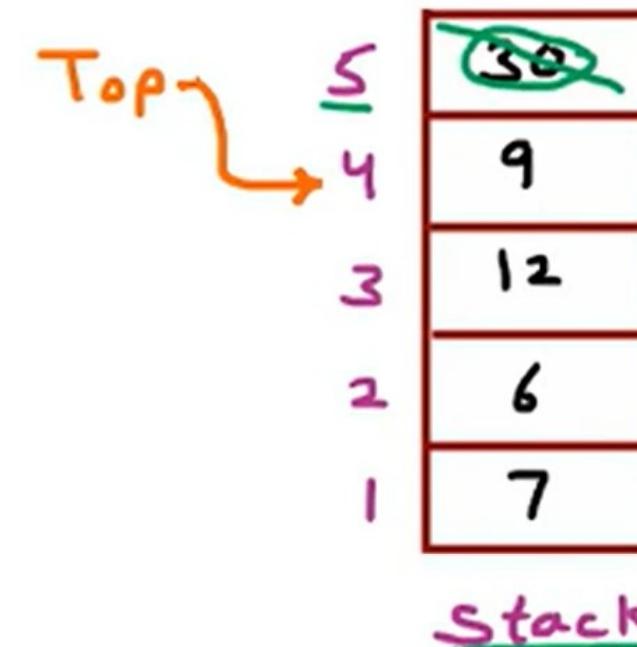


Item  
13

Case 1



Case 2



Item  
30

## Polish Notations

### Infix Notation

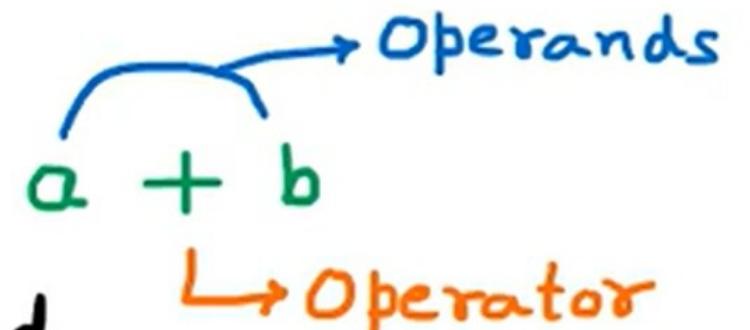
- Operator symbol placed between its two operands
- Arithmetic operations use Infix Notation

### Polish (Prefix) Notation

- Operator symbol placed before its two operands
- Name after Polish Mathematician Jan Lukasiewicz

### Reverse Polish (Postfix)(Suffix) Notation

- Operator symbol placed after its two operands



$a * b$

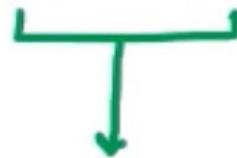
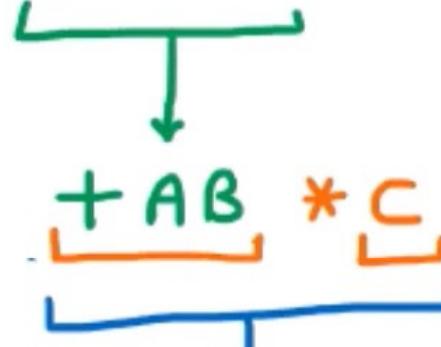
$+ab$

$*ab$

$ab +$

$ab *$

# Convert Infix to Polish Notation

 $a+b$  $+ab$   
(Prefix) $A+B$  $+AB$  $(A+B)*C$  $*+ABC$  $A+(B*C)$  $+A*BC$  $+ab$   
(Prefix) $(A+B)/(C-D)$  $+AB/-CD$  $/+AB-CD$

Convert Infix to Reverse Polish Notation

$a+b$

$$A+B$$

$\downarrow$

$AB+$

$$(A+B)*C$$

$$\underline{AB+} \ * C$$

$\downarrow$

$AB+C *$

$$A+(B*C)$$

$$\underline{A} + \underline{BC*}$$

$\downarrow$

$A BC*+$

$$(A+B) / (C-D)$$

$$\underline{AB+} \ | \ \underline{CD-}$$

$\downarrow$

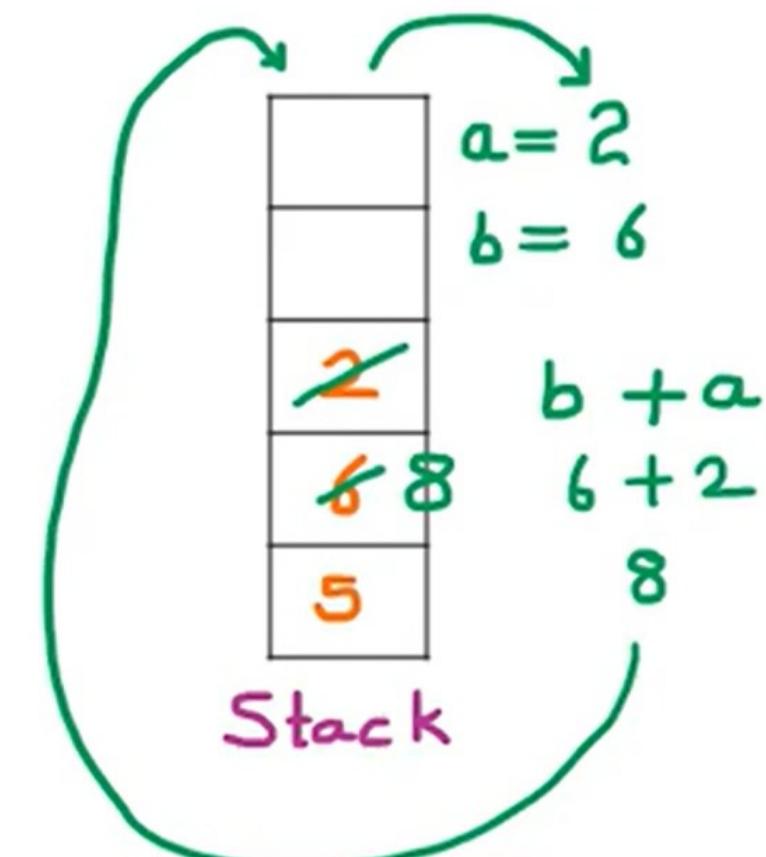
$AB+CD-/$

## Evaluation of Postfix Expression

Evaluate Arithmetic Expression P as Postfix Expression

P:  $\frac{5}{1}, \frac{6}{2}, \frac{2}{3}, +, *, 12, 4, /, -,$   
 $12 \quad 4 \quad 6 \quad 7 \quad 8 \quad 9$

