Q1)

i. Conversion of **A + ((B – C * D) / E) + F – G / H** to postfix.

Next Token	Effect on Stack	Effect on Postfix
A		A
+	+	A
(	+(	A
(	+((	A
B	+((	A B
-	+((-	A B
C	+((-	A B C
*	+((-*	A B C
D	+((-*	A B C D
)	+(	A B C D * -
/	+(/	A B C D * -
E	+(/	A B C D * - E
)	+	A B C D * - E /
+	+	A B C D * - E / +
F	+	A B C D * - E / + F
-	-	A B C D * - E / + F +
G	-	A B C D * - E / + F + G
/	-/	A B C D * - E / + F + G
H	-/	A B C D * - E / + F + G H
End of Input	-	A B C D * - E / + F + G H /
End of Input and Stack		A B C D * - E / + F + G H / -

Conversion of **A + ((B – C * D) / E) + F – G / H** to prefix.

To do this we need to convert this statement to reverse so we will get;

H / G – F + (E / (D * C – B)) + A

Next Token	Effect on Stack	Effect on Prefix
H		H
/	/	H
G	/	H G
-	-	H G /
F	-	H G / F
+	+	H G / F -

(	+(	H G / F -
E	+(	H G / F - E
/	+(/	H G / F - E
(	+(/(	H G / F - E
D	+(/(	H G / F - E D
*	+(/(*	H G / F - E D
C	+(/(*	H G / F - E D C
-	+(/(-	H G / F - E D C *
B	+(/(-	H G / F - E D C * B
)	+(/	H G / F - E D C * B -
)	+	H G / F - E D C * B - /
+	+	H G / F - E D C * B - / +
A	+	H G / F - E D C * B - / + A
End of Input and Stack		H G / F - E D C * B - / + A +
Reverse Prefix		+ A + / - B * C D E - F / G H

Evaluating postfix expression **A B C D * - E / + F + G H / -**

Expression	**Action**	**Stack**
A B C D * - E / + F + G H / -	Push A,B,C,D	A, B, C, D
* - E / + F + G H / -	Pop C, D evaluate C*D push C*D	A, B, (C*D)
- E / + F + G H / -	Pop C*D, B evaluate B - C*D push B - C*D	A, (B - C*D)
E / + F + G H / -	Push E	A, (B - C*D), E
/ + F + G H / -	Pop (B - C*D), E evaluate (B - C*D)/E push (B - C*D) /E	A, (B - C*D)/E
+ F + G H / -	Pop A, (B - C*D)/E evaluate A + (B - C*D)/E push A + (B - C*D) /E	A + (B - C*D) /E
F + G H / -	Push F	A + (B - C*D) /E, F
+ G H / -	Pop (B - C*D)/E + A, F evaluate (B - C*D)/E + A + F push A + (B - C*D) /E + F	A + (B - C*D) /E + F

G H / -	Push G H	A + (B - C*D) /E + F , G,H
/ -	Pop G,H evaluate G/H Push G/H	A + (B - C*D) /E + F , G/H
-	Pop A + (B - C*D) /E + F and G/H evaluate A + (B - C*D) /E + F - G/H push A + (B - C*D) /E + F - G/H	A + (B - C*D) /E + F - (G/H)

Evaluating postfix expression **A B C D * - E / + F + G H / -** with numbers

A = 7, **B** =14 , **C** = 4, **D**=1, **E** =5, **F** =3 , **G**=8 , **H**=2

Result of infix: 7 + ((14 – 4 * 1) / 5) + 3 – 8 / 2 **=** 7 + 2 + 3 – 4 **= 8**

Expression	**Action**	**Stack**
7 14 4 1 * - 5 / + 3 + 8 2 / -	Push 7,14,4,1	7, 14, 4, 1
* - 5 / + 3 + 8 2 / -	Pop 4, 1 Evaluate 4*1 Push 4	7, 14, 4
- 5 / + 3 + 8 2 / -	Pop 14, 4 Evaluate 14-4 Push 10	7, 10
5 / + 3 + 8 2 / -	Push 5	7, 10, 5
/ + 3 + 8 2 / -	Pop 10, 5 Evaluate 10/5 Push 2	7, 2
+ 3 + 8 2 / -	Pop 7, 2 Evaluate 7+2 Push 9	9
3 + 8 2 / -	Push 3	9, 3
+ 8 2 / -	Pop 9, 3 Evaluate 9+3 Push 12	12
8 2 / -	Push 8, 2	12, 8, 2

/ -	Pop 8,2 Evaluate 8/2 Push 4	12, 4
-	Pop 12, 4 Evaluate 12-4 Push 8	8

ii. Conversion of **! (A && ! ((B < C) || (C > D))) || (C < E)** to postfix.

