

Aufgabe 2: wsort (12 Punkte) Bearbeitung **Einzeln!** Keine Zweiergruppen! 05.11.2009

a) Makefile

Erstellen Sie ein einfaches Makefile. Durch den Aufruf von “make” oder “make all” soll Ihr Programm übersetzt werden, mit “make install” das übersetzte Programm aus dem Verzeichnis `src` in das Verzeichnis `bin.i386` kopiert werden und mit “make clean” sollen alle von “make” erzeugten Dateien im Verzeichnis `src` gelöscht werden.

b) Standard-Eingabe sortieren

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 (ein Wort pro Zeile).

In den Dateien `/proj/i4sp/pub/aufgabe2/wlist*` finden Sie Beispiel-Eingabedateien. Mit dem Programm `/proj/i4sp/pub/aufgabe2/wsort` können Sie diese Dateien sortieren und die Ausgabe jeweils mittels **diff(1)** mit der Ausgabe Ihres `wsort`-Programms vergleichen.

Zum Sortieren in Ihrem Programm verwenden Sie die Funktion **qsort(3)** aus der ANSI-C-Bibliothek. Mit der Funktion **strcmp(3)** können Sie den alphabetischen Vergleich zweier Worte durchführen. Mit Hilfe von **malloc(3)** und **realloc(3)** können Sie dynamisch Speicher an- und nachfordern, um die benötigten Datenstrukturen anzulegen bzw. zu erweitern.

c) Kommandozeilenargumente sortieren

Ergänzen Sie das Programm nun so, dass es die Argumente, die an `wsort` übergeben werden, sortiert und auf `stdout` ausgibt. Werden keine Argumente übergeben, soll sich das Programm wie unter (b) verhalten, ansonsten werden nur die Argumente sortiert, keine Daten von der Standardeingabe.

d) Dokumentation

In einer Datei **wsort.txt** in dem Verzeichnis `doc` ist außerdem die Vorgehensweise und der Aufbau der Datenstrukturen zum Aufruf von **qsort(3)** knapp aber präzise und vollständig zu beschreiben. **(Nicht den Quick-Sort-Algorithmus selbst, sondern was man mit den Eingabedaten machen muss, um die qsort-Funktion mit Daten zu versorgen.)**

Hinweise:

- Das C-Programm ist in der Datei **wsort.c** abzulegen. Unterprogramme und Variablendefinitionen sind ausreichend zu kommentieren. Achten Sie bitte außerdem auf saubere Gliederung des Quellcodes.
- Ein Wort enthält alle Zeichen einer Zeile. Zeilen sind durch Newline-Zeichen (`'\n'`) voneinander getrennt, die selbst nicht Teil des Wortes sind. Jede Zeile endet mit einem Newline-Zeichen, mit Ausnahme der letzten Zeile. Enthält die letzte Zeile Zeichen, so ist diese ebenfalls als Wort zu behandeln.
- Sie können davon ausgehen, dass die Wörter eine maximale Länge von 100 Zeichen nicht überschreiten. Anderenfalls soll eine Fehlermeldung ausgegeben und das Wort ignoriert werden.
- Leere Zeilen sind ohne Fehlermeldung zu ignorieren.
- Beim Sortieren von großen Dateien hängt es vom Aufbau des Programms ab, wie schnell `wsort` die Daten sortiert. Sie können mit dem Programm **time(1)** die benötigte Ausführungszeit messen und mit anderen Lösungen vergleichen. Die schnellste Lösung wird mit einem Kasten Siegerbier prämiert. Die Regeln des Wettbewerbs finden Sie in den Folien zu Übung U2.

Abgabe: bis spätestens Dienstag, 17.11.2009, 17:30 Uhr