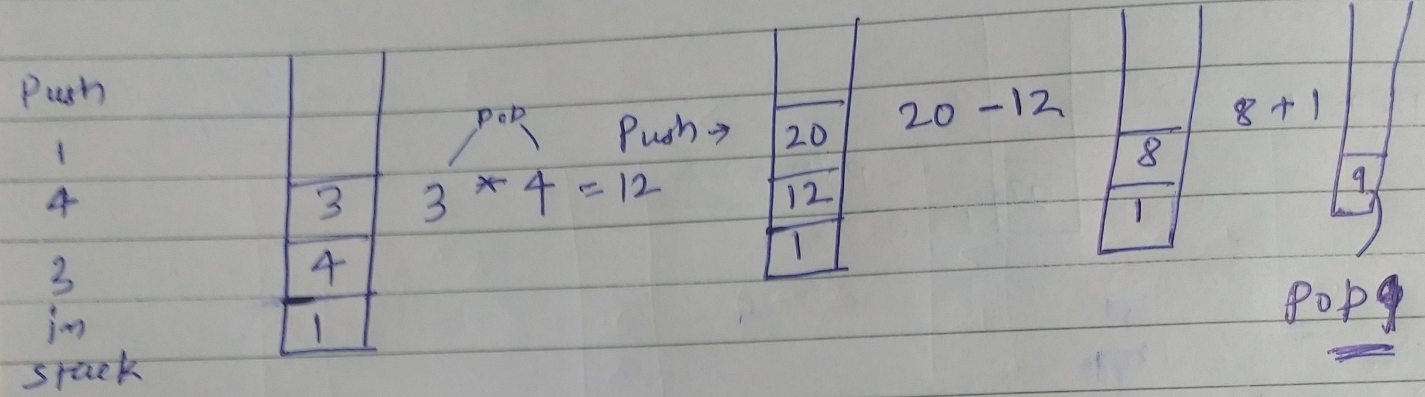
Day 3 Assignment:

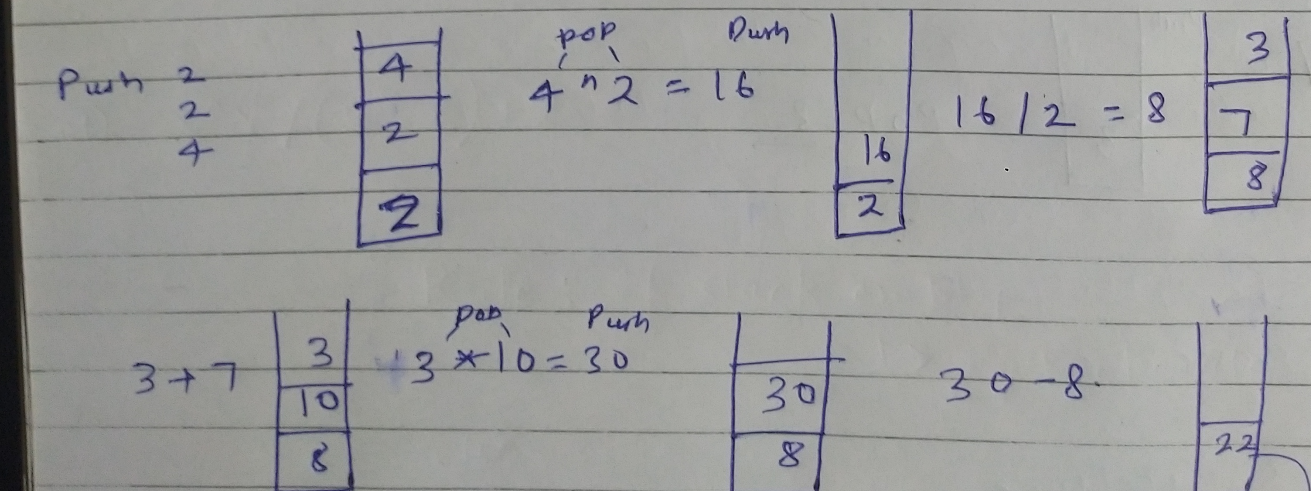
Q 4 → Evaluate following prefix expressions..

