

Informatik I: Einführung in die Programmierung

Prof. Dr. Bernhard Nebel
Dr. Stefan Wölfl, Thorsten Engesser, Tim Schulte
Wintersemester 2016/2017

Universität Freiburg
Institut für Informatik

Übungsblatt 8

Abgabe: Freitag, 16. Dezember 2016, 20:00 Uhr

WICHTIGE HINWEISE: Zur Bearbeitung der Übungsaufgaben legen Sie bitte ein neues Unterverzeichnis `sheet08` im Wurzelverzeichnis Ihrer Arbeitskopie des SVN-Repositorys an. Ihre Lösungen werden dann in Dateien in diesem Unterverzeichnis erwartet.

Beachten Sie bitte bei allen Aufgaben die *Hinweise zur Bearbeitung der Übungsaufgaben* unter der folgenden URL (**insbesondere wird ab sofort die Berücksichtigung unseres Python Style Guides vorausgesetzt**):

<http://gki.informatik.uni-freiburg.de/teaching/ws1617/info1/guide/hinweise.html>

Bewertet wird bei allen Aufgaben die letzte Version, die zur Deadline des Übungsblattes auf dem SVN-Server eingereicht ist.

Aufgabe 8.1 (Brainf*ck; Dateien: `bf.py`, `bangs.b`; Punkte: 4+4)

In dieser Aufgabe soll die Sprache Brainf*ck um ein zusätzliches Sprachelement erweitert werden. Wir wollen eine einfache Möglichkeit zur bedingten Ausführung von Code realisieren, nämlich eine Verzweigung ohne *else*-Zweig. Dafür sollen die neuen Zeichen `{` und `}` eingeführt werden. Idee dabei ist, dass der Code in den geschweiften Klammern ausgeführt wird, *falls* bei Betreten der Verzweigung der Wert in der aktuellen Zelle gleich 0 ist; andernfalls wird der geklammerte Code übersprungen. Genauer besitzen die beiden neuen Zeichen folgende Bedeutung:

- `{`: Falls der Wert in der aktuellen Zelle gleich 0 ist (`data[ptr] == 0`), dann setze die Ausführung mit dem Befehl nach der öffnenden Klammer fort. Andernfalls, springe zum Befehl nach der zugehörigen schließenden Klammer.
- `}`: Fahre mit dem Befehl nach der Klammer fort.

- (a) Laden Sie den Brainf*ck-Interpreter von der Webseite der Vorlesung herunter und erweitern Sie diesen um das eben beschriebene Sprachelement.
- (b) Schreiben Sie ein Brainf*ck-Programm, das für einen beliebigen Eingabestring überprüft, ob in diesem das Zeichen `!` mindestens drei mal vorkommt. Ist dies der Fall, soll das Programm `OK` ausgeben. Andernfalls soll nichts (außer der Mitteilung, dass die Ausführung beendet wurde) ausgegeben werden. Greifen Sie hierbei auf das zuvor implementierte Sprachelement zurück. Dass der Eingabestring bis zum Ende gelesen wurde, können Sie daran erkennen, dass die Einleseoperation `,` den Wert der aktuellen Zelle nicht mehr ändert.

Aufgabe 8.2 (Raute; Dateien: `rhombus.txt`, `hw_geoclasses.py`; Punkte: 2+4+4)

In dieser Aufgabe geht es darum die in der Vorlesung vorgestellten Geometrie-Klassen um eine Klasse `Rhombus` zu erweitern, mit der Rauten¹ repräsentiert werden sollen. Wir wollen dabei nur solche Rauten betrachten, die auf der Spitze stehen, dessen Höhe und Breite also parallel zu den Bildschirmkoordinaten sind. Um diese Aufgabe zu realisieren erweitern Sie die in der Vorlesung vorgestellten Geometrie-Klassen, die in der Datei “`hw_geoclasses.py`” definiert werden und die hier bereitgestellt wird:

http://gki.informatik.uni-freiburg.de/teaching/ws1617/info1/python/hw_geoclasses.py

- (a) Durch welche Parameter kann eine Raute sinnvoll und eindeutig beschrieben werden? Von welcher bereits existierenden Klasse sollte sich Ihre Klasse `Rhombus` sinnvoller Weise ableiten und warum?

Schreiben Sie Ihre Antworten zu dieser Teilaufgabe in die Datei `rhombus.txt` und vergessen Sie nicht, diese ins SVN zu committen.

- (b) Ergänzen Sie die Datei `hw_geoclasses.py` mit Ihrer Implementierung der Klasse `Rhombus`. Diese Klasse soll neben einer geeigneten `__init__`-Methode, analog zu den vorhandenen Klassen, auch die Methoden `area` (zur Berechnung des Flächeninhalts der Raute) und `change_size` (zur prozentualen Veränderung der Größe) enthalten, sowie `stretch_height` und `stretch_width` (zur prozentualen Veränderung von Höhe/Breite, ohne dabei die jeweils andere Dimension zu verändern).
- (c) Nutzen Sie das in der Vorlesung eingeführte Modul `tkinter`, um zwei verschiedene Instanzen der Klasse `Rhombus` nebeneinander als grafische Objekte zu visualisieren.

Aufgabe 8.3 (Erfahrungen; Datei: `erfahrungen.txt`; Punkte: 2)

Legen Sie im Unterverzeichnis `sheet08` eine Textdatei `erfahrungen.txt` an. Notieren Sie in dieser Datei kurz Ihre Erfahrungen beim Bearbeiten der Übungsaufgaben (Probleme, benötigter Zeitaufwand nach Teilaufgabe, Interessantes, etc.).

¹<http://de.wikipedia.org/wiki/Raute>