

Bitte beachten Sie: Die Herausgabe oder der Austausch von Code (auch von Teilen) zu den Programmieraufgaben führt für *alle* Beteiligten zum *sofortigen Scheinverlust*. Die Programmieraufgaben müssen von allen Teilnehmenden alleine bearbeitet werden. Auch Programme aus dem Internet dürfen nicht einfach kopiert werden.

1 Terme

Wir definieren korrekte Terme über $\mathbb{Z}_{\geq 0}$ und ihre Tiefe rekursiv. Ein Term w_0 , der aus genau einer nichtnegativen ganzen Zahl besteht, ist korrekt und hat Tiefe $t_0 = 0$. Wenn w_1 und w_2 korrekte Terme der Tiefe t_1 bzw. der Tiefe t_2 sind, dann sind

- $w_1 + w_2$ und $w_1 * w_2$ korrekte Terme der Tiefe $\max(t_1, t_2)$,
- (w_1) , $\{w_1\}$ und $[w_1]$ korrekte Terme der Tiefe $t_1 + 1$.

2 Aufgabenstellung und Anforderungen

In dieser Aufgabe soll eine Zeichenkette analysiert und ggf. ausgewertet werden. Entweder stellt ein übergebener String einen korrekten Term über den nichtnegativen ganzen Zahlen dar oder er ist im Sinne von Abschnitt 1 syntaktisch inkorrekt. Im ersten Fall soll er unter Berücksichtigung der üblichen Rechenregeln (Punkt-vor-Strich-Regel(n), Distributivgesetz(e)) ausgewertet werden.

Schreiben Sie eine Funktion `evaluate(string)`, die für einen korrekten Term dessen Wert und dessen Tiefe berechnet.

2.1 Eingabe

Es wird ein String übergeben.

2.2 Ausgabe

Falls die Zeichenkette einen korrekten Term darstellt, soll ein 2-Tupel ganzer Zahlen (v, t) mit Termwert v und Termtiefe t zurückgegeben werden. Für jeden syntaktisch inkorrekten Term soll

`Exception('syntaktisch inkorrekt')`

ausgelöst werden.

3 Beispielaufufe

(gekürzter Output)

```
1 >>> evaluate("1+(1+1)*(1+1)")
2 (5, 1)
3 >>> evaluate("{1+1}*[1+1]+38")
4 (42, 1)
5 >>> evaluate("[{1}+5]*({2}+[1*(3)}+2])")
6 (42, 4)
7 >>> evaluate("{3+2)+1")
8 Traceback (most recent call last):
9     ...
10    ...
11    File "PA08solution.py", line 95, in throw_ex
12        raise Exception("syntaktisch inkorrekt")
13 Exception: syntaktisch inkorrekt
```

4 Anmerkungen

In dieser Aufgabe stehen Ihnen mehrere Lösungswege zur Verfügung. Allerdings legt die Definition korrekter Terme bei der Auswertung solcher das Ausnutzen der rekursiven Struktur nahe. Das können Sie beispielsweise folgendermaßen umsetzen:

- Schreiben Sie eine Funktion `eval_depth0`, die korrekte Terme der Tiefe 0 auswertet.
- Benutzen Sie beim Einlesen des Strings eine Variable `current`,
 - a) die bei öffnenden Klammern auf den leeren String gesetzt wird und deren (vorheriger) Inhalt auf einen Stack abgelegt wird,
 - b) die die Zeichenkette bis zur nächsten Klammer speichert,
 - c) deren Inhalt bei schließenden Klammern ausgewertet und mit dem obersten Element des Stacks konkateniert wird.

Hinweis: Es ist über die Standard-python-Bibliotheken hinaus kein zusätzliches Modul zu importieren. Zusätzliche Module wie z.B. `numpy` können vom Comajudge in der Regel nicht importiert werden und führen daher zu Fehlern.