Symbol	Scanned	Stack
1	5	5
2	6	5, 6
3	2	5, 6, 2
4	+	5, 8
5		
6		
7		
8		
9		

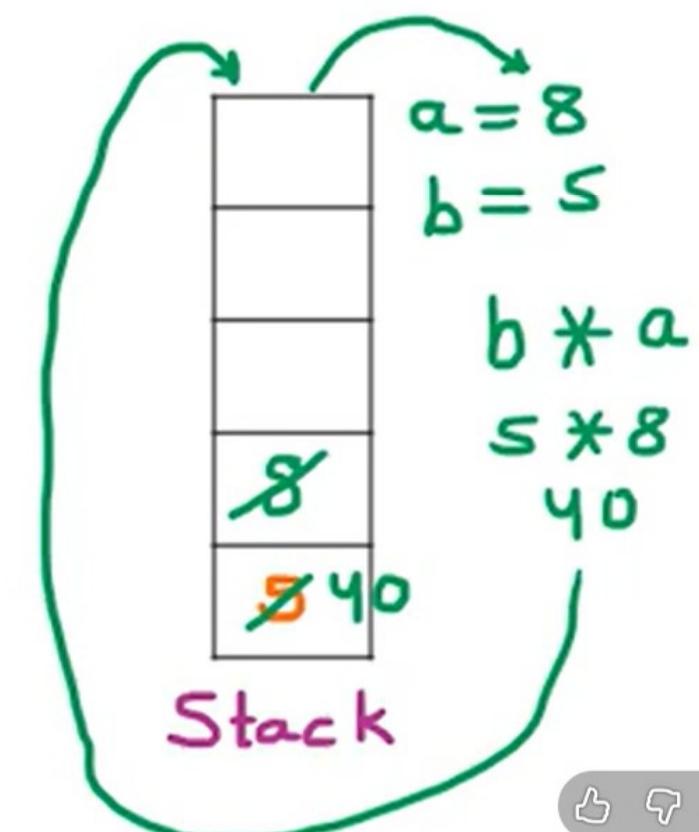


# Evaluation of Postfix Expression

Evaluate Arithmetic Expression P as Postfix Expression

$$P: \frac{5}{1}, \frac{6}{2}, \frac{2}{3}, \frac{+}{4}, \frac{*}{5}, 12, 4, \frac{/}{6}, \frac{-}{7}, 8, 9$$

Symbol	Scanned	Stack
1	5	5
2	6	5, 6
3	2	5, 6, 2
4	+	5, 8
5	*	40
6		
7		
8		
9		

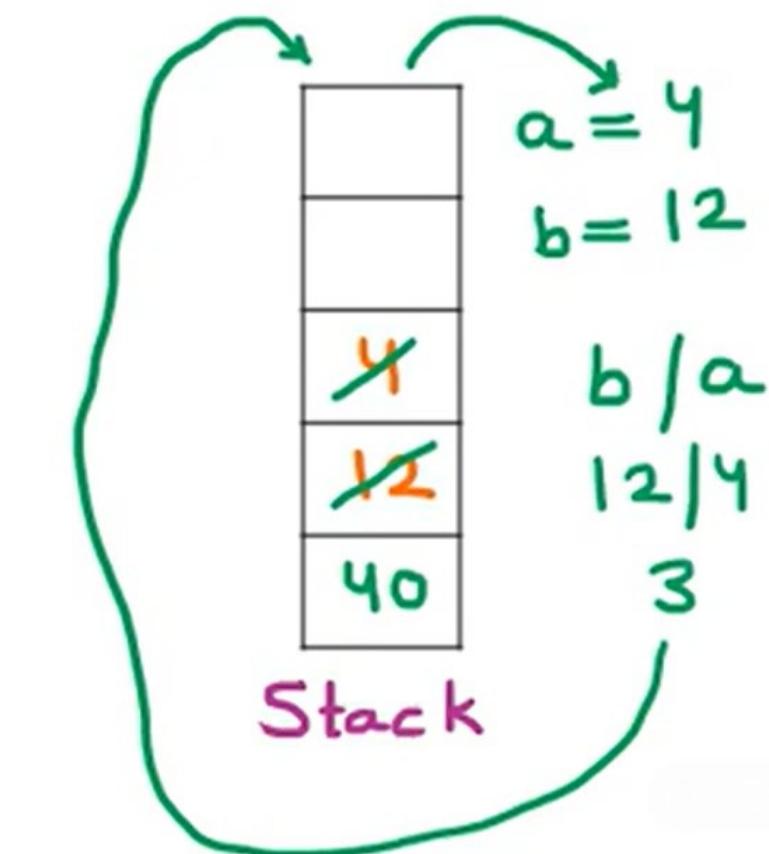


## Evaluation of Postfix Expression

Evaluate Arithmetic Expression P as Postfix Expression

$$P: \frac{5}{1}, \frac{6}{2}, \frac{2}{3}, \frac{+}{4}, \frac{*}{5}, \frac{12}{6}, \frac{4}{7}, \frac{/}{8}, \frac{-}{9}$$

Symbol	Scanned	Stack
1	5	5
2	6	5, 6
3	2	5, 6, 2
4	+	5, 8
5	*	40
6	12	40, 12
7	4	40, 12, 4
8	/	
9		

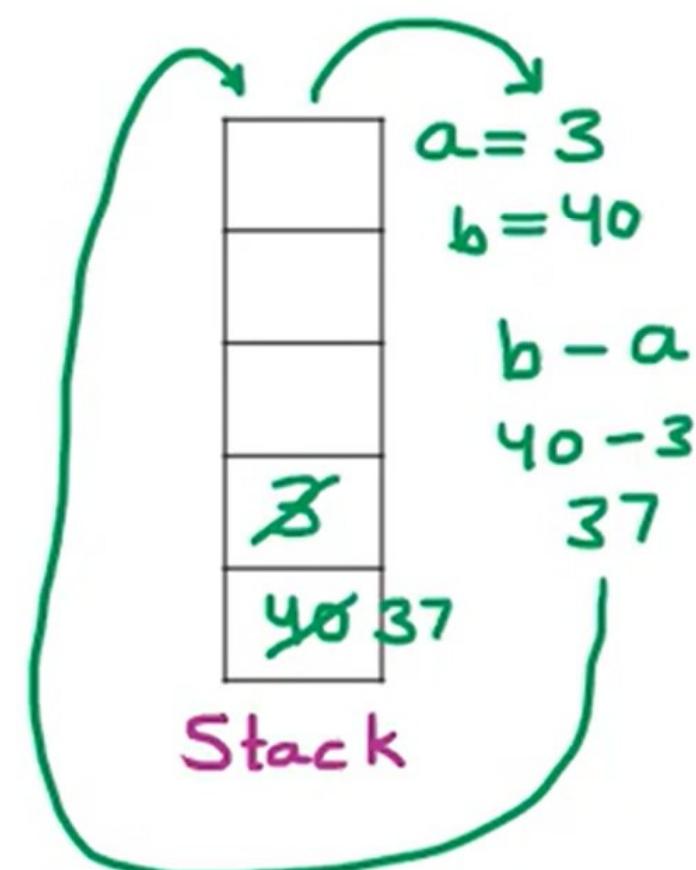


## Evaluation of Postfix Expression

Evaluate Arithmetic Expression P as Postfix Expression

$$P: \frac{5}{1}, \frac{6}{2}, \frac{2}{3}, \frac{+}{4}, \frac{*}{5}, \frac{12}{6}, \frac{4}{7}, \frac{/}{8}, \frac{-}{9}$$

Symbol	Scanned	Stack
1	5	5
2	6	5, 6
3	2	5, 6, 2
4	+	5, 8
5	*	40
6	12	40, 12
7	4	40, 12, 4
8	/	40, 3
9	-	3



## Evaluation of Postfix Expression

Evaluate Arithmetic Expression P as Postfix Expression

$$P: \frac{5}{1}, \frac{6}{2}, \frac{2}{3}, \frac{+}{4}, \frac{*}{5}, \frac{12}{6}, \frac{4}{7}, \frac{/}{8}, \frac{-}{9}$$

Symbol ✓	Scanned ✓	Stack
1	5	5
2	6	5, 6
3	2	5, 6, 2
4	+	5, 8
5	*	40
6	12	40, 12
7	4	40, 12, 4
8	/	40, 3
9	-	37



