**Stack**

LIFO - Last In First Out

FILO - First In Last Out

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| 0 | 1 | 2 | 3 | 4 |

Top = -1[shows the stack is empty]

It has three operations push,pop,display (insertion,delete,displaying data)

Insertion

Push



Delete

Pop

Displaying Data

Display

|  |  |
| --- | --- |
| **Algorithm**  increment top  st(top) = value  st(top) = value | **Program of push ( Function part):**  Void Push(val)  { if top ==size-1 { Print stack is full/ overflow } else { top++; st[top] = val; } } |

Let’s consider a stack of size 5 to see the following operations.

**Push Operation**:-

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Top = -1  push(100) push(200) push(300) push(400) push(500) | | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  | 500 |
|  |  |  |  |  |  |  |  | 400 | top | 400 |
|  |  |  |  |  |  | 300 | top | 300 |  | 300 |
|  |  |  |  | 200 | top | 200 |  | 200 |  | 200 |
|  |  | 100 | top | 100 |  | 100 |  | 100 |  | 100 |

top  
 Top = -1 top=0 top=1 top=2 top=3 top=4

In each step, top is increased by 1 and finally it reached the (size -1)th position. Now if we try to insert another element it says stack is full.

**Pop Operation:-**

|  |  |
| --- | --- |
| **Algorithm**  Decrement top  When top != -1  Then we can delete | **Program for pop ( Function part):-** int pop() { if top == -1 { Print stack is empty/ underflow } else { val = st[top]; st[top--] = 0; return val; } } |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| pop() pop() pop() pop() pop() top=-1 | | | | | | | | | | |
| 500 | top |  |  |  |  |  |  |  |  |  |
| 400 |  | 400 | top |  |  |  |  |  |  |  |
| 300 |  | 300 |  | 300 | top |  |  |  |  |  |
| 200 |  | 200 |  | 200 |  | 200 | top |  |  |  |
| 100 |  | 100 |  | 100 |  | 100 |  | 100 | top |  |

top = 4 top = 3 top = 2 top = 1 top = 0 top = -1

In each step, top is decreased by one and finally it has cleared all the elements in the stack and made the stack as empty. It is impossible to delete an element from the empty stack.

**Display Operation :-**

|  |  |
| --- | --- |
| **Algorithm**  If top == -1  Print “ Stack is empty/underflow”  Otherwise run a loop print the elements in the stack from the positon of top to 0 | **Program of display ( Function part ):**  void display()  { int i;  if(top == -1)  { printf( “ Stack is empty/underflow”\n) ; }  else  { for(i = top; i>=0 ; i--)  { printf(“%d”,st[i]); }  printf( “\n”); }  } |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| display() display() display() display() display() top=-1 | | | | | | | | | | |
| Top = 4 500 | i | Top = 4 500 |  | Top = 4 500 |  | Top = 4 500 |  | Top = 4 500underfllow |  |  |
| 400 |  | 400 | i | 400 |  | 400 |  | 400 |  |  |
| 300 |  | 300 |  | 300 | i | 300 |  | 300 |  |  |
| 200 |  | 200 |  | 200 |  | 200 | i | 200 |  |  |
| 100 |  | 100 |  | 100 |  | 100 |  | 100 | i |  |

**Output of display is**

In this case,

It prints stack is empty

500 400 300 200 100

**Different Types of Notations**:-

Infix a+b

Postfix ab+

Prefix +ab

4

3

2

1

0

INFIX to POSTFIX ALGORITHM:-

Infix = a+b\*c-d/e+f

Postfix =

i++ loop

if i is operand:

append it to postfix

if i is operator:

if stack is empty:

push(i)

else if precedence(i) > precedence(st[top]):

push(i)

else

op = pop()

append it to postfix

continue loop until>

t

**^ 🡪 3**

**\* , / 🡪 2**

**+,- 🡪 1**

Example:-

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | \* | b | - | c | / | d | ^ | e | + | f |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| i |  |  |  |  |  |  |  |  |  |  |

