



## Übungsblatt 8

Willkommen zum Praktikum zu Programmieren 3. Dies sind Aufgaben aus Online-Tests von früheren Jahren.

**Aufgabe 1.** Schreiben Sie ein Python-Skript, das zählt wie häufig verschiedene Wörter als Kommandozeilenparameter vorkommen. Geben Sie alle Wörter mit den häufigsten Vorkommen in alphabetischer Reihenfolge aus. In Beispiel sind zwei Wörter am häufigsten (zwei mal) vorgekommen.

```
$ a1 ein Text der als ein Beispiel benutzt wird und danach weggeworfen wird
2:ein
2:wird
```

**Aufgabe 2.** Implementieren Sie ein einfaches Verfahren zur Komprimierung von Texten. Für jedes Zeichen, das hintereinander mehr als einmal vorkommt ersetzen wir alle diese Zeichen durch einmal dieses Zeichen gefolgt von der Häufigkeit des Vorkommens als Dezimalzahl. Gehen Sie davon aus, dass im Text nur die Zeichen des Alphabets (Groß- und Kleinbuchstaben) vorkommen. Schreiben Sie ein Python-Programm, das alle Kommandozeilenparameter komprimiert und zeilenweise ausgibt.

```
$ a2.py abbccc aaabbc aaaabbbccdefg aaaaaaaaaaaaaaaaaaabb
ab2c3
a3b2c
a4b3c2defg
a20b2
```

**Aufgabe 3.** Schreiben Sie einen Parser, der Dateien im `obj`-Format einliest und ausgibt. Das `obj`-Format besteht aus Zeilen der Form

```
v <x> <y> <z>
```

wobei `<x>`, `<y>`, `<z>` Gleitkommazahlen sind. Es können beliebig viele Whitespaces (`\t\` ) vor und nach den vier Teilen einer Zeile vorkommen. Leerzeilen werden ignoriert, Zeilen mit einer `#` als erstes Zeichen sind Kommentarzeilen und werden ebenfalls ignoriert. Jede Zeile repräsentiert einen Punkt in einem dreidimensionalen Raum. Schreiben Sie eine Funktion `objread(dateiname)`, die eine `obj`-Datei einliest und eine Liste von 3-Tupeln zurückgibt. Zum Beispiel soll die Datei `simple.obj`

```
v 1.0 1.0 1.0
v 3.0 1.0 1.0
v 1.0 5.0 1.0
v 1.0 1.0 7.0
```

die folgende Liste von Punkten ergeben.

```
[(1.0, 1.0, 1.0), (3.0, 1.0, 1.0), (1.0, 5.0, 1.0), (1.0, 1.0, 7.0)]
```

Schreiben Sie ein Skript, das als Kommandozeilenparameter einen Dateinamen erwartet und die Liste von Punkten ausgibt. Sie können davon ausgehen, dass das Format der `obj`-Datei korrekt ist.



**Aufgabe 4.** Wir wollen Bilder der Größe 400x400 aus den eingelesenen Punkten erzeugen und anschauen. Bilder erzeugen Sie mit der Python Imaging Library (PIL) im Modul `Image`.

Mit `im = Image.new("1", (400,400))` erstellen Sie ein neues Bild und binden `im` daran. Mit `im.putpixel((x,y),1)` setzen Sie ein Pixel an den Punkt  $(x,y)$ . Sie können ein Bild speichern mit `im.save(dateiname)`. Verwenden Sie für den Dateinamen die Endung `.png`.

Da Bilder nur zweidimensional sind, müssen wir uns auf zwei der drei Zahlen beschränken. Es gibt drei Möglichkeiten: `x` und `y`, `x` und `z` oder `y` und `z`. Schreiben Sie eine Funktion

```
machebild(punkte, name = "bild", mode="xy")
```

die eine Bilddatei `bild_xy.png` aus den Punkten in `punkte` erzeugt und speichert. Falls `mode` den Wert `"xz"` oder `"yz"` hat, wird das entsprechende Bild erzeugt und unter dem entsprechenden Namen gespeichert. Falls `mode` den Wert `"all"` hat, werden alle drei Bilder erzeugt und gespeichert.

Erstellen Sie ein Skript, das als Kommandozeilenparameter einen Dateinamen, z.B. `d.obj` erwartet und drei Bilder, z.B. `d_xy.png`, `d_xz.png`, `d_yz.png` generiert.

**Aufgabe 5.** Das `obj`-Format kann zusätzlich noch Zeilen der Form

```
f <p1> <p2> <p3>
```

haben, wobei `<p1>`, `<p2>`, `<p3>` Integerzahlen sind und Dreiecke repräsentieren. Jede Zeile mit einem `f` repräsentiert ein Dreieck mit den Eckpunkten `punkte[p1-1]`, `punkte[p2-1]` und `punkte[p3-1]`. Die `pi` sind also die Indizes in die Liste von Punkten beginnend mit 1. Schreiben Sie ein Skript, in der die Funktion `objread` ein Tupel aus der Liste von Punkten (`v`) und aus der Liste der Dreiecke (`f`) zurückgibt, also `(punkte, dreiecke)`.

Um ein Dreieck in ein Bild zu zeichnen verwenden Sie das Modul `ImageDraw`. Das folgende Beispiel zeichnet ein rechtwinkliges Dreieck vom Ursprung.

```
draw = ImageDraw.Draw(im)
draw.polygon([(0,0), (0,10), (10,0)], outline=128)
```

Erweitern Sie die Funktion `machebild`, so dass nicht die Punkte, sondern die Dreiecke gezeichnet werden. Es sollen wieder jeweils 3 Bilder erzeugt werden.

**Hinweis 1.** `import this` – lesen, verstehen, beherrsigen.

<http://www.mi.hs-rm.de/~barth/hsrm/prog3>