

Übung 6

Ausgabe von ganzen Zahlen

Themen: Rekursion, divmod, String-Konkatenation

Schreiben Sie eine Funktion `integer_as_string`, welche eine positive oder negative ganze Zahl als String ausgibt.

Ausgabe von Fließkomma-Zahlen

Themen: Rekursion, divmod, String-Konkatenation

Schreiben Sie eine Funktion `float_as_string`, welche eine Fließkommazahl als String ausgibt. Es sollen maximal 10 Nachkommastellen ausgegeben werden. Es muss nicht gerundet werden.

Ausgabe von Datenstrukturen

Themen: Rekursion, match...case

Schreiben Sie eine Funktion `as_string`, welche eine geschachtelte Python-Datenstruktur als Argument erhält und ähnlich wie die Python-Funktion `str` eine Stringrepräsentation des Arguments zurückgibt. Die Funktion soll Listen, Tupel, Sets, Dictionaries, Strings, Zahlen und Wahrheitswerte korrekt in Strings umwandeln können.

Cäsar-Verschlüsselung

Themen: class, __call__, enumerate, Dictionary Comprehension, join

Bei der Cäsar-Verschlüsselung werden zunächst alle Zeichen auf Zahlen abgebildet. Zur Verschlüsselung wird zu jeder Zahl ein Offset addiert. Dann werden die Zahlen wieder auf Buchstaben abgebildet und ausgegeben.

Beispiel mit 27 Zeichen (einschließlich Leerzeichen) und dem Offset 5:

```
'hallo welt' ⇒ 8 1 12 12 15 0 23 5 12 20  
              ⇒ 13 6 17 17 20 5 1 10 17 25 ⇒ 'mfqqteaajqy'
```

Implementieren Sie eine Klasse `Kodierer`, welche die Cäsar-Verschlüsselung implementiert. Die `init`-Methode bekommt den Offset als Argument. Die Klasse besitzt außerdem eine Methode `__call__`, welche einen Text als Argument erhält und den kodierte Text zurückgibt.

Fügen Sie außerdem Code zum Testen der Klasse hinzu.

Vorüberlegungen:

- Wozu dient die Methode `__call__`?
- Was müssen Sie beachten, wenn Sie 'z' verschlüsseln?