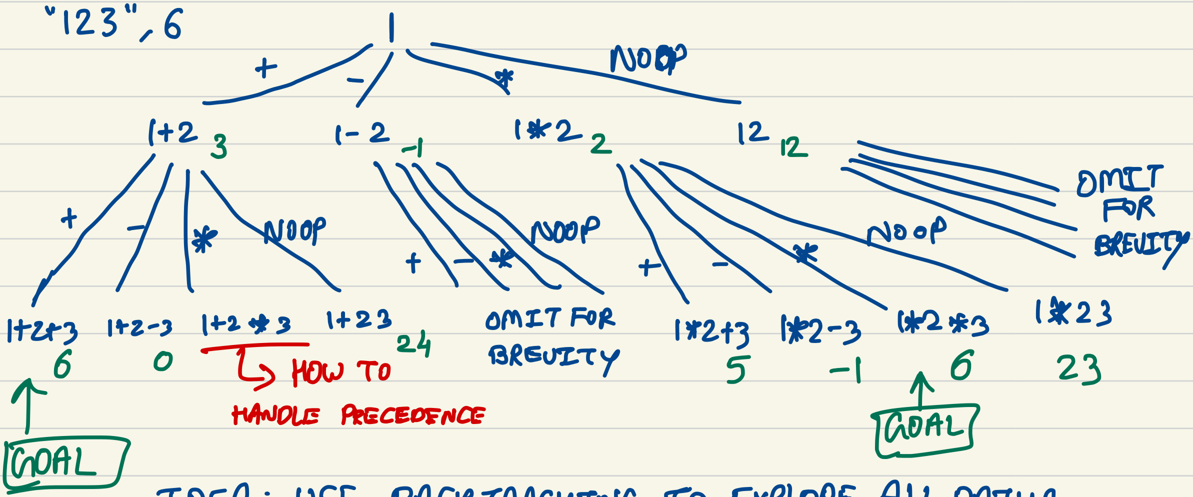
EXPRESSION ADD OPERATORS.

INPUT: STRING OF DIGITS ; TARGET.

TASK: ADD OPERATORS

+, −, *, NOOP

"123", 6



IDEA: USE BACKTRACKING TO EXPLORE ALL PATHS.

HOW TO DETERMINE GOAL STATE:

EVAL EXPRESSION <u>OR</u> COMPUTE ON FLY

↑ EXPENSIVE ↑ EFFICIENT.

HOW TO TAKE CARE PRECEDENCE.

1 + 2 → 3 * OVER + OR −

↕ → 3*3 WRONG

↳ 1 + 2 * 3 → IF WE KNOW LAST OPERATOR AND OPERAND, WE CAN DO IT.

3 UNDO +2

1 + 2 APPLY *3

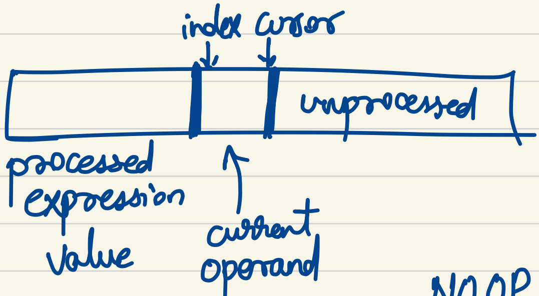
1 + (2 * 3)

↳ 7

Yash Pradhan Study Notes

DEEP DIVE:

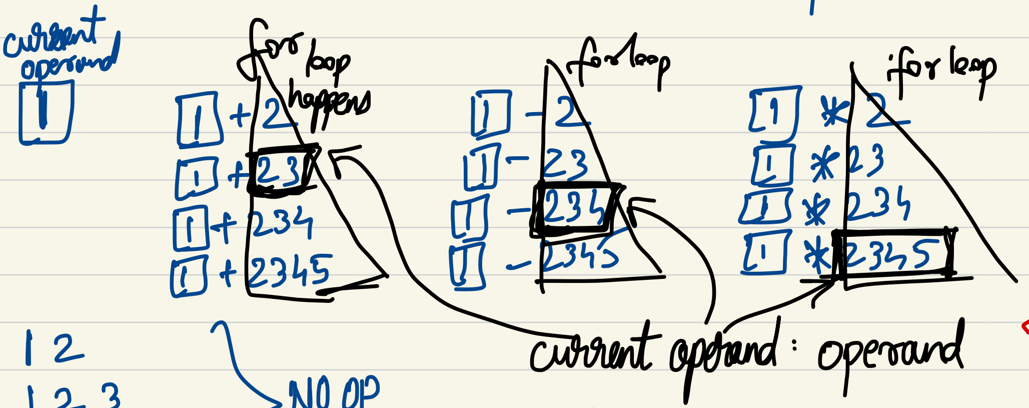
BACK TRACKING: EXPRESSION ADD OPERATORS



2 parts
for loop: NO OP
backtrack:
+/-/*

NO OP taken care here

for loop fans out current operand: index.... end
we parse it.



essentially last thing that happened

12
123
1234
12345
} NO OP

currentCall(expr, val, prevSignedOperand)

next params	expr	val	prevPrevSignedOpen
+	"expr+val"	val+operand	+operand
-	"expr -val"	val- operand	-operand
*	"expr *val"	val-prevSignedOperand + prevSignedOp * operand	+prevSignedOp * operand

AT LAST PRECEDING ZERO HANDLING