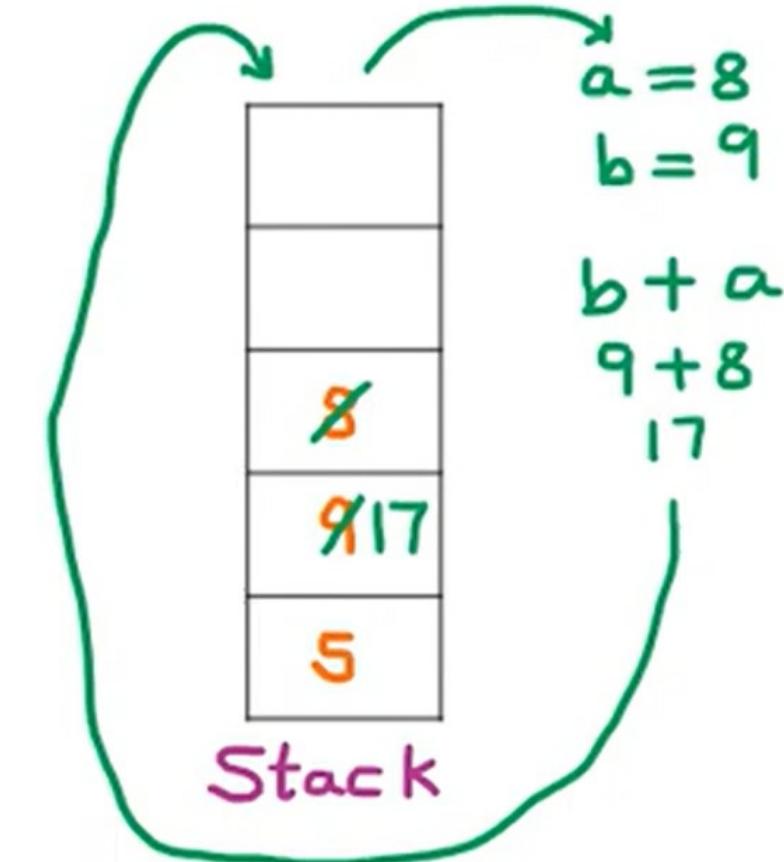
Stack

# Evaluation of Postfix Expression

Evaluate Arithmetic Expression P as Postfix Expression

P:  $\underline{5}, \underline{9}, \underline{8}, \underline{+}, \underline{4}, \underline{6}, \underline{\ast}, \underline{+}, \underline{7}, \underline{-}, \underline{\ast}, \underline{)}$

Symbol	Scanned	Stack
1	5	5
2	9	5, 9
3	8	5, 9, 8
4	+	5, 17
5		
6		
7		
8		
9		
10		
11		
12		

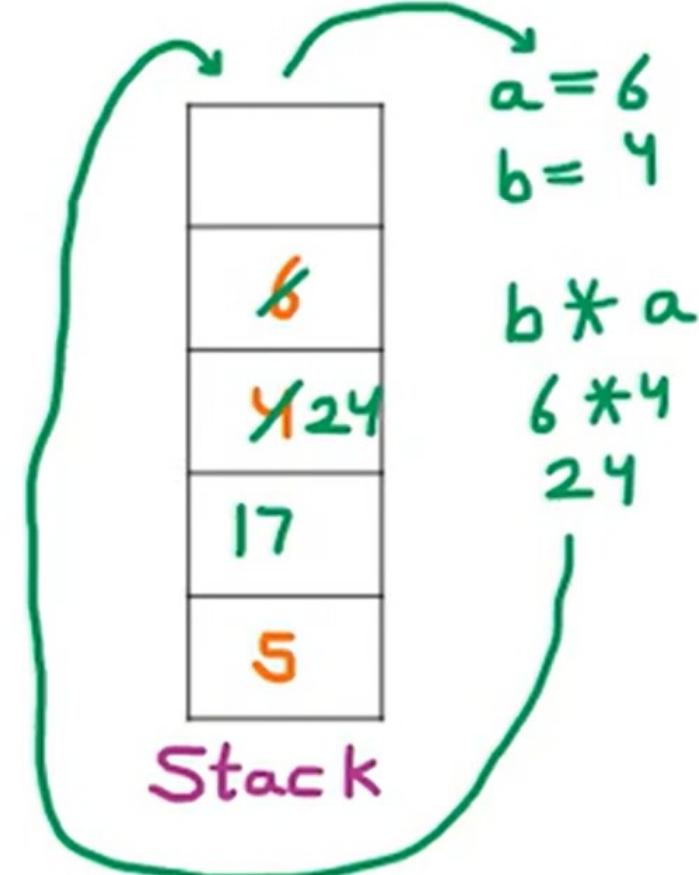


# Evaluation of Postfix Expression

Evaluate Arithmetic Expression P as Postfix Expression

P:  $\frac{5}{1}, \frac{9}{2}, \frac{8}{3}, +, \frac{4}{4}, \frac{6}{5}, \frac{*}{6}, \frac{+}{7}, \frac{7}{8}, \frac{-}{9}, \frac{*}{10}, \frac{)}{11}, \frac{12}{12}$

Symbol	Scanned	Stack
1	5	5
2	9	5, 9
3	8	5, 9, 8
4	+	5, 17
5	4	5, 17, 4
6	6	5, 17, 4, 6
7	*	5, 17, 24
8		
9		
10		
11		
12		

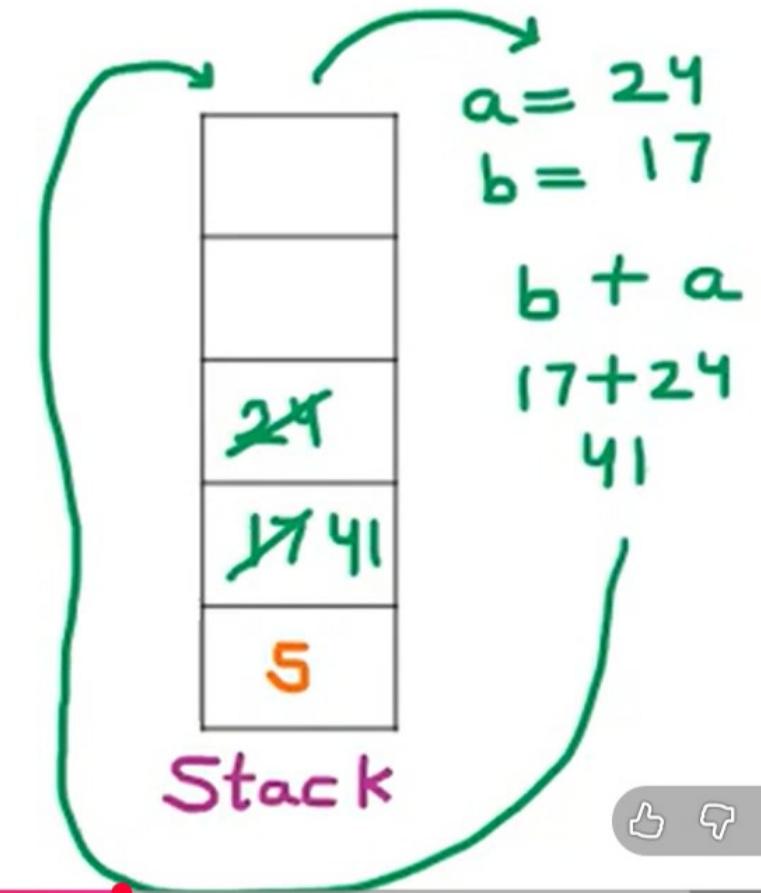


## Evaluation of Postfix Expression

Evaluate Arithmetic Expression P as Postfix Expression

P:  $\frac{5}{1}, \frac{9}{2}, \frac{8}{3}, +, \frac{4}{4}, \frac{6}{5}, *, +, \frac{7}{7}, \frac{9}{8}, -, *, \frac{10}{9}, \frac{11}{10}, \frac{12}{11}$

