



**Aufgabe 1.** Gegeben sei eine Datei, in der ausschließlich Zahlen stehen. In der ersten Zeile stehe eine natürliche Zahl, die angibt wie viele Zahlen noch folgen.

- a) Schreibe ein Programm, das diese Datei einliest, die Zahlen sortiert und die Datei mit der sortierten Liste überschreibt.
- b) Modifiziere dein Programm nun so, dass in der ersten Zeile nicht mehr stehen muss, wie viele Zeilen noch folgen.

**Aufgabe 2.** Erstelle ein Programm `uniq`, das zwei Dateinamen als Argumente erhält, die erste Datei wortweise<sup>1</sup> einliest und ohne doppelte Einträge in die zweite Datei schreibt. Es soll auch möglich sein, dass Ein- und Ausgabedatei die gleiche sind.

**Aufgabe 3.** Werde mit den Aufgaben von den vorherigen Zetteln fertig.

**Aufgabe 4.** Lies im Skript noch den Rest vom Kapitel über Dateieingabe und -ausgabe, einige nützliche Funktionen wirst du dort noch finden. Schreibe dann ein Programm, das ein Labyrinth aus einer Datei einliest:

```
XXXXXXXXXXXXXXXXXX
X X XXXXXXXXXXXX*X
X$X XX      XXX X
X X XX  XXX  XXX X
X   XX  XXX  XXX X
XXX X   XX  XXX X
XXX   X      X
XXXXXXXXXXXXXXXXXX
```

*Bemerkung:* Wir spezifizieren das Labyrinth hier nicht viel näher, entscheide dich selbst vorher für ein Format. Soll die Größe des Labyrinths variabel sein oder fest? Soll die Größe in der ersten Zeile der Datei stehen oder nicht? Soll das Labyrinth quadratisch sein oder nicht? Soll es außen herum immer mit `X`en begrenzt sein oder hast du vielleicht eine andere Lösung?

Flip me.

<sup>1</sup>Worte sind alles, was durch Whitespace (Leerzeichen, Zeilenumbrüche und Tabulatoren) getrennt ist.



Das Programm soll einen Weg vom Startpunkt (dem Stern) zum Schatz (dem Dollarzeichen) finden. Die *X*e sind Wände und Leerzeichen sind Pfade. Markiere einen Weg mit Punkten und gebe das Labyrinth mit Weg in der Konsole aus.

```
XXXXXXXXXXXXXXXXXX
X X XXXXXXXXXXXX*X
X$X XX      XXX.X
X.X XX XXX XXX. X
X...XX XXX XXX. X
XXX.X...XX XXX. X
XXX...X.....X
XXXXXXXXXXXXXXXXXX
```