

Postfix Evaluation And Conversions

1. You are given a postfix expression.
2. You are required to evaluate it and print it's value.
3. You are required to convert it to infix and print it.
4. You are required to convert it to prefix and print it.

Note -> Use brackets in infix expression for indicating precedence. Check sample input output for more details.

Input Format

Input is managed for you

Output Format

value, a number

infix

prefix

Constraints

1. Expression is a valid postfix expression
- *2. *The only operators used are +, -, , /*
3. All operands are single digit numbers.

Sample Input

264*8/+3-

Sample Output

((2+((6*4)/8))-3)

***-+2/6483**

```
def postToInPre(string):
    prefix = []
    infix = []

    for i in range(len(string)):
        ch = string[i]
        if ch in {'+', '-', '*', '/'}:
            inv2 = infix.pop()
            inv1 = infix.pop()
            temp1 = '('+inv1+ch+inv2+')'
            infix.append(temp1)
```

```
        prv2 = prefix.pop()
        prv1 = prefix.pop()
        temp2 = ch + prv1 + prv2
        prefix.append(temp2)
    else:
        prefix.append(ch)
        infix.append(ch)
    return prefix[-1],infix[-1]
```

```
string = '264*8/+3-'
```

```
print(postToInPre(string))
```