# Evaluator

Kit Lao

April 26th, 2020

## Goal

Compute the value.

## Design

### Numbers

This module is essentially a wrapper around the computed values. It's not really necessary but good practice for extending this project into a compiler.

### Evaluator

So like the hard part is already done. Perform a post-order traversal on the parse-tree and perform the operation associated with each node. If the node is a leaf, just return it. Edge case for division if the right operand is 0.

```
function eval(tree):
    if tree is leaf:
        return tree.value
    else if tree.op is ADD:
        return eval(tree.x).value + eval(tree.y).value
    else if tree.op is SUBTRACT:
        return eval(tree.x).value - eval(y).value
    else if tree.op is MULTIPLY:
        return eval(tree.x).value * eval(tree.y).value
    else if tree.op is DIVIDE:
        return eval(tree.x).value / eval(tree.y).value
    else if tree.op is PLUS:
        return eval(tree.x).value
    else if tree.op is MINUS:
        return -1 * eval(tree.x).value
```

## Testing

It should work like a calculator.