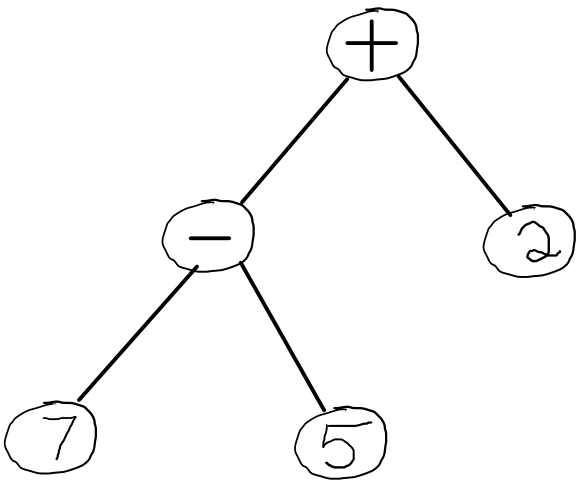
Example - Generate an AST from Math Expression

7 - 5 + 2



Associativity - direction of operation.

operation is left to right ---------->

-or- right to left <----------

Example: Plus and Minus ---->

$7 - 5 + 2$

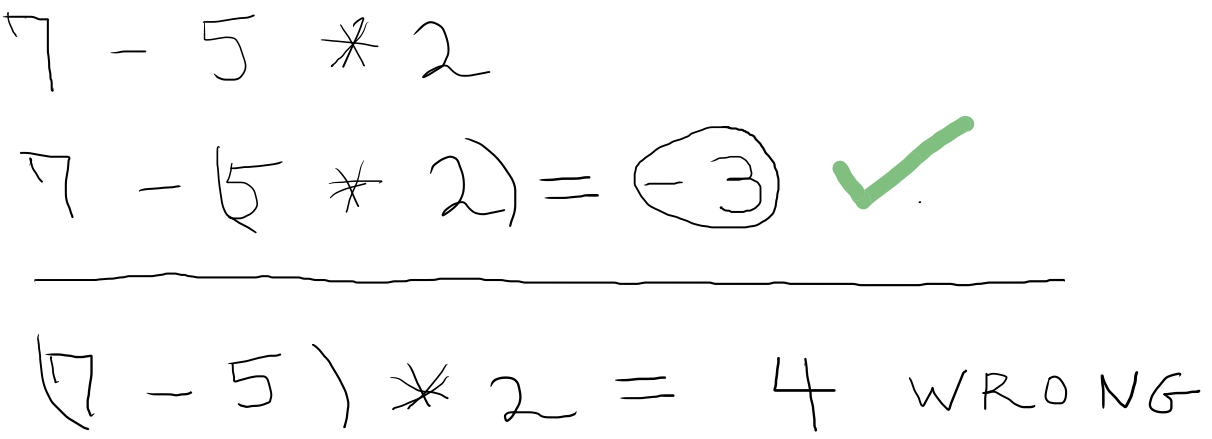
$(7 - 5) + 2 = 4$ ✓

$7 - (5 + 2) = 0$ WRONG

Precedence – order of operations.

Which operation is done first?

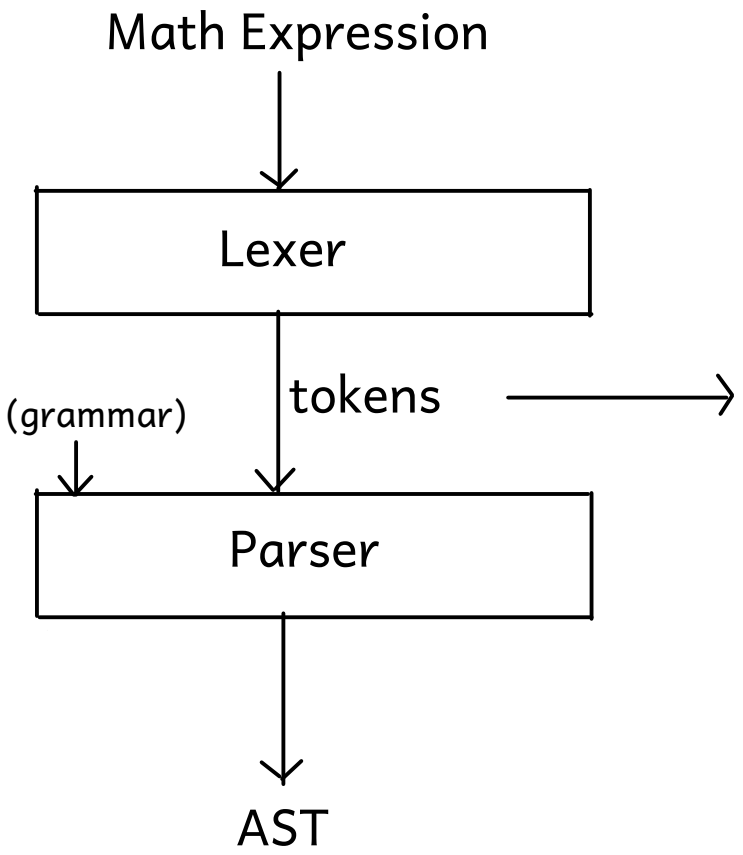
Example: Multiplication before Subtraction.