① + , − , 20 , * , 3 , 4 , 1

L ← R

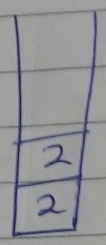


② − , * , 3 , + , 3 , 7 , / , ^ , 4 , 2 , 2 ←

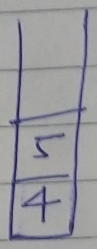


③ −, /, *, 3, ^, 5, 2, 15, −, 5, ^, 2, 2 ←

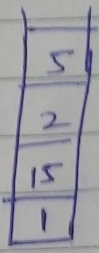
Push 2, 2 in stack



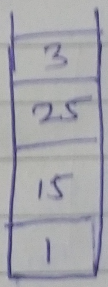
2 ^ 2

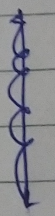


5 − 4

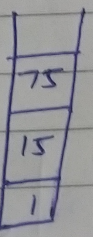


5 ^ 2

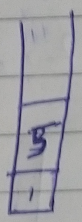




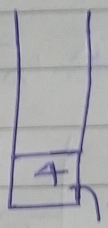
3 × 25 = 75



75 / 15



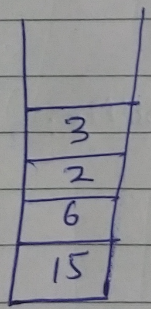
5 − 1



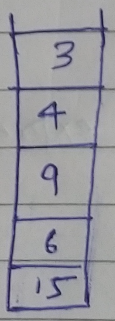
Pop 4 =

④ +, −, +, /, *, 2, 20, 2, *, +, 3, 4, ^, 3, 2, 6, 15 ←

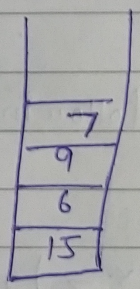
Push 15, 6, 2, 3 in stack



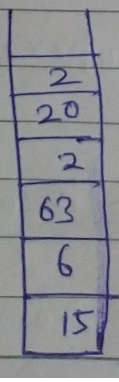
3 ^ 2



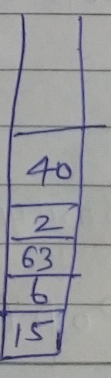
3 + 4



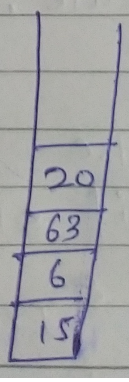
