

Imperative Programmierung

Aufgabenblatt 4

Hinweise:

Bearbeitungszeit: 2 Wochen, Abgabefrist: Siehe Juniorstud.IP

Ladet eure Lösungen der Aufgaben unter "Aufgaben" im Juniorstud.IP als PDF-Dokument bzw.

C-Quelltext oder als txt-Datei hoch. Aufkommende Fragen bitte im Forum stellen, damit diese dort für alle geklärt werden können. Bei verspäteter Abgabe wird je Tag 10% der zu erreichenden Gesamtpunktzahl abgezogen.

1. Das Programm zum Zählen von Zeilen aus der Vorlesung (`lcount.c`, Skript Seite 52) ignoriert eine unvollständige letzte Zeile (d. h., eine Buchstabenfolge, die vor dem EOF nicht mit einem Zeilenende `\n` abgeschlossen wird).

Erweitert das Programm so, dass (1) die unvollständige Zeile (falls vorhanden) mitgezählt wird und (2) in diesem Fall eine zusätzliche Meldung der Form "letzte Zeile unvollständig mit n Zeichen" ausgegeben wird.

10 Punkte

2. Entwickelt ein Programm, das Zeichenfolgen in einer Datei klassifiziert und zählt: Worte sind Folgen von Buchstaben, Zahlen sind Folgen von Ziffern und gemischte Sequenzen sind von Folgen von Ziffern und Buchstaben. Für die Datei

```
This are not! "17" and<\n>
also_not 42, 399 or 12 words<\n>
this123is mixed and 98this126 too<\n>
and the last!\n
sequence is incomplete<EOF>
```

(wobei Zeichen in spitzen Klammern für entsprechende Sonderzeichen stehen) soll dann beispielsweise folgende Ausgabe geliefert werden:

```
122 Zeichen
17 Worte
4 Zahlen
2 gemischte Sequenzen
27 sonstige Zeichen.
Letzte Zeichenfolge unvollstaendig.
```

Das heißt, ihr Programm muss auch die Zeichen zählen, die zu keiner der klassifizierten Folgen gehört und es muss erkennen, ob eine Folge vom Dateiende beendet wird.

- Findet heraus, was `isdigit` und `isalpha` sind.
5 Punkte
- Zeichnet ein Zustandsdiagramm das definiert, wie euer Programm die Zeichen verarbeitet (wir hatten so etwas für das Programm `wcount.c` (Skript Seite 55) in der Vorlesung an an der Tafel entwickelt.)
5 Punkte

- Entwickelt das Programm.
20 Punkte