Symbol	Scanned	Stack
1	5	5
2	9	5, 9
3	8	5, 9, 8
4	+	5, 17
5	4	5, 17, 4
6	6	5, 17, 4, 6
7	*	5, 17, 24
8	+	5, 41
9		
10		
11		

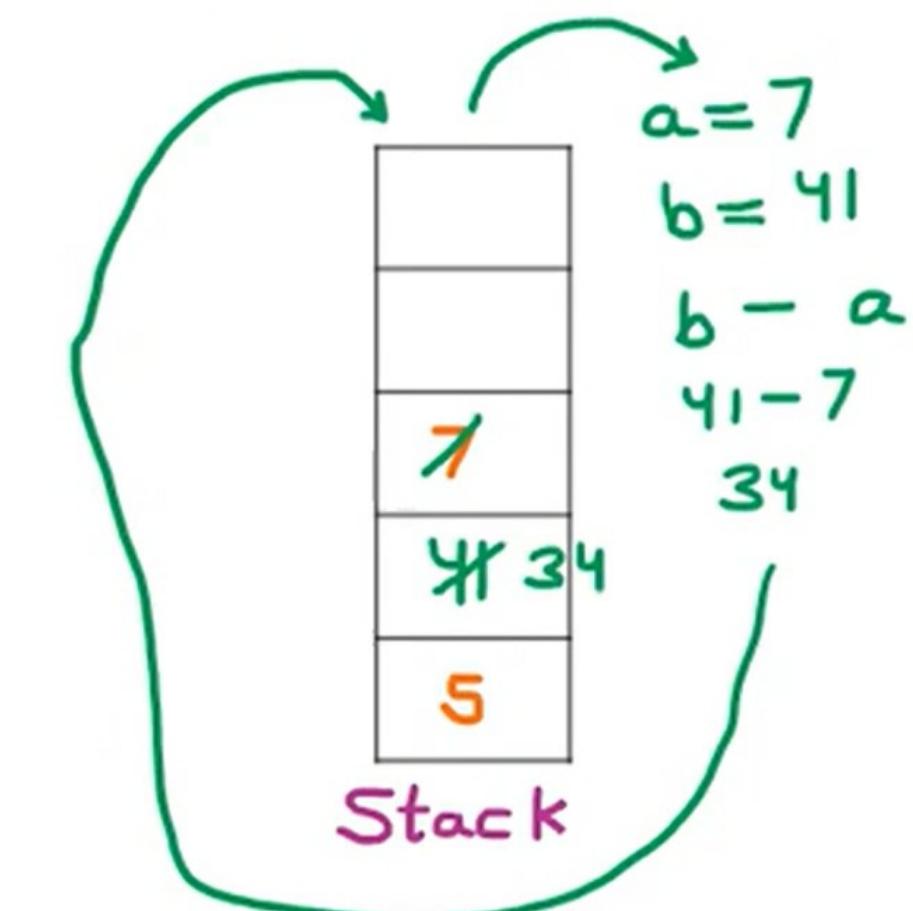


# Evaluation of Postfix Expression

Evaluate Arithmetic Expression P as Postfix Expression

P:  $\frac{5}{1}, \frac{9}{2}, \frac{8}{3}, +, \frac{4}{4}, \frac{6}{5}, *, +, \frac{7}{8}, \frac{-}{9}, \frac{*}{10}, \frac{11}{11}, \frac{*}{12}$

Symbol	Scanned	Stack
1	5	5
2	9	5, 9
3	8	5, 9, 8
4	+	5, 17
5	4	5, 17, 4
6	6	5, 17, 4, 6
7	*	5, 17, 24
8	+	5, 41
9	7	5, 41, 7
10	-	5, 34
11		
12		

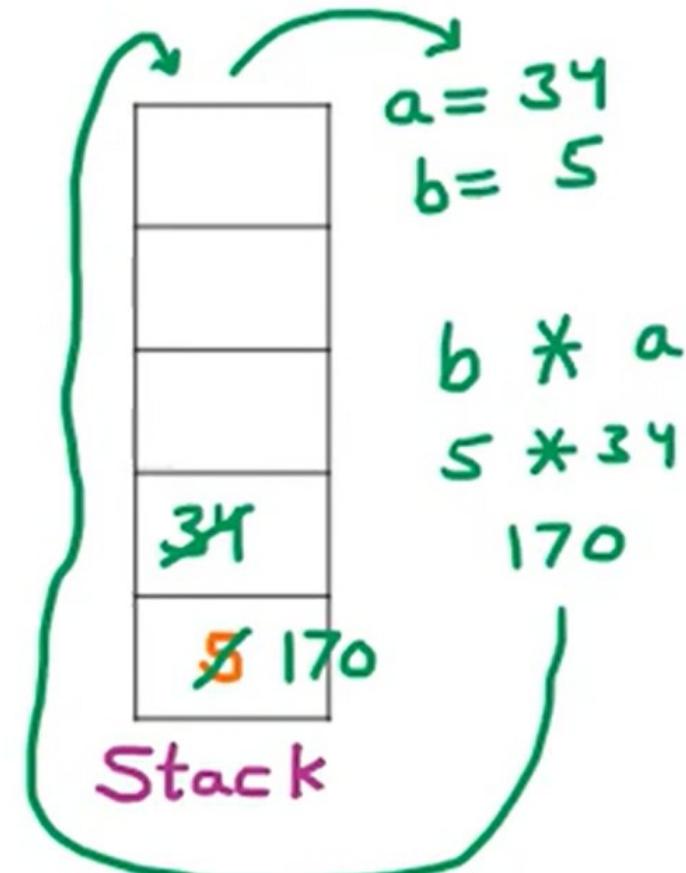


# Evaluation of Postfix Expression

Evaluate Arithmetic Expression P as Postfix Expression

P:  $\frac{5}{1}, \frac{9}{2}, \frac{8}{3}, +, \frac{4}{4}, \frac{6}{5}, *, +, \frac{7}{8}, \frac{-}{9}, \frac{*}{10}, \frac{34}{11}, \frac{5}{12}$

Symbol	Scanned	Stack
1	5	5
2	9	5, 9
3	8	5, 9, 8
4	+	5, 17
5	4	5, 17, 4
6	6	5, 17, 4, 6
7	*	5, 17, 24
8	+	5, 41
9	7	5, 41, 7
10	-	5, 34
11	*	170
12		



## Evaluation of Postfix Expression

Evaluate Arithmetic Expression P as Postfix Expression

P:  $\frac{5}{1}, \frac{9}{2}, \frac{8}{3}, +, \frac{4}{4}, \frac{6}{5}, *, +, \frac{7}{8}, \frac{-}{9}, \frac{*}{10}, \frac{)}{11}, \frac{)}{12}$

Symbol	Scanned	Stack
1	5	5
2	9	5, 9
3	8	5, 9, 8
4	+	5, 17
5	4	5, 17, 4
6	6	5, 17, 4, 6
7	*	5, 17, 24
8	+	5, 41
9	7	5, 41, 7
10	-	5, 34
11	*	170
12	)	

