Mathematisches Institut
Prof. Dr. Achim Schädle
Marina Fischer

Heinrich Heine Universität Düsseldorf

25.10.2018

Computergestützte Mathematik zur linearen Algebra – 3. Übungsblatt

Hinweis: Bearbeiten Sie das Blatt in Spyder. Erstellen Sie für jede Aufgabe ein neues Skript und speichern Sie diese als *AufgabeX.py*, wobei X die jeweilige Aufgabennummer ist.

Aufgabe 9: (*Größter gemeinsamer Teiler*)
Befehle: `if`, `while`
Für zwei Zahlen a, $b \in \mathbb{Z}$ ist der größte gemeinsame Teiler (ggT) die größte natürliche Zahl N, die a und b ohne Rest teilt. Implementieren Sie zur Bestimmung des ggT von $a, b \in \mathbb{N}$...

(a) ... eine iterative Variante `ggt_it`(a, b):
Hierbei werden in der Funktion `ggt_it`(a, b) die drei Schritte

```
h = Divisionsrest von a/b
a = b
b = h
```

ausgeführt, solange $b \neq 0$ gilt. Sobald $b = 0$ erreicht ist, setzen wir $N = a$ und haben den ggT gefunden.

Zusatzfrage: Wie lässt sich die iterative Variante implementieren ohne die Hilfsvariable h zu verwenden?

(b) ... eine rekursive Variante `ggt_rek(`a`,`b`)`:
Falls $b = 0$ gilt, nimmt N den Wert a an. Andernfalls rufen wir die Funktion `ggt_rek` mit den Eingabeargumenten b und h erneut auf, wobei h der Divisionsrest von a/b ist, und speichern die Auswertung in der Variable N. Ist die Rekursion beendet, gibt N den ggT an.

(c) Testen Sie Ihre Funktionen mit den Zahlenpaaren (2469134, 8641969), (-345, 15), (7892389, -3).

Aufgabe 10: (*Eigene Klasse: Brüche*)
Floats können in Python nicht so groß wie *Integer* werden, d.h. es gibt ganze Zahlen, die als *Integer*, aber nicht als *Floats* gespeichert werden können. Wir wollen daher eine Klasse `Bruch` entwickeln, die IMMER nur mit Integern rechnet und mit der man exakt mit Brüchen rechnen kann.
Ein (leeres) Grundgerüst für diese Klasse finden Sie auf der Internetseite zur Vorlesung.

(a) `Bruch` soll den Zähler und den Nenner als Integer übergeben bekommen und diese speichern. (Der Aufruf `a=Bruch(2,7)` entspricht also $^2/_7$, wobei uns `a.zaehler` den Wert 2 und `a.nenner` den Wert 7 ausgeben soll.) Der Nenner soll intern immer größer als 0 sein. Sollte er 0 sein, werden Zähler und Nenner auf 0 gesetzt und es wird eine sinnvolle Warnung ausgegeben.

(b) Um mit den Objekten der Klasse `Bruch` rechnen zu können, wollen wir die Standardsymbole $+$, $-$, $*$ und $/$ verwenden können. Dies geht durch das Implementieren bestimmter Methoden. Beispielsweise bekommt `__mul__(self, other)` (Multiplikation) sich selbst (`self`) und den anderen Faktor (`other`) übergeben und gibt ein Objekt der Klasse `Bruch` zurück.
D.h. für `a=Bruch(3,2)` und `b=Bruch(-5,9)` greift der Befehl `a*b` auf die Methode `a.__mul__(b)` zurück.
Achtung: Beim Dividieren kann es vorkommen, dass Sie durch 0 teilen (siehe (a)).

(c) Implementieren Sie die Klassenmethode `kuerzen(zaehler, nenner)`, welche den gekürzten Zähler und Nenner zurückgibt. Ihre Klasse soll jetzt immer den gekürzten Bruch speichern,

d.h für `a=Bruch(8,6)` soll `a.zaehler=4` und `a.nenner=3` sein.
Verwenden Sie hierfür entweder eines Ihrer eigenen Verfahren zur Bestimmung des ggT aus Aufg. 9 oder führen Sie außerhalb der Klasse den Befehl `from math import gcd` aus und verwenden die PYTHON-Implementierung `gcd`.

