



Systemnahe Software I (WS 2012/2013)

Abgabe bis zum 17. Januar 2013, 16:00 Uhr

Lernziele:

- Funktionszeiger verwenden
- Einlesen aus Dateien (`fopen`, `fscanf`, `fclose`)

Aufgabe 14: Text vergleichen (10 Punkte)

Das hier gesuchte Programm soll zwei Texte zeilenweise vergleichen, indem es diejenigen Zeilen, die es als unterschiedlich erkennt, auf der Standardausgabe ausgibt.

Stellt Euch vor, Ihr habt viele Funktionen von der Bauart

```
void bearbeite(char* str)
```

zur Verfügung. Wie kann man einen Zeiger auf eine solche Funktion deklarieren, zuweisen und benutzen?

Diese Funktionen wollen wir nun vor dem Vergleich zeilenweise auf den Text anwenden, bevor wir den zeichenweisen Vergleich ausführen; sie werden im Programm als *Text-Filter* benutzt, die durch Kommandozeilenoptionen eingeschaltet werden können. Folgende Filter sollen eingebaut werden:

<i>Option</i>	<i>Filter</i>
-c	Groß-/Kleinschreibung ignorieren
-w	Whitespace (<code>isspace</code> aus <code>ctype.h</code>) in einer Zeile ignorieren
-s	jegliche Ansammlung von Whitespace als ein Leerzeichen betrachten (wie z. B. der Präprozessor es tut)
-a	nur alphanumerische Zeichen (<code>isalnum</code>) betrachten

Dabei sollen die Filter beliebig kombiniert werden können. Um dies zu ermöglichen, sollt Ihr zu Beginn Eures Programms die Kommandozeilenoptionen in eine Liste verwandeln, deren Elemente jeweils den Zeiger auf eine auszuführende Funktion beinhalten. Danach könnt

Ihr dann die beiden Dateien Zeile für Zeile durchgehen und alle Funktionen der Liste hintereinander darauf anwenden bevor Ihr den jeweiligen Vergleich (mit `strcmp`) durchführt und unterschiedliche Zeilen ausgeben.

Beispiel:

```
thales$ gcc -std=gnu99 -Wall compare.c -o compare
thales$ cat a.txt
1 A
2 B
3 c
thales$ cat b.txt
1 A
2B
3    C
thales$ ./compare a.txt b.txt
a.txt:2: 2 B
b.txt:2: 2B
a.txt:3: 3 c
b.txt:3: 3    C
thales$ ./compare -s -c a.txt b.txt
a.txt:2: 2 b
b.txt:2: 2b
thales$ ./compare -c -w a.txt b.txt
thales$ echo in a steht mehr >> a.txt
thales$ ./compare -c -w a.txt b.txt
premature end of b.txt
```

Im Beispiel sieht man, wie die als unterschiedlich erkannten Zeilen ausgegeben werden, je nachdem, welche Filter eingeschaltet wurden. Wird beim Lesen einer der beiden Dateien das Ende früher erreicht, soll der Vergleich beendet und eine entsprechende Meldung ausgegeben werden (siehe letzte Zeile des Beispiels).

```
submit ssl 14 compare.c team
```

Viel Erfolg!