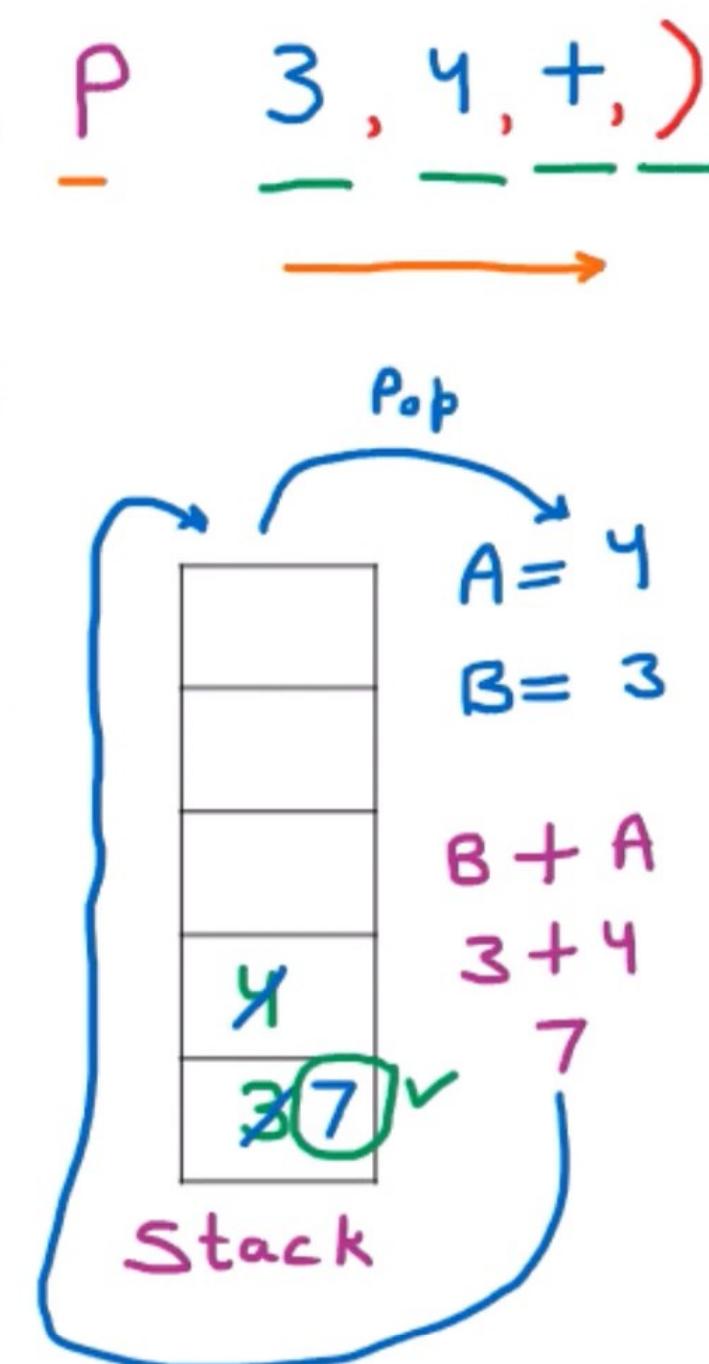


Stack

## ALGORITHM: Evaluation of Postfix Expression

This Algorithm finds the VALUE of an arithmetic expression P written in postfix notation.

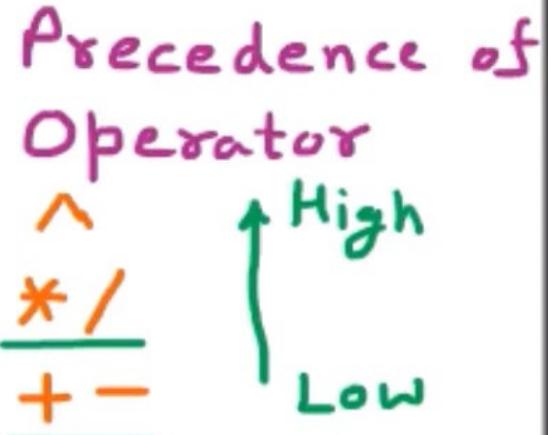
1. Add a right parenthesis ")" at the end of P.
2. Scan P from left to right and repeat Steps 3 & 4 for each element of P until the sentinel ")" is encountered.
3. If an operand is encountered, put in on STACK.
4. If an operator  $\Theta$  is encountered then:
  - a) Remove the two top elements of STACK, where A is top element and B is next-to-top element.
  - b) Evaluate  $B \Theta A$ .
  - c) Place the result of (b) back on STACK.
5. Set VALUE equal to the top element on STACK.
6. Exit



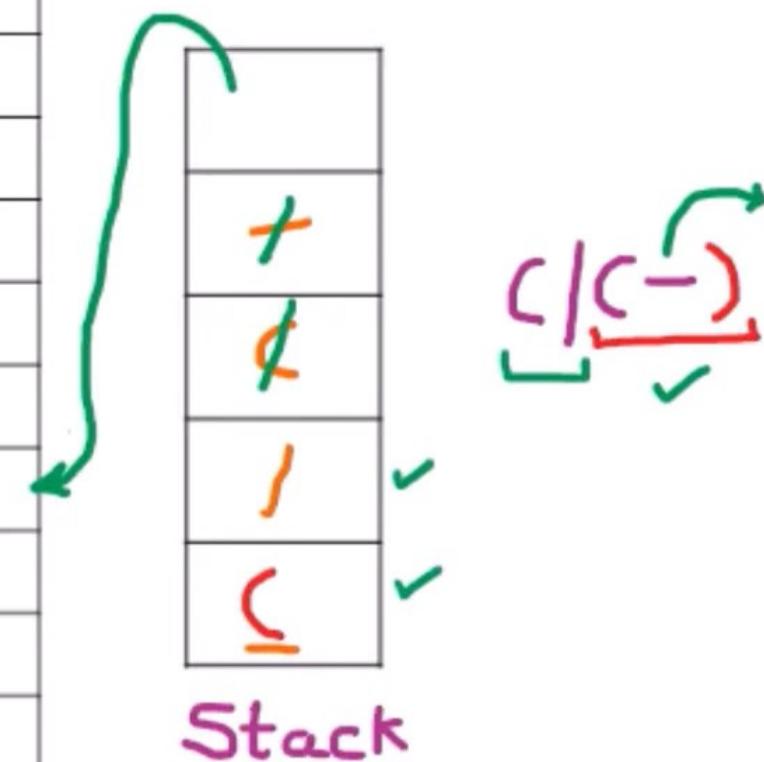
# Infix to Postfix Conversion using Stack

Transform Q into equivalent Postfix Expression

Q:  $A \frac{1}{I} (B \frac{-}{C} \frac{D}{E}) * D + E$



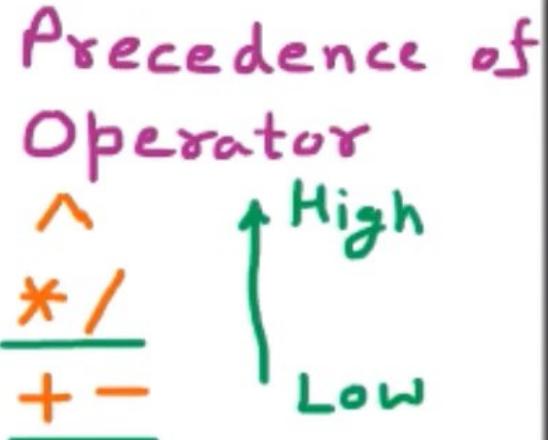
Symbol	Scanned	Stack	Expression
1	A	(	A
2	/	(/	A
3	(	(/(C	A
4	B	(/(C	AB
5	-	(/(C-	AB
6	C	(/(C-	ABC
7	)	C)	ABC-
8			
9			
10			
11			
12			



