

5. Übung am 22.11.2010

=====

Anleitung:

Während jeder Übung ist von jeder Gruppe ein (kurzes) Protokoll zu erstellen. Das Protokoll ist eine einfache ASCII-Text-Datei, die mit einem Text-Editor (z.B. gedit, kate) mit dem Sie auch Ihre Programme schreiben, erzeugt wird. Nennen Sie diese Datei unbedingt "PROTOKOLL.txt".

Das Protokoll muss Folgendes enthalten:

1. Datum, Übungsnummer, Gruppennummer, Name(n) der mitwirkenden Studierenden,
2. Benötigter Zeitaufwand für die gestellten Aufgaben (circa),
3. Namen der erstellten Programme (KEINE Listings),
4. Kurze Antwort auf eventuell weiter unten gestellte Fragen,
5. Eventuelle Probleme oder Besonderheiten, falls diese aufgetreten sind.

Sämtliche während der Übung erstellten Dateien (Protokoll, Source Codes, ausführbare Programme, etc.) verbleiben im Verzeichnis für den jeweiligen Übungstag, also z.B. "05Ue2010-11-22/" Ihrer Gruppe.

Das Protokoll und die Übungsprogramme sollten am jeweiligen Übungsnachmittag erstellt werden, spätestens jedoch bis zum nächstfolgenden Übungstag (Montag), 14:00! (Spätere Ausarbeitungen können nur in begründeten Fällen und nach Rücksprache mit Ihrem Betreuer bzw. Tutor berücksichtigt werden!)

=====

Aufgaben zu Kapitel 5 (3 Punkte):

1. Verwenden Sie ein Datenfile nach folgendem Muster:

- a. eine beliebige Anzahl (≥ 1) von Textzeilen, jeweils beginnend mit dem Zeichen '#'
- b. gefolgt von einer beliebigen Anzahl (≥ 1) von Datenzeilen, jeweils bestehend aus 2 Gleitkommazahlen, getrennt durch eine beliebige Anzahl von Blanks.

z.B.: siehe die mitgelieferte Datei "Pt111.tbl":

```
-----  
# CALIBRATION TABLE  
# Sensor:    Pt111  
# SerialNo.: U645892  
# ...  
175.84      6.  
212.02      7.  
247.04      8.  
280.90      9.  
...  
-----
```

2. Schreiben Sie ein C-Programm, das folgende Aufgaben erfüllt:

- i. Ausgeben und zählen der Textzeilen in der Datei "Pt111.tbl"
- ii. Suchen des größten und kleinsten x-Wertes (linke Spalte) aus den Datenzeilen

Ausgabe der Min/Max-Werte auf "stdout"

Ausgabe der gesamten Anzahl der Datenzeilen auf "stdout"

Dafür benötigen Sie wahrscheinlich folgende Funktionen:

fopen, fgets, fscanf, sscanf, printf, fclose

Online-Hilfe bekommen Sie für C-Funktionen mit "man Funktionsname".

Beachten Sie beim "Verarbeiten" einer Datei Folgendes:

Daten-Files kommen meist von einem Benutzer. Allgemein gilt:

Ein Benutzer kann Ihrem Programm absichtlich oder unabsichtlich korrupte Daten liefern. Verarbeiten Sie daher NIEMALS ungeprüfte Benutzereingaben in einem ernsthaften Programm!

Prüfen und aufbereiten von Benutzereingaben machen oft mehr als die Hälfte eines Programms aus.

Beachten Sie für später:

Ein "böser" Benutzer kann durch korrupte Daten, die ungeprüft übernommen werden, Ihr gesamtes System kompromitieren!

Im speziellen Beispiel:

Hier ist es NICHT Ihre Aufgabe, die Eingabe-Datei detailliert zu analysieren. Prüfen Sie, ob die Datei im Wesentlichen den gegebenen Anforderungen entspricht. Ist das nicht der Fall, geben Sie eine möglichst aussagekräftige Fehlermeldung aus und brechen Sie die weitere Verarbeitung ab.

Fehlermeldungen: in C immer auf "stderr" ausgeben.

"Normale" Ausgaben: auf "stdout" ausgeben.

Dazu können folgende Funktionen nützlich sein:

`perror, fprintf(stderr, "..."), exit;`

"Füttern" Sie Ihr Programm mit fehlerhaften Dateien, beobachten Sie, was geschieht.

Fragen: entfallen