

## Aufgabe 2: wsort (Abgabe bis Mi. 14.06.06 10:00 Uhr)

Schreiben Sie ein Programm **wsort**, welches eine Liste von Wörtern (jedes Wort steht in einer eigenen Zeile!) vom Standard-Eingabekanal (stdin) einliest, diese Liste alphabetisch sortiert und die sortierte Liste auf dem Standard-Ausgabekanal (stdout) wieder ausgibt (wieder ein Wort pro Zeile).

Gehen Sie bei der Lösung der Aufgabe in folgenden Schritten vor:

- a) Schreiben Sie eine Funktion

```
int readwords(char feld[]);
```

die die Wörter am besten Zeichen für Zeichen einliest und sie in dem übergebenen char-Feld abspeichert.

Sie können davon ausgehen, dass alle Wörter zusammen nicht mehr als 10.000 Zeichen umfassen und das Feld entsprechend dimensionieren.

Die Funktion `readwords` soll darauf achten, dass sie in keinem Fall über das Ende des Feldes hinausschreibt, selbst wenn sie mehr als 10.000 Zeichen übergeben bekommt.

`readwords` liefert als Ergebnis die Zahl der eingelesenen Wörter zurück.

- b) Allokieren Sie nun mit Hilfe von **malloc(3)** ein Feld `words`, das für jedes Wort einen Zeiger auf den Wortanfang enthalten soll und setzen Sie die Zeiger in dem Feld entsprechend. Jedes Wort muss jetzt ein \0-terminierter String sein!
- c) Sortieren Sie `words` nun mit Hilfe der Funktion **qsort(3)** aus der ANSI-C-Bibliothek (Hinweise hierzu in den Erläuterungsfolien zur 2. Übung!).
- d) Schreiben Sie schließlich eine Funktion

```
void printwords(char *words[]);
```

die die sortierten Wörter mit Hilfe von **printf(3)** ausgibt..