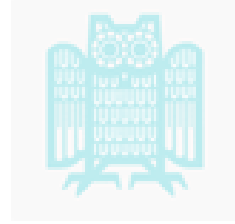




mpi

Universität  
des  
Saarlandes  
FR 6.2 Informatik



Prof. Dr. Kurt Mehlhorn, Dr. Konstantinos Panagiotou

WiSe 2011/2012

## Übungen zu Computational Thinking

<http://www.mpi-inf.mpg.de/departments/d1/teaching/ws11/ct/>

Blatt 4

Abgabeschluss: 28. 11. 11 16:00

**Regeln:** Bis zum Semesterende müssen mindestens 42% der maximal erreichbaren Punkte aller Übungszettel erworben werden.

Programmcode ist elektronisch per E-Mail abzugeben. Zusätzlich müssen die Ausgaben einer exemplarischen Programmausführung mitgeliefert werden.

**Aufgabe 1 (10 Punkte)** Beantworten Sie die folgenden Fragen aus Jeopardy! und beschreiben Sie die Strategie, nach der Sie vorgegangen sind (machen Sie das bitte auch, wenn sie die Antwort ohne Nachschlagen wissen).

- Making him a revolutionary martyr, the Bolivian Army tracked down this man and killed him in 1967.
- After the death of her husband in 1861, she did not go out in public again for three years, and refused to wear anything but black.
- The original head shrinker, he fled to London in 1938 following the Nazi annexation of Austria.
- This Four Quartets poet was awarded both the Nobel Prize and the Order of Merit in 1948.
- Now retired, this foreign-born American influenced U.S. national policy for decades.

**Aufgabe 2 (25 Punkte)** Betrachten Sie den Graph von Webseiten in Abbildung 1 und bestimmen Sie die Wichtigkeit der einzelnen Knoten.

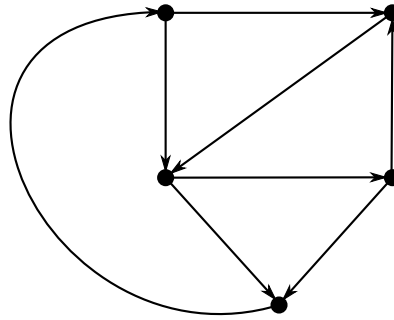


Abbildung 1: Ein Webgraph

- a) 15 Punkte. Stellen Sie ein Gleichungssystem auf und lösen Sie es.
- b) 10 Punkte. Machen Sie 20 Schritte eines Random Walks. Benutzen Sie bitte einen Würfel für die Zufallsentscheidungen.

**Aufgabe 3** (0 Punkte) Lesen Sie Kapitel 5 bis 7 des Python Tutorials<sup>1</sup>.

**Aufgabe 5** (15 Punkte) Ein Graph lässt sich leicht als ein Python-Wörterbuch darstellen. Die Schlüssel des Wörterbuchs sind Knoten, die Werte eine Liste mit Nachbarsknoten. Wenn man in den Listen Tupel speichert, kann man zusätzliche Daten wie etwa Kantenlängen verwenden.

Schreiben Sie eine Klasse `Graph`, die ein solches Wörterbuch verwaltet und Methoden bietet, um

- a) einen Knoten hinzuzufügen oder zu löschen,
- b) eine Kante hinzuzufügen oder zu löschen,
- c) festzustellen, ob eine Kante im Graphen ist,
- d) den an einer Kante gespeicherten Wert zu bekommen,
- e) eine Liste der Nachbarn eines Knoten zu erhalten,
- f) eine Liste der Knoten zu erhalten.

Installieren Sie `GraphViz`<sup>2</sup> auf Ihrem Computer und lesen Sie genug vom `Dotguide`<sup>3</sup>, um

<sup>1</sup><http://docs.python.org/tutorial/datastructures.html>

<sup>2</sup><http://www.graphviz.org/>

<sup>3</sup><http://www.graphviz.org/Documentation/dotguide.pdf>

eine Funktion zu schreiben, die eine textuelle Repräsentation eines `Graph`-Objekts in eine Textdatei schreibt, so dass der Graph mit `GraphViz` betrachtet werden kann.

**Aufgabe 7 (10 Punkte)** Schreiben Sie ein Programm, das Ihnen das Anfertigen der Essays zur Vorlesung abnimmt. Gehen Sie dabei so vor:

- a) Ausgehend von einer Textdatei mit relevantem Inhalt, wie zum Beispiel einem Wikipediaartikel, erstellen Sie ein Wörterbuch, das zu jedem Wort  $w$  des Textes eine Liste mit Wörtern enthält, die im Text auf  $w$  direkt folgen. Die Liste enthält ein Wort  $x$  mehrmals, wenn es im Text mehrere Stellen gibt, an denen  $x$  auf  $w$  folgt.
- b) Machen Sie einen Random Walk auf diesem Wort-Graphen: Von einem zufälligen Wort des Textes startend, geben Sie das Wort aus und wählen Sie ein zufälliges Wort aus der Folgewortliste aus. Wiederholen Sie den Vorgang so lange, bis ein ausreichend langes Essay erstellt wurde. Sollte ein Wort kein Folgewort haben, wird stattdessen mit einem zufälligen Wort weitergemacht.

Probieren Sie ihr Programm auf den gesammelten Werken von Shakespeare, erhältlich aus dem Projekt Gutenberg<sup>4</sup>, aus, um einen Text mit 100 Wörtern zu erzeugen.

---

<sup>4</sup><http://www.gutenberg.org/cache/epub/100/pg100.txt>