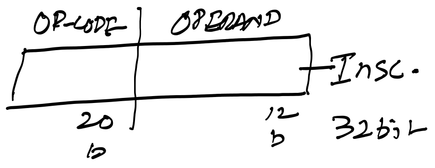
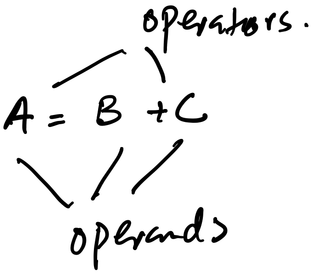
Code generation

Sunday, 29 November 2020 4:24 PM

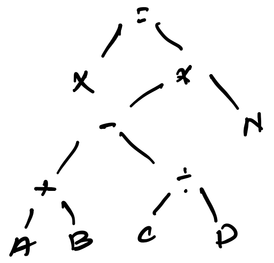


Three Addres Code (TAC):



$X = ((A+B) - (C \div D)) * N$

$E = A + B$
$F = C \div D$
$G = E - F$
$X = G * N$



$R1 = 2 * 3$ ✓
$R2 = 3 \div R1$ ✓
$R3 = R1 + 4$ ✓
$R5 = 3 \div R1$ ✗
$R6 = R5 - 1$ $R6 = R2 - 2$
$R7 = 2 * R6$ ✓

Optimised code:

$R1 = 2 * 3$
$R2 = 3 \div R1$
$R3 = R1 + 4$
$R6 = R2 - 1$
$R7 = 2 * R6$

OPTIMISATION

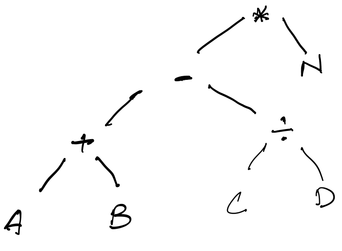
- less Code size
- faster Code execution speed.

- To find a balance b/w the size of the code and the speed of execution.

$X = ((A+B) - (C \div D)) * N$ Infix Notation.

Binary Tree

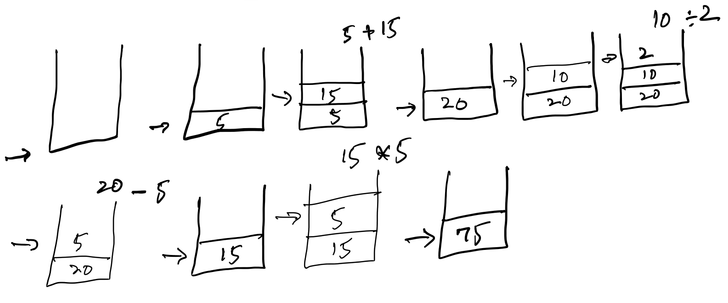
$A \leftarrow 5$
$B \leftarrow 15$
$C \leftarrow 10$
$D \leftarrow 2$
$N \leftarrow 5$

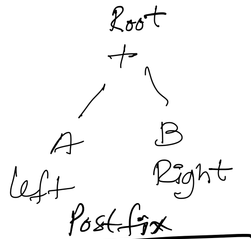


Infix

Left, Root, Right
A + B

RPN $A\ B\ +\ C\ D\ \div\ -\ N\ *$ Postfix Notation.





Postfix

left, Right, Root
A B +

Reverse Polis Notation.