

Class3

2021/5/25

宿題1 (「*」「/」に対応)

式を2巡する。

1回目は、「*」と「/」を計算する。

「*」や「/」があったら前後の値で計算を行い、式を削除して解を挿入する。

例)

$3.0 + 4 * 2 - 1 / 5 \rightarrow 3.0 + 8 - 0.2$ (4*2, 1/5を計算)

2回目は、「+」と「-」を計算する。

掛け算と割り算は既に計算され「+」「-」だけになっているので、前から順番に計算していく。

(関数evaluate_plus_minus)

例)

$3.0 + 8 - 0.2 \rightarrow 10.8$

宿題3 (括弧に対応)

前括弧「(」があったら、

token = {'type': 'FRONTBRACKET', 'index': index}

として、自身のindexを覚えておく。

後ろ括弧「)」があったら、

token = {'type': 'BACKBRACKET', 'front_index': front_index}

として、対応する前括弧のindexを覚えておく。(対応する前括弧を探すのに使う)

式に前括弧「(」があったら括弧の次の値からの式をevaluateに入れ、再帰的に計算を行う。計算が終わって解が返ってきたら、tokensから今計算した式と前後の括弧を削除し、解を挿入する。

例) (3.0 + 4 * (2 - 1)) / 5 の計算

evaluate((3.0 + 4 * (2 - 1)) / 5)

->

evaluate(evaluate(3.0 + 4 * (2 - 1)) / 5)

->

```
evaluate(evaluate(3.0 + 4 * evaluate(2 - 1)) / 5)
->
evaluate(evaluate(3.0 + 4 * evaluate(2 - 1)) / 5)
->
evaluate(evaluate(3.0 + 4 * 1) / 5)
->
evaluate(evaluate(3.0 + 4 * 1) / 5)
->
evaluate(7 / 5)
->
1.4
```

コメント

前括弧と後ろ括弧の対応を見つけて、式を削除するのがもう少しスマートにできないかな、と思っています...(今は前括弧のindexを持つ後ろ括弧を式の中から線形探索している)
最初、計算に関わる式はevaluateに全て書いていたのですが、冗長になってしまったので”
足し算引き算”、”掛け算割り算”、”括弧”で関数を分けてみたのですが読みやすいコードなのかよくわからなくなってきました...

テストケースには入れていないのですが0で割った時は

```
> 1/0
Can't divide by 0
answer = 0.000000
こんな感じで出ます
```