UNIVERSITÄT LEIPZIG

LPZ E-BUSINESS applied telematics

Übungsblatt 1

Abgabe: Vorlesung am 07.11.2005

1 Eigenschaften von Algorithmen

In der Fakultät für vergleichende Textilwissenschaften stehen die Studenten vor dem Sekretariat Schlange, um sich in Übungsgruppen einteilen zu lassen. Dabei wird folgender Algorithmus zur Aufteilung auf die Übungsgruppen 1...*n* eingesetzt:

→ **falls** vor dem Sekretariat eine Schlange von Studenten[1] steht
        dann erfrage die Matrikelnummer des ersten Studenten in der Schlange
                dividiere die Mat.-Nr. durch *n* und betrachte den ganzzahligen Rest *r*
                falls *r* = 0 ist
                        dann ordne den Studenten in Übungsgruppe *n* ein
                        sonst ordne den Studenten in Übungsgruppe *r* ein
                schicke den Studenten in die Mensa
                springe zur mit → markierten Operation
        sonst gehe selbst in die Mensa

1. Welche Bedeutung hat die Einrückung der Operationen im Textsatz? Wird durch ihre (Nicht-)Berücksichtigung der Sinn des Algorithmus verändert? Gehen Sie im Folgenden von der sinnvolleren Version des Algorithmus aus. *(1 Punkt)*

2. Ist der Algorithmus abstrakt, finit, terminierend, deterministisch, determiniert? Begründen Sie Ihre Einschätzung jeder Eigenschaft am obigen Beispiel und beschreiben Sie, wie das Beispiel verändert werden müsste, um die jeweils gegenteilige Eigenschaft zu erfüllen. *(5 Punkte)*

2 Formulierung von Algorithmen

Im nächsten Semester sollen Studenten die Möglichkeit bekommen, Wünsche für die Übungsgruppeneinteilung zu äußern: In der ersten Vorlesung muss jeder der m_{Semester} Studenten eine Wunschgruppe mit hoher und eine weitere mit niedriger Priorität angeben.

Nachdem alle Wünsche abgegeben wurden, verteilt das Sekretariat die Studenten unter Berücksichtigung ihrer Wünsche auf die angebotenen Übungsgruppen 1...*n*,

---

[1] In Anlehnung an Pierre Daninos *(„Jeder Engländer ist der Anfang einer Schlange")* bestehe eine Schlange aus einem oder mehreren Studenten.

wobei in jeder Übungsgruppe maximal m_{Gruppe} Studenten untergebracht werden können (dabei sei $m_{\text{Gruppe}} << m_{\text{Semester}}$).

Formulieren Sie einen Algorithmus, anhand dessen das Sekretariat eine möglichst wunschgetreue[2] Aufteilung der Studenten auf die Übungsgruppen vornehmen kann. *(4 Punkte)*

3 Äquivalenzproblem

A sei die Menge aller Algorithmen. Die Funktion g: $\mathbf{A} \times \mathbf{A} \rightarrow \{\texttt{true}, \texttt{false}\}$ liefere für $g(a_1, a_2) = \texttt{true}$, falls die Algorithmen a_1 und a_2 die gleiche Funktion berechnen (also für beliebige Eingaben das gleiche oder kein Resultat berechnen). In allen anderen Fällen liefere g das Ergebnis `false`.

Diese Aufgabenstellung wird als Äquivalenzproblem bezeichnet, denn die beschriebene Funktion g ist nicht berechenbar.

1. Für die Nichtberechenbarkeit von g wurde folgender Beweis vorgeschlagen:

   „Wäre g berechenbar, so müsste es einen Algorithmus f geben, der die Ausgaben von a_1 und a_2 bei identischer Eingabe vergleicht. Dazu müsste f insbesondere feststellen, für welche Eingaben a_1 bzw. a_2 überhaupt zu einem Ergebnis kommt (also hält) und für welche Eingaben nicht. Das bedeutet, f müsste für beliebige Algorithmen a_1 und a_2 feststellen können, ob sie bei einer beliebigen Eingabe x halten oder nicht. Aus dem in der Vorlesung besprochenen Beweis für das Halteproblem wissen wir aber, dass es einen solchen Algorithmus nicht geben kann. Somit ist das Äquivalenzproblem auf das Halteproblem zurückgeführt, und einen Algorithmus f mit der angegebenen Eigenschaft kann es nicht geben."

   Begründen Sie, warum dieser Beweis nicht korrekt ist. *(2 Punkte)*

2. Geben Sie einen korrekten Beweis dafür an, dass g nicht berechenbar ist. *(6 Punkte)*

4 Turing-Programm

Die Theorie zeigt, dass auch eine Turing-Maschine mit nur zwei Zeichen (d.h. mit dem Bandalphabet $\mathbf{B} = \{a, z\}$, wobei z das Leerzeichen ist) bei entsprechend mehr Zuständen grundsätzlich die gleiche Mächtigkeit hat wie jede andere.

Formulieren Sie das Turing-Programm zur Subtraktion zweier natürlicher Zahlen aus der Vorlesung so um, dass es auch ohne das Zeichen b auskommt. *(8 Punkte)*

Abgaben bitte mit Namen, Matrikelnummer und Übungsgruppennummer versehen!

[2] Beachten Sie die Formulierung „möglichst wunschgetreu"! Einfach irgendeine Aufteilung zu finden, ist nicht ausreichend. Machen Sie sich Gedanken, wie Sie die Formulierung „möglichst wunschgetreu" interpretieren und in Ihrem Algorithmus realisieren können. Wie so oft in der Praxis der Softwareentwicklung, gibt es verschiedene Möglichkeiten, diese Anforderung zu deuten.