

# Übung 2: Textverarbeitung mit der Bash

## 1) Abfahrt

In der Datei `abfahrt.txt` befindet sich das Ranking eines Abfahrtslaufes. Diese Datei hat folgende Spalten: *Rang*, *Nationalität*, *Vorname*, *Nachname* und *Zeit*. Trennzeichen zwischen den einzelnen Feldern sind `<TAB>`s. Der Name und Vorname des Abfahrers sind durch ein Leerzeichen getrennt. Die Zeit wird für alle ab Platz 2 relativ zum Sieger angegeben.

Die Datei `abfahrt.txt` hat folgendes Aussehen:

```
1  N   Kjetil Jansrud  1:40.86m
2  CH  Didier Defago   +0.14s
3  A   Georg Streitberger +0.21s
4  USA Steven Nyman    +0.59s
5  A   Max Franz       +0.61s
6  CH  Patrick Kueng   +0.64s
7  CH  Beat Feuz       +0.66s
8  A   Hannes Reichelt +0.70s
8  F   Guillermo Fayed +0.70s
10 USA Marco Sullivan  +0.75s
11 A   Romed Baumann   +0.77s
12 F   Adrien Theaux   +0.82s
13 A   Klaus Kroell    +0.98s
14 CH  Carlo Janka     +1.05s
15 A   Matthias Mayer  +1.12s
15 F   Johan Clarey    +1.12s
17 I   Peter Fill      +1.16s
18 A   Otmar Striedinger +1.47s
19 USA Travis Ganong   +1.55s
20 I   Werner Heel     +1.62s
21 A   Vincent Kriechmayr +1.64s
22 CAN Benjamin Thomsen +1.84s
23 I   Dominik Paris    +1.87s
24 USA Ted Ligety      +2.09s
25 I   Henri Battilani +2.59s
-   Kanada  Manuel Osborne-Paradis DNF
```

Bearbeiten Sie folgende Fragen mit Hilfe der Kommandos `cut`, `sort`, `head`, `tail`, `uniq` und `wc`.

1. Wieviele Läufer haben an diesem Rennen teilgenommen?

*Lösung:*

```
wc -l abfahrt.txt
```

2. Erstellen Sie in einer neuen Datei `teilnehmer.txt` eine sortierte Liste aller Abfahrer, aufsteigend nach dem Nachnamen sortiert! Sie soll nur Vorname und Nachname enthalten.

*Lösung:*

```
cut -f3 abfahrt.txt | sort -k 2 > teilnehmer.txt
```

3. Geben Sie in einer neuen Datei `podium.txt` eine Liste der ersten drei Abfahrer aus, die Rang, Vorname und Nachname umfassen soll.

*Lösung:*

```
head -n3 abfahrt.txt | cut -f1,3 > podium.txt
```

4. Aus welchen Ländern stammen die Abfahrer? Geben Sie eine sortierte Liste der Länderkurzzeichen in `countries.txt` aus! Jedes Land darf nur einmal vorkommen.

*Lösung:*

```
cut -f2 abfahrt.txt | sort | uniq > countries.txt
```

5. Geben Sie Liste aus, die angibt, wie viele Abfahrer aus den jeweiligen Ländern stammen. Die Liste soll Anzahl der Fahrer und Land enthalten und absteigend nach Anzahl sortiert sein.

*Lösung:*

```
cut -f2 abfahrt.txt | sort | uniq -c | sort -n -r
```

6. Aus welchem Land kommen die meisten Abfahrer?

*Lösung:*

```
cut -f2 abfahrt.txt | sort | uniq -c | sort -n -r | head -n1
```

7. Welcher Abfahrer wurde disqualifiziert? Nachname und Vorname sind auszugeben!

*Lösung:*

```
tail -n1 abfahrt.txt | cut -f3
```

8. Welcher Abfahrer wurde als Dritter klassifiziert? Geben Sie nur seinen Namen aus.

*Lösung:*

```
head -n3 abfahrt.txt | tail -n1 | cut -f3
```

## 2) Filme

Folgende Datei `movies.txt` ist gegeben. Die Datei enthält folgende Felder: *Filmtitel*, *Genre*, *Erscheinungsdatum* und eine *Punktwertung*, die von Kinobesuchern nach Ansehen des jeweiligen Films vergeben wurde. Die einzelnen Felder sind durch das Trennzeichen `|` von einander getrennt.

Die Datei `movies.txt` hat folgendes Aussehen:

```
Die Verurteilten|Crime|1994|92
Die 12 Geschworenen|Drama|1957|89
Pulp Fiction|Crime|1994|89
The Lord of the Rings: The Return of the King |Adventure|2003|89
Zwei glorreiche Halunken|Western|1966|89
Fight Club |Drama|1999|88
Forrest Gump|Drama|1994|87
The Dark Knight|Action|2008|86
Der Pate|Crime|1972|92
Der Pate 2|Crime|1974|90
Schindlers Liste|History|1993|89
```

Bearbeiten Sie folgende Fragen mit Hilfe der Kommandos `cut`, `sort`, `head`, `tail`, `uniq` und `wc`.

1. Geben Sie die letzten drei Filme der Datei `movies.txt` aus!

*Lösung:*

```
tail -n3 movies.txt
```

2. Erzeugen Sie eine nach Erscheinungsdatum absteigend sortierte Liste der Datei `movies.txt` und speichern Sie diese in einer neuen Datei `year.txt`!

*Lösung:*

```
sort -k3r -t '|' movies.txt > year.txt
```

3. Welcher Film ist der jüngste? Geben Sie nur den Filmtitel aus!

*Lösung:*

```
sort -k3r -t '|' movies.txt | cut -f1 -d '|' | head -n1
```

4. Geben Sie ein Ranking mit Filmtitel, Genre und Wertung der letzten drei Filme aus, sodass folgendes Listing entsteht:

```
The Dark Knight |Action|86  
Forrest Gump|Drama|87  
Fight Club|Drama|88
```

*Lösung:*

```
sort -k4 -t '|' movies.txt | cut -f1,2,4 -d '|' | head -31
```

5. In welchem Genre gibt's die meisten Filme? Geben Sie Anzahl und Genre aus!

*Lösung:*

```
cut -f2 -d '|' movies.txt | sort | uniq -c | sort | tail -11
```