7 × 9 = 63



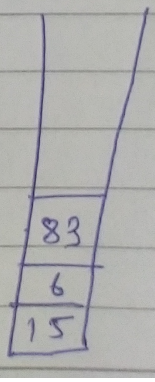
2 × 20



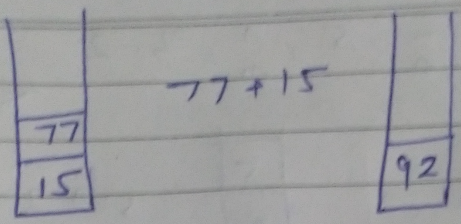
40 / 2



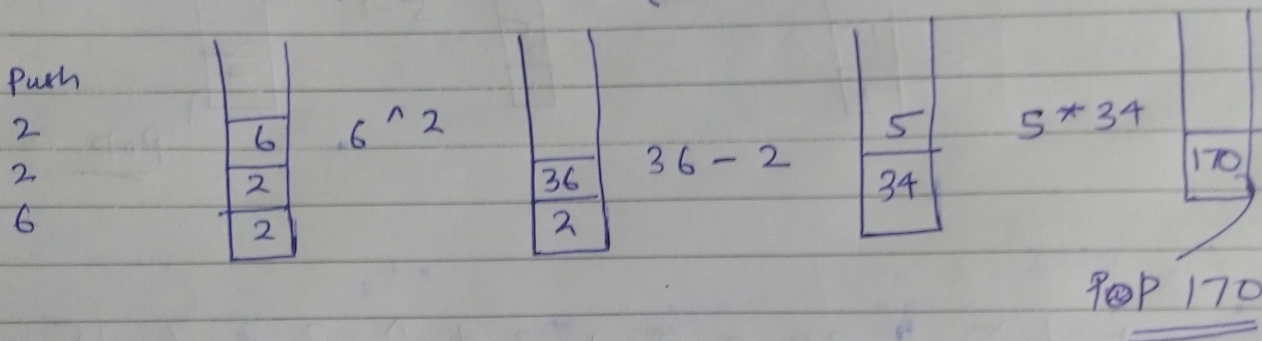
20 + 63



83 - 6



(5) *, 5, -, ^, 6, 2, 2 ←



Q5 Convert following infix expression to postfix:

(a) x^y/(5*z)+2

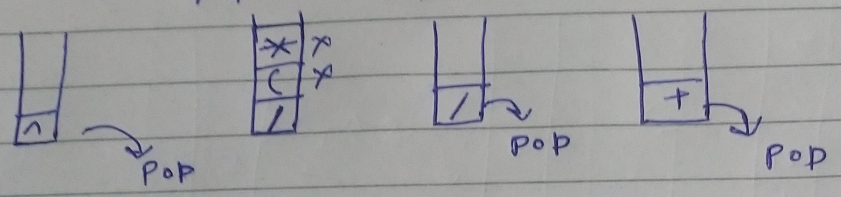
Rule: if stack empty
↳ Push operator in stack

else
if stack priority(operator) ≥ incoming operator (new operator)
{ True → Pop (operator from stack)
{ false → Push (into stack)

priority

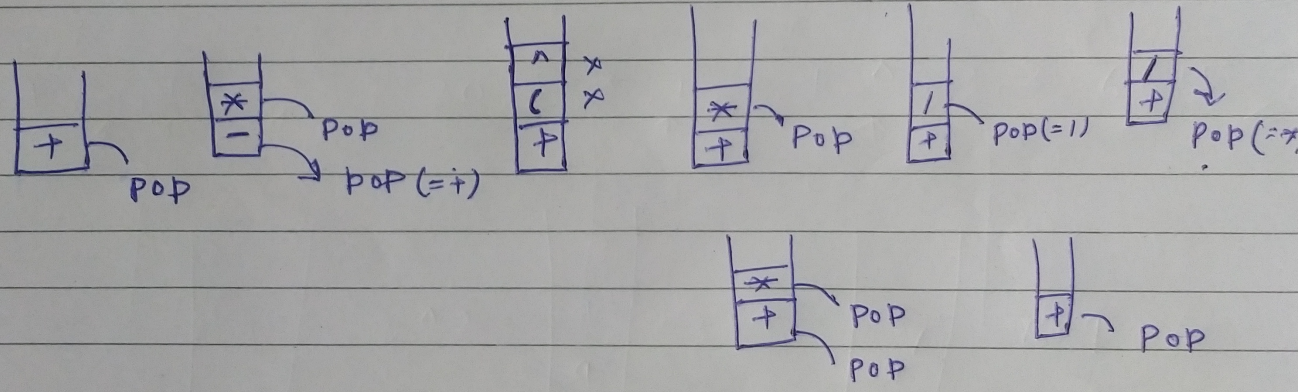
( )
[ ]
{ }
[exponent] L ← R
[*, /] L → R
[+, -] L → R

(a) X ^ Y / (5 * Z) + 2



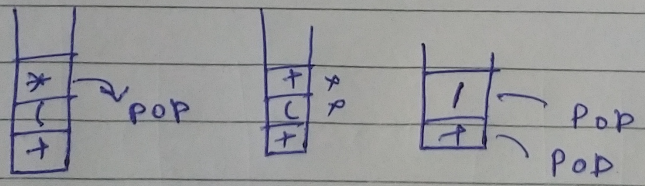
Postfix exp = X Y ^ 5 Z * 2 +

(b) K + L - M * N + (O ^ P) * W / U / V * T + Q



Postfix exp: K L + M N * - O P ^ W * U / V / T * + Q +

(c) A + (B * C + D) / E



A B C * D + E / + → Postfix expression