Operations Table

Operation	Associativity	Precedence
+ -	→	↓
* /	→	
* *	←	
( )	→	

Overall Design - based on compiler theory.

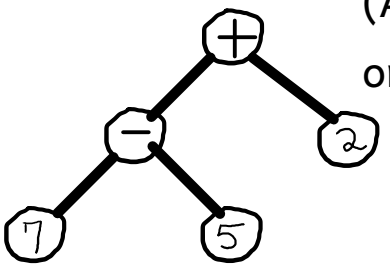


Example:

7 − 5 + 2

Num: 7
Minus
Num: 5
Plus
Num: 2

(A token is a string with an assigned meaning.)



(AST generated based on a grammar.)

Grammar

Set of rules for the arrangement of token to correctly produce an AST.

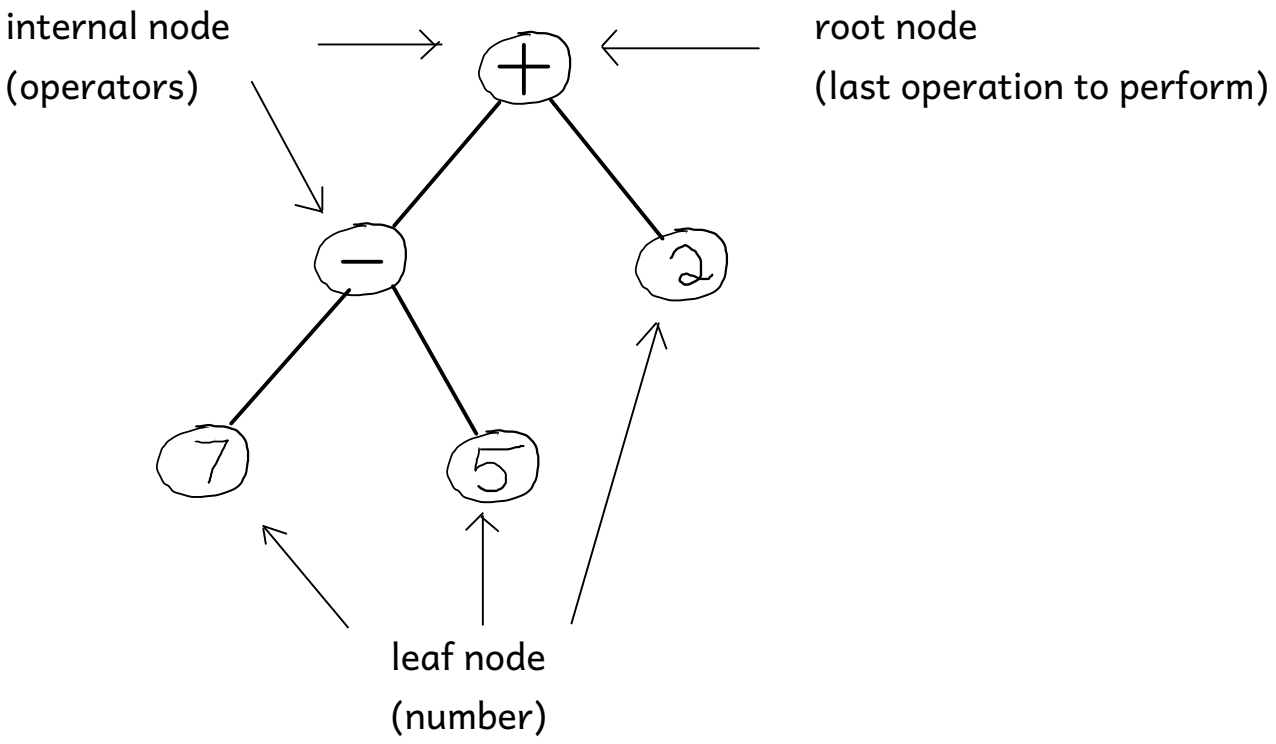
Grammar:

expr	-> additive
additive	-> multiplicative ((PLUS | MINUS) multiplicative)*
multiplicative	-> unary (((MULTIPLY | DIVIDE) unary) | LPAREN expr RPAREN)*
unary	-> (PLUS | MINUS) unary | exponential
exponential	-> primary POW exponential | primary
primary	-> NUMBER | LPAREN expr RPAREN

Note: Precedence increases as you move down. Associativity is left for EBNF (rules with '*'). For BNF, the operation is left/right associative if the rule is left/right recursive.

AST - Abstract Syntax Tree

7 - 5 + 2



Extras

- Visitor Pattern

- AST Visualization