Programmierkurs I (Inf., W-Info.), Übungsblatt 7

Claus/Weicker, Wintersemester 03/04 — Abgabetermin: 11.12.2003, 23:59 Uhr

Aufgabe 1: Fibonaccizahlen (leicht) 3+3 Punkte

Die Fibonaccizahlen $F(n)$ ($n \geq 0$) sind eine Folge natürlicher Zahlen, die folgender Gesetzmäßigkeit gehorchen:

$$F(0) = 0 \qquad F(1) = 1 \qquad F(n+2) = F(n) + F(n+1) \text{ für } n \geq 0$$

a) Schreiben Sie ein rekursives Programm `fibonacci1.adb`, welches für ein eingegebenes n den Wert von $F(n)$ bestimmt.

b) Schreiben Sie ein iteratives Programm `fibonacci2.adb`, welches für ein eingegebenes n den Wert von $F(n)$ bestimmt.

```
Geben Sie n ein: 8
Ergebnis: 21
```

Aufgabe 2: Anagramme (mittel) 6 Punkte

Eine Zeichenkette t nennt man ein Anagramm einer Zeichenkette s, wenn man t erhalten kann, indem man die Zeichen von s in einer neuen Reihenfolge anordnet. Zum Beispiel ist `internet` eines von `renitent` und `nikolaustag` eines von `glasauktion`.
Man kann diesen Sachverhalt auch mathematisch präzise ausdrücken: Haben s und t die gleiche Länge n und sind von der Form $s = s_1 \dots s_n$ und $t = t_1 \dots t_n$, so sind s und t genau dann Anagramme voneinander, wenn es eine bijektive Funktion $f : \{1, \dots, n\} \to \{1, \dots, n\}$ (d.h. eine Permutation) gibt, so dass $s_i = t_{f(i)}$ gilt für alle $1 \leq i \leq n$.
Schreiben Sie ein Programm `anagramm.adb`, welches von zwei eingegebenen Zeichenketten (beliebiger Länge) überprüft, ob sie Anagramme voneinander sind.

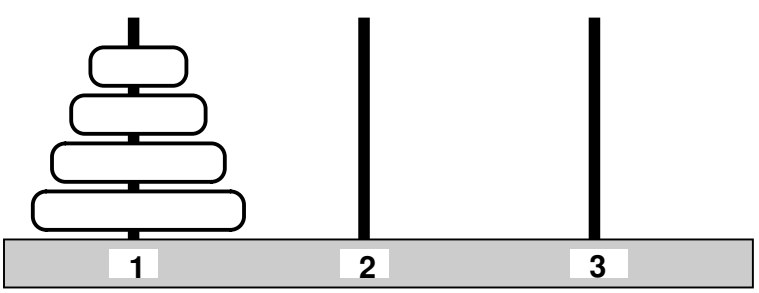
```
Erste Zeichenkette: internet
Zweite Zeichenkette: renitent
Es handelt sich um Anagramme.

Erste Zeichenkette: internet
Zweite Zeichenkette: computer
Es sind keine Anagramme.
```

Hinweis: Betrachten Sie die Häufigkeiten der Buchstaben in den beiden eingegebenen Wörtern.

Aufgabe 3: Türme von Hanoi (mittel) 8 Punkte

Bei dem Spiel „Türme von Hanoi“ ist ein Turm aus Scheiben auf einem Pfosten aufgestapelt. Dieser Turm soll auf einen anderen Pfosten transferiert werden, wobei ein dritter Pfosten als Zwischenablage mitbenutzt werden kann. Die Schwierigkeit besteht nun darin, dass immer nur eine Scheibe bewegt werden darf und niemals eine größere Scheibe auf einer kleineren liegen darf.
Schreiben Sie ein rekursives Programm `hanoi.adb`, welches die Anweisungen zur Verlagerung eines Turms der Höhe n vom Pfosten 1 zum Pfosten 2 ausgibt.



Die Grundidee zur Verlagerung eines Turms der Höhe k von a nach b ($a, b \in \{1, 2, 3\}$, $a \neq b$) ist hierbei, zunächst einen Turm der Höhe $k-1$ von a nach c ($c \in \{1, 2, 3\}$, $c \neq b$, $c \neq a$) zu bewegen, die Scheibe von a nach b zu legen, und den Turm der Höhe $k-1$ von c nach b zu bewegen. Dies wird jeweils rekursiv angewandt, bis Türme der Höhe $k = 0$ erreicht sind.

```
Geben Sie n ein: 3
1. Schritt: Scheibe 1 von 1 nach 2
2. Schritt: Scheibe 2 von 1 nach 3
3. Schritt: Scheibe 1 von 2 nach 3
4. Schritt: Scheibe 3 von 1 nach 2
5. Schritt: Scheibe 1 von 3 nach 1
6. Schritt: Scheibe 2 von 3 nach 2
7. Schritt: Scheibe 1 von 1 nach 2
```

Hinweise

- Zur Abgabe wird das eClaus-System verwendet:
  `http://eclaus.informatik.uni-stuttgart.de`
- Die Abgabe zu jeder Teilaufgabe besteht aus einem kompilierbaren Ada-Quelldatei. In jeder Aufgabe wird ein Dateiname vorgegeben. Verwenden Sie diesen bitte für das Hauptprogramm. Auch sind in der Aufgabe Angaben zu Ein- und Ausgabetexten sowie zur Formatierung der Ausgabe enthalten. Bitte folgen Sie diesen Angaben so genau wie möglich.
- Beachten Sie beim Programmieren bitte die folgenden Hinweise („kleine Programmierrichtlinie“).
  - Halten Sie einzelne Bestandteile überschaubar, z.B. indem Sie nur eine Anweisung pro Zeile schreiben, sowie pro Zeile höchstens 80 Zeichen, höchstens 40 Zeilen pro Prozedur/Funktion, nicht mehr als 800 Zeilen pro Datei und höchstens 5 Parameter bei Prozeduren/Funktionen benutzen.
  - Bezeichner sollen selbsterklärend und maximal 15 Zeichen lang sein. Bezeichner enthalten nur Buchstaben, den Bindestrich oder den Unterstrich.
  - Durch Einrückung um 2 Zeichen soll die logische Gliederung eines Programms verdeutlicht werden. Beispielsweise wird der Anweisungsteil einer IF-Verzweigung eingerückt, während „`end if;`“ nicht mehr eingerückt wird. Auch auf der nächsten Zeile fortgesetzte Zeilen werden um 2 Zeichen eingerückt.
  - Zu Beginn des Programms muss in Kommentaren das Konzept und die Lösungsidee des Programms ausführlich erläutert werden.
  - Auch im Programmtext sind Kommentare einzufügen, um den Code zu erläutern.
- Beachten Sie, dass jede Abgabe individuell vom jeweiligen Studierenden erstellt werden muss. Werden von den Tutoren Plagiate erkannt, d.h. exakte oder leicht modifizierte Kopien, werden für die Aufgabe keine Punkte vergeben. Falls Sie die Lösung der Aufgaben vor der Abgabe in Gruppen besprechen, achten Sie darauf, dort nur das generelle Konzept abzuklären und die Programmierung jedem selbst zu überlassen.
- Bei Fragen wenden Sie sich bitte an Ihren Tutor oder an die Übungsleitung:
  `Karsten.Weicker@fmi.uni-stuttgart.de` oder Tel. 7816-337
- Weitere Veranstaltungshinweise einschließlich der Übungsblätter finden Sie unter:
  `http://www.informatik.uni-stuttgart.de/ifi/fk/lehre/ws03-04/ada95/`