## Formale Methoden der Softwaretechnik

Till Mossakowski, Christoph Lüth
Sommersemester 2011

Universität Bremen
FB 3 — Mathematik und Informatik

# Übungsblatt 3

**Abgabe: 18.07.2011**

**Aufgabe 3.1** (Texte formatieren)
Gegeben sei folgende Spezifikation:

```
The program's input is a stream of characters whose end is signalled
with a special end-of-text character, ET. There is exactly one ET
character in each input stream. Characters are classified as:

* break characters - BL (blank) or NL (new line)
* non-break characters - all others except ET;
* the end-of-text indicator - ET.

A word is a nonempty sequence of non-break characters. A break is a
sequence of one or more break characters. Thus, the input can be
viewed as a sequence of words separated by breaks, with possibly
leading and trailing breaks, and ending with ET.

The program's output should be the same sequence of words as in the (J.
input, with the exception that an oversize word (i.e. a word
containing more than MAXPOS characters, where MAXPOS is a positive
integer) should cause an error exit from the program (i.e. a variable,
Alarm, should have the value TRUE). Up to the point of an error, the
program's output should have the following properties:

a) A new line should start only between words and at the beginning
  of the output text, if any.
b) A break in  the input is reduced  to a single break character in
  the output.
c) As many words as possible should be placed on each line
  (i.e. between successive NL characters).
d) No line may contain more than MAXPOS characters (words and BLs).
```

Goodenough, S. Gerhart: *Towards a theory of test: Data selection criteria*, in Current Trends in Programming Methodology Vol. 2, T. Yeh (ed.), Prentice-Hall 1977, pp. 44-79.)

(i) Gebe fünf Probleme in der Spezifikation an. Erläutere das Problem, eventuell mit einem Gegenbeispiel, und klassifiziere es anhand der folgenden "Sieben Todsünden des Spezifizierers" (B. Meyer):

**Noise:** *Unnecessary text that diverts the attention of the reader.*

**Remorse:** *Restrictions to a specification element made at the point where the element is used and not where it is defined.*

**Silence:** *Omission of specification elements which are considered "obvious" by the specifier.*

**Contradiction:** *Inconsistent statements made in the specification.*

**Over-specification:** *Description of a solution instead of pointing out the problem.*

**Ambiguity:** *Statements that can be interpreted in several ways.*

**Forward references:** *Usage of a concept before its proper definition.*

(ii) Formuliere (natürlichsprachlich) eine funktionale Spezifikation des Problems, analog zu Aufgabe 3.2.

(iii) Formuliere diese Spezifikation formal in Isabelle.

(iv) Gebe eine Implementation in Isabelle an.

(v) Beweise, dass die Implementation die Spezifikation erfüllt.

Die folgende Aufgabe wird in der Übung am 20.06.2011 gemeinsam gelöst und braucht nicht abgegeben zu werden:

**Aufgabe 3.2** (Worte zählen)

Gegeben folgende funktionale Spezifikation:

> The program should read in a text and produce a count of the words in the text in the form of pairs of words occuring in the text, and their frequency, sorted by descending frequency. Here, a text is a sequence of characters, and a word a sequence of alphanumeric characters separated by spaces, punctuation, or other non-alphanumeric characters.

(i) Formuliere diese Spezifikation formal in Isabelle.

(ii) Gebe eine Implementation in Isabelle an.

(iii) Beweise, dass die Implementation die Spezifikation erfüllt.

Dies ist Revision 1491 vom 2011-06-20.