Pre(a) = false j=a j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 |  |  |

Top = -1

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| j |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | \* | b | - | c | / | d | ^ | e | + | f |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  | i |  |  |  |  |  |  |  |  |  |

Pre(\*) =2 top++ top=\*

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | \* | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  | j |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | \* | b | - | c | / | d | ^ | e | + | f |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  | i |  |  |  |  |  |  |  |  |

Pre(b) = false j=b j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | \* | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  | j |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | \* | b | - | c | / | d | ^ | e | + | f |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  | i |  |  |  |  |  |  |  |

Pre(-) = 1 1<2 top-- - and j = \* ,j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | - | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | \* | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  | j |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | \* | b | - | c | / | d | ^ | e | + | f |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  | i |  |  |  |  |  |  |

Pre(c) = false j = c j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | - | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | \* | c | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  | j |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | \* | b | - | c | / | d | ^ | e | + | f |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  | i |  |  |  |  |  |

Pre(/) = 2,true 2>1 ++top top = /

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 | / | top |
| 0 | - |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | \* | c | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  | j |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | \* | b | - | c | / | d | ^ | e | + | f |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  | i |  |  |  |  |

Pre(d) = false j=d j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 | / | top |
| 0 | - |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | \* | c | d | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  | j |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | \* | b | - | c | / | d | ^ | e | + | f |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  | i |  |  |  |

Pre(^) = 3 3>2 ++top top = ^

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 | ^ | top |
| 1 | / |  |
| 0 | - |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | \* | c | d | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  | j |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | \* | b | - | c | / | d | ^ | e | + | f |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  | i |  |  |

Pre(e) = false j = e j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 | ^ | top |
| 1 | / |  |
| 0 | - |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | \* | c | d | e | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  | j |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | \* | b | - | c | / | d | ^ | e | + | f |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  |  | i |  |

Pre(+) = 1 1<3 & top != -1 🡪 replace ^ by + & write ^ in postfix

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 | + | top |
| 1 | / |  |
| 0 | - |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | \* | c | d | e | ^ | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  | j |  |  |  |

Pre(+) = 1 1<2 & top != -1 & top-- 🡪 replace / by + & write / in postfix

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 | + | top |
| 0 | - |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | \* | c | d | e | ^ | / | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  | j |  |  |

Pre(+) = 1 1=1 & top != -1 & top - - 🡪 replace - by + & write - in postfix

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | + | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | \* | c | d | e | ^ | / | - | 9 | 10 |
|  |  |  |  |  |  |  |  |  | j |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | \* | b | - | c | / | d | ^ | e | + | f |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  |  |  | i |

Pre(f) = false j=f j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | + | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | \* | c | d | e | ^ | / | - | f | 10 |
|  |  |  |  |  |  |  |  |  |  | j |

Top = + top != -1 🡪 j=+

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | + | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | \* | c | d | e | ^ | / | - | f | + |
|  |  |  |  |  |  |  |  |  |  | j |

This is the required postfix expression converted from infix.

**If we have open braces in our infix then we have to apply the following changes in our program:-**

Precedence of ‘(‘ is -1

If i==’)’ :

Pop all operator upto open bracket and append it to postfix

If i==’)’

op = st[top--]

while(op!=’(‘)

append post op

op = st[top -- ]

Example :- **a+(b\*c+d)-e**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | + | ( | b | \* | c | + | d | ) | - | e |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| i |  |  |  |  |  |  |  |  |  |  |

Pre(a) = false j=a j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 |  |  |

Top = -1

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| j |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | + | ( | b | \* | c | + | d | ) | - | e |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  | i |  |  |  |  |  |  |  |  |  |

Pre(+) = 1 j=a j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | + | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  | j |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | + | ( | b | \* | c | + | d | ) | - | e |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  | i |  |  |  |  |  |  |  |  |

Pre( ‘(‘ ) = -1 infix[i] = ‘(‘ ++ top top = (

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 | ( | top |
| 0 | + |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  | j |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | + | ( | b | \* | c | + | d | ) | - | e |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  | i |  |  |  |  |  |  |  |

