

Aufgabe 1

Erstellen Sie ein Skript, dem Sie einen Dateinamen als Parameter übergeben können. Falls die Datei existiert und lesbar ist, bietet das Skript eine Auswahl an Aktionen zur angegebenen Datei an. Im einzelnen soll es den Dateiinhalt, den Dateityp, die Dateieigenschaften oder die Dateigröße anzeigen können (siehe `file`, `stat`, `du`). Bei einer ungültigen Auswahl soll eine Fehlermeldung ausgegeben werden.

Eine Beispielausgabe:

```
Die Datei hallo_welt.c existiert und ist lesbar.
```

```
1 um den Dateiinhalt auszugeben,  
2 um den Dateityp anzuzeigen,  
3 um die Dateieigenschaften anzuzeigen,  
4 um die Dateigroesse anzuzeigen.
```

Auswahl:

Aufgabe 2

- (a) Schreiben Sie ein Skript, welches überprüft, ob genau 2 Parameter angegeben wurden. Sind es weniger oder mehr als 2 Parameter soll eine entsprechende Fehlermeldung ausgegeben werden. (z.B.: „Fehler: zu viele Parameter.“)
- (b) Erweitern Sie Ihr Skript, so dass bei genau 2 Parametern überprüft und ausgegeben wird, ob es sich bei den Parametern (oder einem davon) um eine Datei oder ein Verzeichnis im HOME-Verzeichnis handelt. Wenn ja, soll eine entsprechende Meldung ausgegeben werden.
(z.B. „`readme` existiert in `/home/94711`“)

Aufgabe 3

Schreiben Sie ein kleines Ratespiel, bei dem der Benutzer eine Zahl zwischen 0 und 10 erraten soll. Er hat hierfür unendlich viele Versuche. Das Skript soll ihn nach Fehlversuchen unterstützen, indem es eine Ausgabe mit „Geheime Zahl ist größer“ oder „Geheime Zahl ist kleiner“ macht. Wenn der Benutzer richtig getippt hat, wird ihm gratuliert und das Skript beendet sich.

Hilfestellung: Eine Zufallszahl „zahl“ zwischen 0 und 10 kann folgendermaßen erzeugt werden: `zahl=$((expr $RANDOM % 11))`

Zusatzaufgabe: Schreiben Sie eine Variante mit begrenzt vielen Rateversuchen.

Aufgabe 4

Erstellen Sie eine Shellfunktion `add()`, welche zwei Zahlen addiert und die Rechnung sowie die Lösung (z.B. „`5 + 3 = 8`“) auf dem Bildschirm ausgibt (siehe `man expr`).