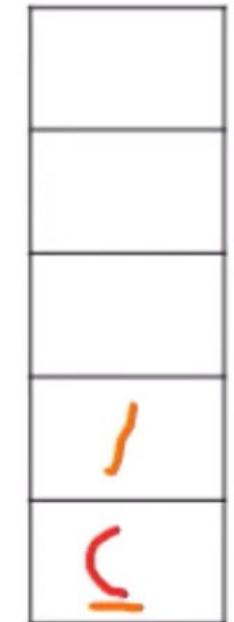
# Infix to Postfix Conversion using Stack

Transform Q into equivalent Postfix Expression

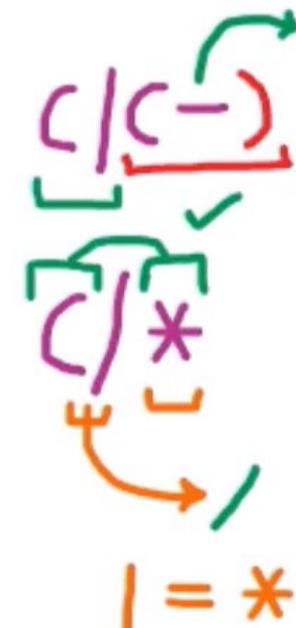
Q:  $A \frac{1}{I} (B \frac{2}{-} C \frac{3}{5} ) \frac{4}{*} D \frac{6}{+} E \frac{7}{8} \frac{9}{9} \frac{10}{10} \frac{11}{11} \frac{12}{12}$



Symbol	Scanned	Stack	Expression
1	A	(	A
2	/	(/	A
3	(	(/()	A
4	B	(/()C	AB
5	-	(/()C-	AB
6	C	(/()C-	ABC
7	)	(/)	ABC-
8	*	(*	ABC-/
9			
10			
11			
12			



Stack



| = \*

# Infix to Postfix Conversion using Stack

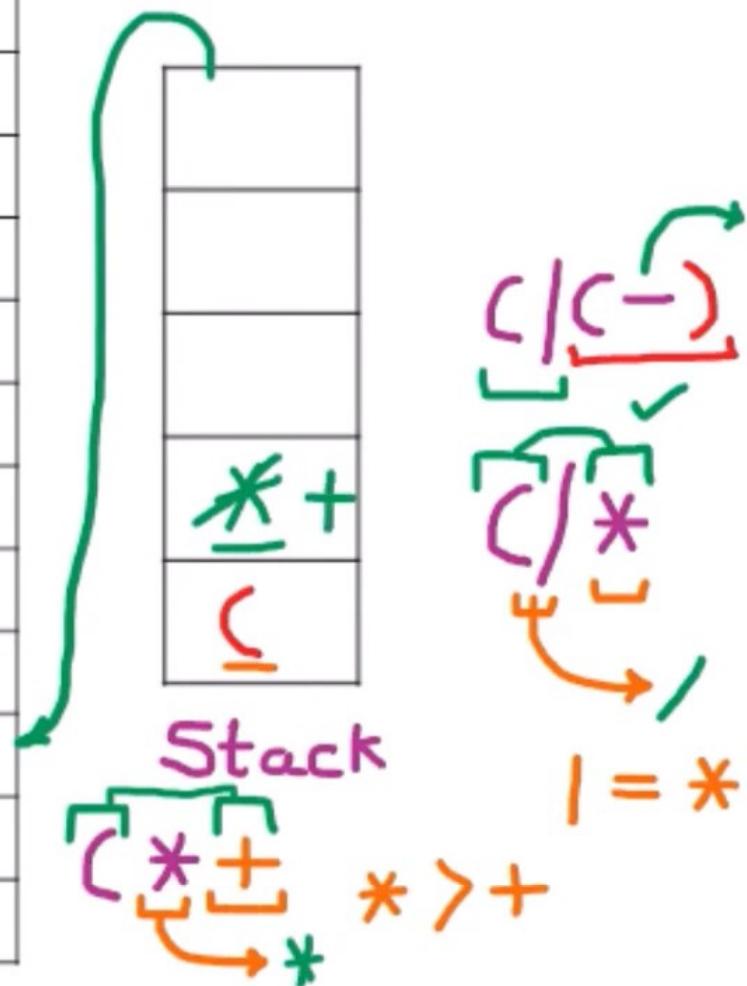
Transform Q into equivalent Postfix Expression  
Q: A | (B = C) \* D + E

$$Q: \frac{A}{1} \frac{1}{2} \frac{(B-C)}{3} \frac{5}{4} \frac{*}{6} \frac{D}{7} \frac{+}{8} \frac{E}{9} \frac{10}{11} \frac{12}{13}$$

Precedence of Operator

^	High	
*	/	
+	-	Low

<u>Symbol</u>	<u>Scanned</u>	Stack	Expression
1	A	(	A
2	/	(/	A
3	(	(/(	A
4	B	(/(C	AB
5	-	(/(-	AB
6	C	(/(-	ABC
7	)	<u>C/</u>	ABC-
8	*	<u>C*</u>	ABC-/
9	D	<u>C*</u>	ABC-/D
10	+	<u>(+)</u>	ABC-/D*
11			
12			



# Infix to Postfix Conversion using Stack

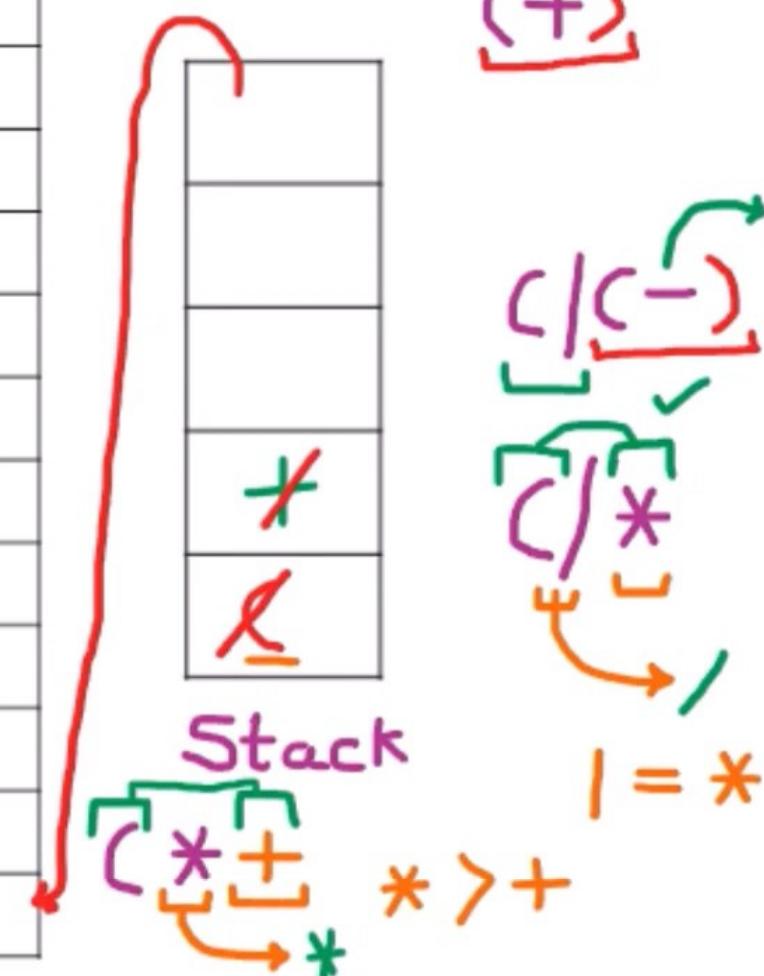
Transform Q into equivalent Postfix Expression