Next Token	Effect on Stack	Effect on Postfix
!	!	
(	!(	
A	!(	A
&&	!(&&	A
!	!(&&!	A
(	!(&&!(	A
(	!(&&!((	A
B	!(&&!((	A B
<	!(&&!((<	A B
C	!(&&!((<	A B C
)	!(&&!(	A B C <
||	!(&&!(||	A B C <
(	!(&&!(||(	A B C <
C	!(&&!(||(	A B C < C
>	!(&&!(||(>	A B C < C
D	!(&&!(||(>	A B C < C D
)	!(&&!(||	A B C < C D >
)	!(&&!	A B C < C D > ||
)	!	A B C < CD > || ! &&
||	||	A B C < C D > || ! && !
(	||(	A B C < CD > || ! && !
C	||(	A B C < CD > || ! && ! C
<	||(<	A B C < C D > || ! && ! C
E	||(<	A B C < CD > || ! && ! C E
)	||	A B C < C D > || ! && ! C E <
End of Input	||	A B C < C D > || ! && ! C E <
End of Input and Stack		A B C < C D > || ! && ! C E < ||

Conversion of **! (A && ! ((B < C) || (C > D))) || (C < E)** to prefix.

To do this we need to convert this statement to reverse so we will get;

(E < C) || (((D>C) || (C < B))! && A)!

Next Token	Effect on Stack	Effect on Postfix
(	(	
E	(	E
<	(<	E
C	(<	E C
)		E C <
||	||	E C <
(	||(	E C <
(	||((	E C <
(	||(((	E C <
D	||(((	E C < D
>	||(((>	E C < D
C	||(((>	E C < D C
)	||((	E C < D C >
||	||((||	E C < D C >
(	||((||(	E C < D C >
C	||((||(	E C < D C > C
<	||((||(<	E C < D C > C
B	||((||(<	E C < D C > C B
)	||((||	E C < D C > C B <
)	||(	E C < D C > C B < ||
!	||(!	E C < D C > C B < ||
&&	||(&&	E C < D C > C B < || !
A	||(&&	E C < D C > C B < || ! A
)	||	E C < D C > C B < || ! A &&
!	||!	E C < D C > C B < || ! A &&
End of Input	||	E C < D C > C B < || ! A && !
End of Input and Stack		E C < D C > C B < || ! A && ! ||
Reverse of the statement		|| ! && A ! || < B C > C D < C E

Evaluating postfix expression **A B C < C D > || ! && ! C E < ||**

Expression	Action	Stack
A B C < C D > || ! && ! C E < ||	Push A,B,C	A B C
< C D > || ! && ! C E < ||	Pop B, C evaluate B < C push B < C	A B < C
C D > || ! && ! C E < ||	Push C, D	A B < C C D
> || ! && ! C E < ||	Pop C, D evaluate C > D push C > D	A B < C C > D
|| ! && ! C E < ||	Pop B < C, C > D evaluate B < C || C > D push B < C || C > D	A B < C || C > D
! && ! C E < ||	Pop B < C || C > D evaluate !(B < C || C > D) push !(B < C || C > D)	A !(B < C || C > D)
&& ! C E < ||	Pop A, !(B < C || C > D) evaluate A && !(B < C || C > D) push A && !(B < C || C > D)	A && !(B < C || C > D)
! C E < ||	Pop A && !(B < C || C > D) evaluate !(A && !(B < C || C > D)) push !(A && !(B < C || C > D))	!(A && !(B < C || C > D))
C E < ||	Push C, E	!(A && !(B < C || C > D)) C E
< ||	Pop C E evaluate C < E push C < E	!(A && !(B < C || C > D)) C < E
||	Pop !(A && !(B < C || C > D)), C < E evaluate !(A && !(B < C || C > D)) || C < E push !(A && !(B < C || C > D)) || C < E	!(A && !(B < C || C > D)) || C < E

Evaluating postfix expression **A B C < C D > || ! && ! C E < ||** with numbers

A = 7, **B** =2 , **C** = 3, **D**=4, **E** =5

Result of infix: ! (7 && ! ((2 < 3) || (3 > 4))) || (3 < 5)

= !(7 && 0) || 0 = !0 || 0 = 1 || 0 = **1**

Expression	Action	Stack
7 2 3 < 3 4 > || ! && ! 3 5 < ||	Push 7,2,3	7 2 3
< 3 4 > || ! && ! 3 5 < ||	Pop 2, 3 Evaluate 2 < 3 Push 0	7 0
3 4 > || ! && ! 3 5 < ||	Push 3, 4	7 0 3 4
> || ! && ! 3 5 < ||	Pop 3, 4 Evaluate 3 > 4 Push 0	7 0 0
|| ! && ! 3 5 < ||	Pop 0, 0 Evaluate 0 || 0 Push 0	7 0
! && ! 3 5 < ||	Pop 0 Evaluate !0 Push 1	7 1
&& ! 3 5 < ||	Pop 7, 1 Evaluate 7 && 1 Push 7	7
! 3 5 < ||	Pop 7 Evaluate !7 Push !7	!7
3 5 < ||	Push 3, 5	!7 3 5
< ||	Pop 3 5 Evaluate 3 < 5 Push 1	!7 1
||	Pop !7, 1 Evaluate !7 || 1 Push 1	1