Einführung Informatik/Algorithmen und Datenstrukturen WS 97/98
Prof. Dr. R. Kruse / I. Blümel / C. Borgelt

Spielregeln:

1. Die Übungen werden mit einem Schein abgeschlossen. Dieser dient gleichzeitig als Zulassungsvoraussetzung für die 4stündige Klausur im Sommer 1998.
2. Organisatorische Regelungen für den Übungsschein:
   a) Zu den Übungen gibt es Aufgabenblätter. Diese sind im WWW abrufbar. Es müssen (bis 1h vor jeder Übung per e-mail) von jedem Studenten Aufgaben votiert werden. Zu den votierten Aufgaben ist der Student bereit, vorzutragen (Lösungsvorschläge werden diskutiert, sie müssen nicht gleich korrekt sein!).
   b) Am Rechner werden ein Testat und ein Beleg abgenommen.
   c) Wer **mindestens**
      - *2/3* der Übungsaufgaben votiert,
      - *viermal* in den Übungen vorgetragen,
      - das *Testat* erfüllt hat und
      - den *Beleg* erfolgreich verteidigt hat,

      bekommt den Schein.
3. Zusätzlich zur Vorlesung und Übung werden Tutorien angeboten. Der Besuch dieser Veranstaltung ist freiwillig und dient
   - dem Erlernen des Programmierens
     * am Unix-Rechner in C (Informatiker und Wirtschaftsinformatiker) bzw.
     * in Smalltalk (VisualWorks) unter UNIX oder Windows (Computervisualisten).
   - Die Tutorien werden nicht bewertet.

Aktuelle Hinweise zur Lehrveranstaltung und die Übungsblätter finden Sie unter:
http://fuzzy.cs.uni-magdeburg.de/studium/ads.html

1. Übungsblatt
Algorithmische Grundbegriffe

Aufgabe 1

Skizzieren Sie alle nicht-deterministischen Möglichkeiten, die folgende Formel schrittweise auszuwerten:

2 * (25 + 3 * (4 + 5)) - 314 - 4 * 63

Aufgabe 2

Ein Bonbonglas ist mit Lakritz und Karamel-Bonbons gefüllt. Außerdem liegt neben dem Glas eine "Wundertüte", die mit enorm viel Lakritz gefüllt ist. Der folgende Vorgang soll nun solange ausgeführt werden, bis nur noch eine einzelne Süßigkeit im Glas übrig ist:

- Entnehmen Sie zwei beliebige Süßigkeiten aus dem Glas.
- Falls beide Bonbons die gleiche Geschmacksrichtung haben, essen Sie beide auf und füllen ein Lakritz aus der nebenstehenden Tüte in das Glas.
- Haben Sie ein Lakritz und ein Karamel gezogen, dürfen Sie nur das Lakritz naschen. Das Karamel-Bonbon müssen Sie wieder in das Glas zurücklegen.

Terminiert dieser Vorgang? Wenn ja, welche Aussage kann allgemein über die letzte im Glas zurückbleibende Süßigkeit getroffen werden?

Hinweis: Wir empfehlen, eine genügend große Anzahl von Karamel- und Lakritz-Bonbons zur praktischen Demonstration Ihrer Lösung mitzubringen. Evtl. kann man auch gelbe und grüne Gummibärchen nutzen (je nach Geschmack!).

Aufgabe 3

In der Vorlesung wurden Operationen für den Datentyp *integer* angegeben. Geben Sie nun analog dazu fünf unterschiedliche Operationen und ihre Bedeutung für den Datentyp *string* an.

Aufgabe 4

Bisher wurden in der Vorlesung vorwiegend terminierende Algorithmen behandelt. Es gibt durchaus auch Beispiele für sinnvolle, nicht-terminierende Algorithmen, z. B. die Steuerung von Ampelanlagen, die Überwachung einer Intensivstation oder auch die Berechnung der Zahl e.

- Geben Sie drei weitere sinnvolle Beispiele für nicht-terminierende Algorithmen an.
- Wie kann der Nicht-Terminismus bei der Programmierung derartiger Algorithmen aufgelöst werden? Gehen Sie dabei auf die Berechnung der Zahl e nach einer der folgenden Varianten ein:

$$e = \sum_{n=0}^{\infty} \frac{1}{n!},\ e = \lim_{n\to\infty}\left(1-\frac{1}{n}\right)^{-n},\ e = \lim_{n\to\infty}\left(1+\frac{1}{n}\right)^{+n}$$

Aufgabe 5 (Programmieraufgabe)

Es ist ein Programm zu entwerfen, das drei ganze Zahlen einliest und verarbeitet. Die Summe, das Produkt, das arithmetische Mittel und die größte der drei Zahlen sind auszugeben.

Aufgabe 6 (Programmieraufgabe)

C kennt die drei Arten der Wiederholanweisung:

while (*ausdr*) *anw*;
do *anw* while (*ausdr*);
for(*Ini-Ausdr; Abbruchkrit; ausdr*) *anw*;

Realisieren Sie für die Summation von 1+5+9+13+ ... +1001 durch Anwendung dieser drei Schleifen je eine Variante.

Aufgabe 7 (Programmieraufgabe)

Nach dem Gregorianischen Kalender (gilt seit 1583) wird der Wochentag aus einem beliebigen Datum nach folgender Berechnungsvorschrift ermittelt:

$$T = J + \left\lfloor \frac{J-1}{4} \right\rfloor - \left\lfloor \frac{J-1}{100} \right\rfloor + \left\lfloor \frac{J-1}{400} \right\rfloor + D$$

J: Jahr
$\lfloor\ \rfloor$: ganzzahliger Teil
D: Tage bis zum zu berechnenden Datum
T* = T mod 7 → 0: Samstag, 1: Sonntag, 2: Montag, ... ,6: Freitag
Beispiel: 1. Januar 1994, D=1, T=1994+498-19+4+1=2478, T*=2478 mod 7 = 0 → Samstag

- Schreiben Sie ein Programm, das für den ersten Januar eines beliebigen Jahres den Wochentag bestimmt.
- Erweitern Sie das Programm für ein beliebiges Datum.

Hinweis: Schaltjahre sind alle Jahre, die durch 4 teilbar sind, mit Ausnahme der Jahre, die durch 100, aber nicht durch 400 teilbar sind.