

Übungen PdP-Praktikum — Blatt 4

Aufgabe 0: (falls noch nicht geschehen, dies ist die Zusatzaufgabe von Blatt 2)

In dem Ordner aus Aufgabe 1 von Blatt 2 finden Sie eine kleine Sammlung von Bildern im `pbm`-Format. Öffnen Sie eines in einem Texteditor und finden heraus, wie dieses Dateiformat funktioniert (oder wikipedieren sie es oder...)

Finden Sie dann eine Befehlsfolge, die ein solches Bild invertiert (also schwarz wird zu weiß und umgekehrt) und das Ergebnis in eine Datei `invers.pbm` schreibt.

Schreiben Sie dann ein Shellskript, das das tut. Gestalten Sie das Shellskript so, dass es zwei Argumente nutzt: `invert datei.pbm name` soll die existierende `pbm`-Datei `datei.pbm` invertieren und das invertierte Bild in die Datei `name.pbm` schreiben.

Hinweis: Sie können das Programm `gimp` benutzen, um Bilder als `pbm` zu speichern (“Exportieren”) oder um die `pbm`-Bilder zu öffnen und anzusehen (und zu bearbeiten, aber das erfordert etwas Einarbeitung).

Aufgabe 1

Unter der URL

<http://www.math.uni-bielefeld.de/~frettløe/teach/unix/verbrauch.txt>

können Sie eine Datei `verbrauch.txt` herunterladen.

Diese Datei `verbrauch.txt` soll in ein CSV-Format (character separated values) umgewandelt werden um sie mit einem anderen Werkzeug weiter verarbeiten zu können. Geben Sie eine *Pipe* aus dem Befehl `sed` und zwei `tr`-Befehlen an, um

- im Datumsfeld den Monat Januar durch `1.` zu ersetzen
- alle Leerzeichen zu komprimieren und durch ein Komma zu ersetzen.

Beispielausgabe:

```
$ # Pipe-Aufruf über sed und tr
Stadt,Tief,Hoch,Datum
Bielefeld,20332,25235,1.1.2011
Guetersloh,4447,23371,1.1.2011
Paderborn,22491,31014,1.1.2011
Bielefeld,2678,8707,2.1.2011
...
```

Aufgabe 2

Unter der URL

<http://www.math.uni-bielefeld.de/~frettloe/teach/unix/pfade.txt>

können Sie eine Datei mit dem folgenden Inhalt herunterladen:

```
Bielefeld->Gütersloh->Osnabrück->Münster
Gütersloh->Rheda-Wiedenbrück->Warendorf
Münster->Warendorf->Bielefeld->Bad Salzufen
...
```

Ersetzen Sie in dieser Datei alle Umlaute ü durch ue.

Schreiben Sie dann ein Skript, das für eine anzugebende Stadt ermittelt, wie oft sie als Start- und Zielpunkt eines Pfades vorkommt. Beispielaufrufe:

```
$ ./suche-nach.sh Bielefeld
3 Pfade beginnen in Bielefeld.
7 Pfade enden in Bielefeld.
$ ./suche-nach.sh Münster
3 Pfade beginnen in Münster.
3 Pfade enden in Münster.
```

Lösungshinweise: Zur Lösung der Aufgabe benötigen Sie das Ablegen von Werten in Variablen, das Rechnen mit Variablen sowie die Ausgabe von Variablen mit dem `echo`-Befehl. Um innerhalb der Pfad-Datei nach Start- und Zielorten zu suchen, erinnern Sie sich, dass `grep` auch gezielt am Anfang oder am Ende der Zeilen suchen kann. (In dieser Aufgabe werden keine Fallunterscheidungen benötigt.)

Zusatz: Recherchieren Sie, wie Sie in der Shell Zahlen addieren können, und geben Sie auch die Summe der Start- und Zielpunkte aus.

Aufgabe 3

Unter den URLs

<http://www.math.uni-bielefeld.de/~frettloe/teach/unix/verbrauch-max.txt>

<http://www.math.uni-bielefeld.de/~frettloe/teach/unix/verbrauch-min.txt>

können Sie zwei Dateien mit etwa dem folgenden Inhalt herunterladen:

```
      Stadt      Tief
47 Bielefeld    20332
28 Guetersloh   4447
10 Paderborn    22491
...
```

Die ersten Zahlen sollen die Nummer der Messung sein (also z.B. Messung Nr 47). Ihre Aufgabe ist es, die Hochwerte (erste Datei) und Tiefwerte der jeweiligen Messung zusammen mit Nummer der Messung und Stadt in eine Datei `verbrauch-min-max.csv` zu schreiben. Die soll also am Ende etwa so aussehen:

```
1,Guetersloh,33796,14174
2,Paderborn,39644,29395
3,Guetersloh,19318,14097
...
```

Lösungshinweise: Zur Lösung der Aufgabe benötigen Sie die Befehle `sort`, `join`, `cut` und `tr`, siehe die Folien von heute.