step\_3 homework

* 掛け算と割り算の実装

トークン列を受け取り、掛け算と割り算をした後の新しいトークン列を返す。

* new\_tokensを作り、予め最初の要素（＋記号）を入れておく。
* 要素数が２以下の場合は渡されたトークン列をそのまま返す。
* tokensの要素を前から順に見ていって、次の要素が「\*」か「/」だったら、

更に次の要素と計算した結果をresに代入し、indexを２つ進める。

* 現在見ている要素の次の要素が「\*」または「/」以外の場合、新しいトークン列に現在のresと、次の要素を追加し、indexを１つ進める。
* tokensの要素が残り一つになったら、その要素は必ず数字なのでnew\_tokensにそのまま追加する
* （）の実装

トークン列を受け取り、最も外側で最初の括弧の中を再帰的に計算していくことで、全てのかっこを処理し、新しいトークン列を返す。

* new\_tokensを作り、予め最初の要素（＋記号）を入れておく。
* かっこを記録するための空のリスト、parenthesesを作る
* トークン列を最初から見ていって、開きかっこが出てきたら、そのindexをparenthesesに追加する
* 閉じかっこが現れたら、parenthesesの最後の要素から削除していき、parenthesesの要素が0になった時、かっこが閉じていることになるので、最後に削除した時の要素から現在のindexまでのトークン列を計算し、その結果を新しいトークン列に代入する。
* parenthesesの要素数が0のときは括弧の外なので、要素をそのまま新しいトークン列に追加する