Q:  $A \frac{1}{1} | (B \frac{2}{-} C \frac{3}{5} ) \frac{4}{8} * D \frac{6}{9} + E \frac{7}{10} \frac{11}{12}$

Precedence of Operator

$^$	High
$*$	
$/$	
$+$	Low
$-$	

Symbol	Scanned	Stack	Expression
1	A	(	A
2	/	(/	A
3	(	(/()	A
4	B	(/()C	AB
5	-	(/()C-	AB
6	C	(/()C-	ABC
7	)	(/)	ABC-
8	*	(*	ABC-/
9	D	(*	ABC-/D
10	+	(+	ABC-/D*
11	E	(+)	ABC-/D*D+E
12	)		ABC-/D*D+E+

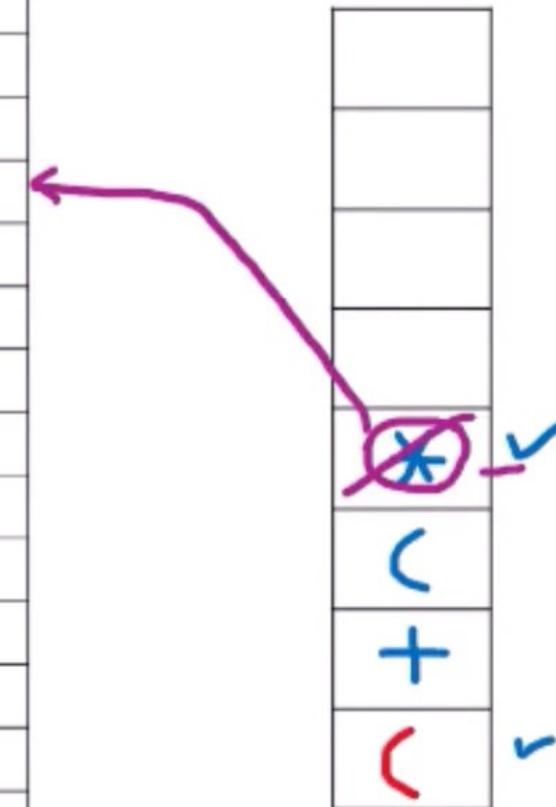


Infix

$$\frac{A}{1} + \left( \frac{B}{2} \times \frac{C}{5} \right) = \frac{D}{7} / \left( E \wedge F \right) * G * H$$

8 9 10 11 12 13 14 15 16 17 18 19 20

Symbol	Scanned	Stack	Expression
1	A	C	A
2	+	(+	A
3	(	(+(	A
4	B	(+(C	AB
5	*	(+(C*	AB
6	C	(+(C*	ABC
7	-	(+(C-	ABC*
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			



Stack

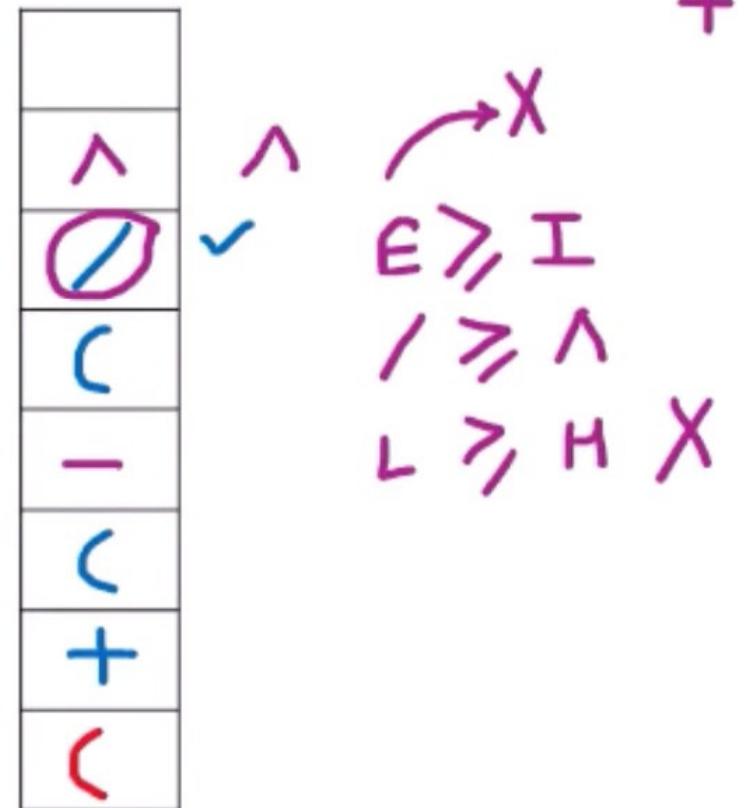


Infix

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

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

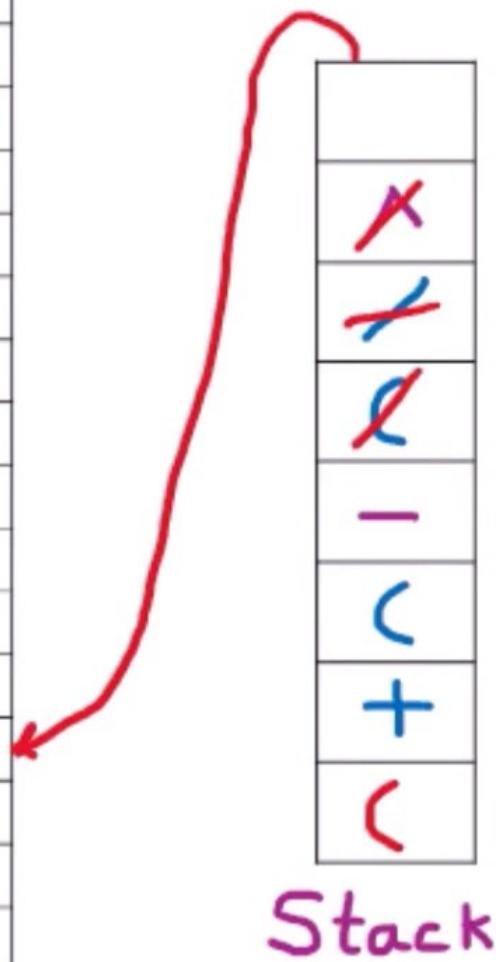
Symbol	Scanned	Stack	Expression
1	A	C	A
2	+	(+	A
3	(	(+(	A
4	B	(+(C	AB
5	*	(+(C*	AB
6	C	(+(C*	ABC
7	-	(+(C-	ABC*
8	(	(+(C-(	ABC*
9	D	(+(C-(C	ABC*D
10	/	(+(C-(C)	ABC*D
11	E	(+(C-(C/	ABC*DDE
12	^	(+(C-(C/	ABC*DDE
13			
14			
15			
16			
17			
18			
19			
20			