(d) Implementieren Sie die Klassenmethode `tofloat()`, welche den Bruch als Gleitkommazahl zurückgibt.

(e) Testen Sie Ihre Klasse an einigen Beispielen.

WICHTIG: Wie oben schon erwähnt, rechnen Sie in dieser Klasse IMMER nur mit Integern!

Aufgabe 11: (*Eigene Klasse: Chemische Elemente*)

(a) Erstellen Sie eine PYTHON-Klasse `Element`, die zur Initialisierung folgende Parameter übergeben bekommt: Name (Name des Elementes als String), Symbol (Symbol als String, Gold → 'Au'), Ordnungszahl (als Integer), Schmelzpunkt (in °C als Float), Siedepunkt (in °C als Float) und Temperatur (aktuelle Temperatur des Objektes in °C als Float). Sollte keine Temperatur übergeben werden, wird diese auf 20°C gesetzt.

(b) Schreiben Sie eine Methode `aendereTemperatur(temp)`, welche die Temperatur des Objektes auf `temp` ändert und die aktuelle Temperatur als Attribut `Temperatur` speichert, d.h. `Gold.Temperatur` liefert immer die momentane Temperatur des Objekts `Gold`.

(c) Erweitern Sie die Methode aus (b) so, dass zusätzlich das Attribut `Aggregatzustand` ('fest', 'flüssig' oder 'gasförmig' als String), basierend auf der aktuellen Temperatur und dem Schmelz- und Siedepunkt, erzeugt wird. Achten Sie darauf, den Zustand zu aktualisieren, wenn Sie die Temperatur ändern.

(d) Der untere Grenzwert für die Temperatur beträgt -273.15°C. Erweitern Sie Ihre Klasse so, dass kleinere übergebene Temperaturen auf den absoluten Nullpunkt gesetzt werden.

(e) Schreiben Sie eine Methode `eigenschaften()`, welche eine kurze Zusammenfassung des Objektes (Name, Symbol, Temperatur und Aggregatzustand) ausgibt, beispielsweise so:
```
Name: Gold (Au)
Temperatur: 42°C (fest)
```

(f) Testen Sie Ihre Klasse an den Elementen Stickstoff (Temperatur: 25°C), Erbium (-5.3°C) und Dysprosium. Lassen Sie sich die Eigenschaften ausgeben. Ändern Sie die Temperatur des Stickstoffs auf -200.02°C, die des Dysprosiums auf -314°C und die des Erbiums auf 3456.7°C. Geben Sie dann noch einmal die Eigenschaften aus. (Die Schmelz- und Siedetemperaturen finden Sie z.B. bei Wikipedia.)

Aufgabe 12: (*Erweiterung der Klasse: Polynome*)
In der Vorlesung (Lektion 3) haben Sie eine Klasse `Polynom` gesehen, die ein Dictionary mit den Koeffizienten $a_0, \ldots, a_n$ eines Polynoms $p(x) = \sum_{i=0}^{n} a_i\, x^i,\ n \in \mathbb{N}$, als Eingabe erhält. Erweitern Sie die Klasse nun um folgende Methoden:

(a) Zur Multiplikation eines Polynoms vom Grad n und eines vom Grad m mit $m, n \in \mathbb{N}$ benötigen wir die Implementierung einer Methode `__mul__(self, other)`.

(b) Schreiben Sie eine Methode `diff()`, die die erste Ableitung eines Polynoms p angibt.

(c) Wir wollen auch noch das Integral $\int_a^b p(x)\,\mathrm{d}x$ für $a, b \in \mathbb{R}$ auswerten können. Schreiben Sie hierzu eine passende Methode `integral(a,b)`.

(d) Testen Sie Ihre Methode am Polynom $p(x) = 3 + x + 8x^2 + 5x^4$. Geben Sie hierbei die erste Ableitung von p an und werten Sie das Integral über p in den Grenzen 0 und 1 aus.

Besprechung in den Übungen vom 29. Oktober - 9. November 2018.