Pre( b ) = false j = b j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 | ( | top |
| 0 | + |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  | j |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | + | ( | b | \* | c | + | d | ) | - | e |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  | i |  |  |  |  |  |  |

Pre( \* ) = 2 2 > -1 ++ top top = \*

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 | \* | top |
| 1 | ( |  |
| 0 | + |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  | j |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | + | ( | b | \* | c | + | d | ) | - | e |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  | i |  |  |  |  |  |

Pre( c ) = false j = c j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 | \* | top |
| 1 | ( |  |
| 0 | + |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | c | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  | j |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | + | ( | b | \* | c | + | d | ) | - | e |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  | i |  |  |  |  |

Pre(+) = 1 1<2 & top != -1 & top - - 🡪 replace \* by + & write \* in postfix then j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 | + | top |
| 1 | ( |  |
| 0 | + |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | c | \* | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  | j |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | + | ( | b | \* | c | + | d | ) | - | e |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  | i |  |  |  |

Pre( d) = false j = d j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 | + | top |
| 1 | ( |  |
| 0 | + |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | c | \* | d | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  | j |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | + | ( | b | \* | c | + | d | ) | - | e |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  | i |  |  |

Pre( ‘)‘ ) = -1 infix[i] = ‘)‘ j = + j++ upto st[top] == ‘(‘ remove ‘(‘ from stack decrese top by 1

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | + | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | c | \* | d | + | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  | j |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | + | ( | b | \* | c | + | d | ) | - | e |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  |  | i |  |

Pre( - ) = 1 1=1 replace + by – j = + j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | - | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | c | \* | d | + | + | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  | j |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | + | ( | b | \* | c | + | d | ) | - | e |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  |  |  | i |

Pre( e) = false j = e j++

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | - | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | c | \* | d | + | + | e | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  | j |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Infix** | a | + | ( | b | \* | c | + | d | ) | - | e |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  |  |  | i |

Pre( e) = false j = e j++ and I reached it’s end

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | - | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | c | \* | d | + | + | e | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  | j |  |  |

Top = - top != -1 🡪 j = - and I reached it’s end

|  |  |  |
| --- | --- | --- |
| 4 |  |  |
| 3 |  |  |
| 2 |  |  |
| 1 |  |  |
| 0 | - | top |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | c | \* | d | + | + | e | - | 9 | 10 |
|  |  |  |  |  |  |  |  | j |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Postfix** | a | b | c | \* | d | + | + | e | - |
|  |  |  |  |  |  |  |  | j |

This is the required postfix equation of given infix.

**Infix to Prefix notation:-**

1. Follow the same rules of infix to postfix notation
2. Apply the algorithm of infix to postfix
3. Get the reverse of the string of the infix to postfix which will be the required infix to prefix notation

**Postfix Evalution:-**

If i is operand :

Push into stack

If i is operator:

Pop two values from stack

Op1

Op2

Then apply the operator as op2 op(+,-,\*,/,%) op1 and store in res

Then push(res) into stack

Finally stack[0] is the answer

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 4 | 5 | 2 | 3 | \* | - | \* | 2 | 3 | \* | + |

|  |
| --- |
|  |
|  |
|  |
| 6 |
| 5 |
| 4 |

5-6=-1

2\*3 = 6

|  |
| --- |
|  |
|  |
| 3 |
| 2 |
| 5 |
| 4 |

|  |
| --- |
|  |
|  |
|  |
|  |
| -1 |
| 4 |

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
| -4 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 4 | 5 | 2 | 3 | \* | - | \* | 2 | 3 | \* | + |

|  |
| --- |
|  |
|  |
|  |
| 3 |
| 2 |
| -4 |

|  |
| --- |
|  |
|  |
|  |
|  |
| 6 |
| -4 |

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
| 2 |

(-1)\*4 = (-4)

Completed part

6+(-4) = 2

3\*2 = 6

Infix to prefix

Prefix evalution

Post to infix

Prefix to infix

Postfix to prefix

prefix