Stack

Infix  $A + (B * C - D / E ^ F) * G * H$

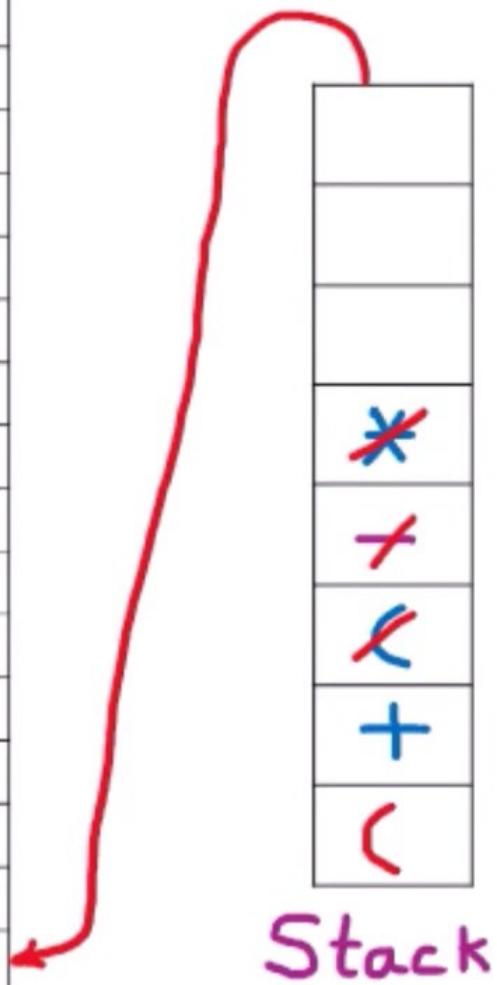
Symbol	Scanned	Stack	Expression
1	A	C	A
2	+	(+	A
3	(	(+(C	A
4	B	(+(C	AB
5	*	(+(C*	AB
6	C	(+(C*	ABC
7	-	(+(C-	ABC*
8	(	(+(C-(C	ABC*
9	D	(+(C-(C	ABC*D
10	/	(+(C-(C)	ABC*D
11	E	(+(C-(C/	ABC*DE
12	^	(+(C-(C/^	ABC*DE
13	F	(+(C-(C/^	ABC*DEF
14	)	(+(C-	ABC*DEF^/
15			
16			
17			
18			
19			
20			



$\wedge$   
 $*$  /  
+ -

Infix  $A \frac{1}{2} + \frac{3}{4} \times \frac{5}{6} - \left( \frac{D}{7} \frac{8}{9} \frac{E}{10} \frac{11}{12} \frac{F}{13} \frac{14}{15} \frac{G}{16} \frac{17}{18} \frac{H}{19} \right) * \frac{1}{20}$

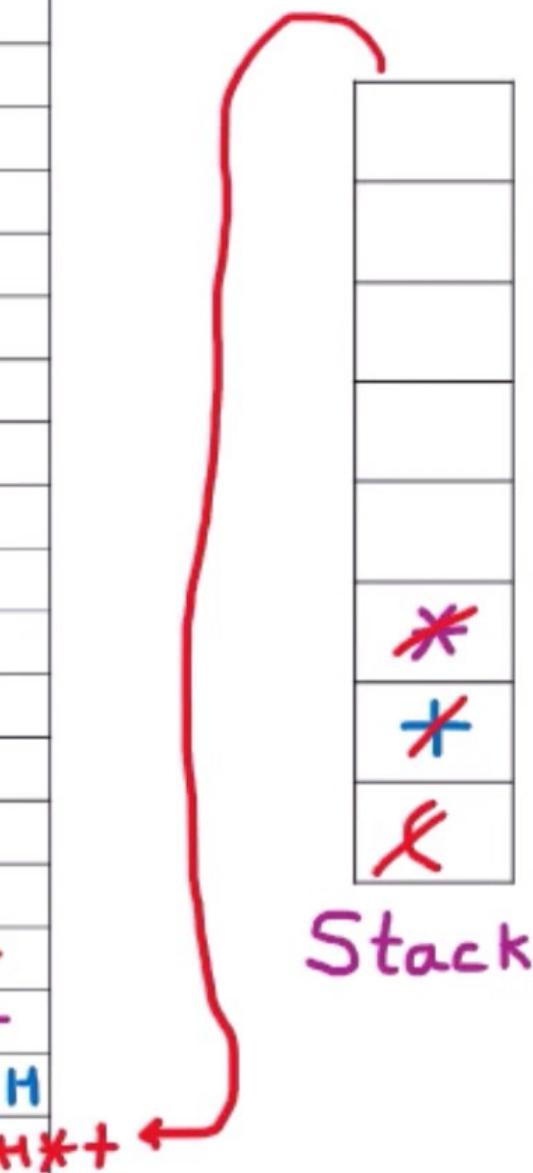
Symbol	Scanned	Stack	Expression
1	A	C	A
2	+	(+	A
3	(	(+(C	A
4	B	(+(C	AB
5	*	(+(C*	AB
6	C	(+(C*	ABC
7	-	(+(C-	ABC*
8	(	(+(C-(	ABC*
9	D	(+(C-(C	ABC*D
10	/	(+(C-(C)	ABC*D
11	E	(+(C-(C/	ABC*DE
12	^	(+(C-(C/	ABC*DE
13	F	(+(C-(C/	ABC*DEF
14	)	(+(C-	ABC*DEF^/
15	*	(+(C-*	ABC*DEF^/
16	G	(+(C-*	ABC*DEF^/G
17	)	(+	ABC*DEF^/G*-
18			
19			
20			



^  
\*/  
+-

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

Symbol	Scanned	Stack	Expression
1	A	C	A
2	+	(+	A
3	(	(+(C	A
4	B	(+(C	AB
5	*	(+(C*	AB
6	C	(+(C*	ABC
7	-	(+(C-	ABC*
8	(	(+(C-(	ABC*
9	D	(+(C-(C	ABC*D
10	/	(+(C-(C)	ABC*D
11	E	(+(C-(C/	ABC*DE
12	$\wedge$	(+(C-(C/\wedge	ABC*DE
13	F	(+(C-(C/\wedge	ABC*DEF
14	)	(+(C-	ABC*DEF/\wedge
15	*	(+(C-*	ABC*DEF/\wedge
16	G	(+(C-*	ABC*DEF/\wedge G
17	)	(+	ABC*DEF/\wedge G*-
18	*	(+*	ABC*DEF/\wedge G*-
19	H	(+*	ABC*DEF/\wedge G*-H
20	)		ABC*DEF/\wedge G*-H**+



$\wedge$   
 $*$  /  
+ -

## Infix

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

1    2    3    4    5    6    7    8    9    10    11    12    13    14    15    16    17    18    19    20  
 A    +    (    B    ×    C    )    -    (    D    /    E    ∧    F    )    \*    G    )    \*    H

Symbol	Scanned	Stack	Expression
1	A	(	A
2	+	(+	A
3	(	(+(	A
4	B	(+(C	AB
5	*	(+(C*	AB
6	C	(+(C*	ABC
7	-	(+(C-	ABC*
8	(	(+(C-(	ABC*
9	D	(+(C-C	ABC*D
10	/	(+(C-C)	ABC*D
11	E	(+(C-C/	ABC*D E
12	^	(+(C-C/ ^	ABC*D E
13	F	(+(C-C/ ^	ABC*D E F
14	)	(+(C-	ABC*D E F A /
15	*	(+(C-*	ABC*D E F A /
16	G	(+(C-*	ABC*D E F A / G
17	)	(+	ABC*D E F A / G * -
18	*	(+*	ABC*D E F A / G * -
19	H	(+*	ABC*D E F A / G * - H
20			ABC*D E F A / G * - H * + Postfix



Stack

