Programmierung	**WS 2000 / 2001**	**Prof. H Lichter** **Th. von der Maßen**

Übung 4

Ausgabe: 15.11.00	Abgabe: 22./23.11.00	Besprechung: 29./30.11.00 in den Gruppen

WICHTIG:
Alle Programme sind sowohl auf Papier in ihrer Übungsgruppe als auch elektronisch mit Hilfe des Online-Systems abzugeben. Die Aufgaben sind allein mit den Mitteln der funktionalen Programmierung zu lösen. Das heißt, es dürfen nur die Modula-3 Elemente verwendet werden, die im Abschnitt „Funktionale Programmierung“ in der Vorlesung vorgestellt wurden. Lösungen, die andere Modula-3 Elemente, wie Schleifen oder Variablen enthalten, werden nicht akzeptiert.

Aufgabe 4.1: Zeichenersetzung (3 Punkte)

Schreiben Sie eine rekursive Modula-3 Funktion „Substitution“, welche einen Text einliest und in diesem Text alle Vokale nach dem folgenden Schema ersetzt:

```
a → e          A → E
e → I          E → I
i → o          I → O
o → u          O → U
u → a          U → A
```

Der substituierte Text soll von der Funktion zurückgegeben und dann am Bildschirm ausgegeben werden. Nach dem Start des Programms soll der Benutzer einen beliebigen Text eingeben können, welcher an die Funktion „Substitution“ übergeben wird. Testen Sie Ihre Funktion mit einem geeigneten Hauptprogramm und mit dem folgenden Text:
„*Drei Chinesen mit dem Kontrabass, sitzen auf der Strasse und erzaehlen sich was. Dann kam die Polizei und was ist denn das? Drei Chinesen mit dem Kontrabass*.“

Aufgabe 4.2: Rekursive Funktion (4 Punkte)

Eine mathematische Funktion *f* sei wie folgt definiert:

$$f(n) = \begin{cases} 1, & \text{falls} \quad n = 1,2,3 \\ f(n-3) - 2 * f(n-2) + 3 * f(n-1), & \text{falls} \quad n \geq 4 \end{cases}$$

a) Geben Sie die Funktionswerte für n=1 bis n=10 an. Schreiben sie ebenfalls den Funktionsausdruck auf. Die Funktionswerte von *f(1)* bis einschließlich *f(n-3)* können direkt eingesetzt und brauchen nicht mehr weiter aufgelöst werden. Im folgenden ist der Funktionsausdruck für n=7 angegeben (die Funktionswerte bis *f(4)* können eingesetzt werden):

$$\begin{aligned} f(7) &= f(4)-2*f(5)+3*f(6) \\ &= 2-2*(f(2)-2*f(3)+3*f(4))+3*(f(3)-2*f(4)+3*f(5)) \\ &= 2-2*(1-2*1+3*2)+3*(1-2*2+3*(f(2)-2*f(3)+3*f(4))) \\ &= 2-2*(1-2*1+3*2)+3*(1-2*2+3*(1-2*1+3*2)) \\ &= 28 \end{aligned}$$

Ein Ergebnis allein wird nicht gewertet.

b) Implementieren Sie eine rekursive Funktion „Berechne“ in Modula-3, welche die oben angegebene Funktion f implementiert. Testen Sie ihre Funktion mit einem geeigneten Hauptprogramm und berechnen Sie die Funktionswerte für n=1 bis n=25 und geben Sie diese an.

Aufgabe 4.3: Kubikwurzelberechnung (7 Punkte)

Die Kubikwurzel einer positiven reellen Zahl kann mit Hilfe des Newton-Verfahrens approximiert werden. Das Verfahren definiert eine Folge $(x_j)_{j\geq 0}$ mit $x_j \xrightarrow[j\to\infty]{} \sqrt[3]{x}$. Die Folge ist folgendermaßen definiert:

$$x_0 = x$$

$$x_j = \frac{x/(x_{j-1})^2 + 2x_{j-1}}{3} \qquad j > 0$$

Schreiben Sie eine Programm „Kubikwurzel“, welches eine Zahl einliest und die Folgenglieder $x_1, x_2, x_3, \ldots, x_k$ berechnet, bis $\left|x_k^3 - x\right|$ kleiner als eine vorgegeben Schranke von 0,0001 ist und x_k als Näherungswert für $\sqrt[3]{x}$ liefert.

a) Schreiben Sie eine Funktion `Abstand(y, x: REAL): REAL`, welche $\left|y^3 - x\right|$ berechnet.
b) Implementieren Sie eine Funktion `GutGenug(abstand, e: REAL): BOOLEAN`, welche testet, ob der Näherungswert innerhalb der Fehlertoleranzschranke liegt.
c) Implementieren Sie eine rekursive Funktion `BerechneKubikwurzel(x, y, fehler: REAL): REAL`, die einen weiteren Näherungswert berechnet, falls der aktuelle nicht gut genug ist. Ist der Wert gut genug, soll dieser zurückgegeben werden.
d) Schreiben sie ein geeignetes Hauptprogramm, um die von Ihnen implementierten Prozeduren und Funktionen zu testen. Lassen Sie Ihr Programm die Kubikwurzel aus folgenden Zahlen berechnen: 27 181 78,961 632,5 64

Aufgabe 4.4: Vermehrung (6 Punkte)

Auf einem unbekannten Planeten leben Wesen, die sich von den uns bekannten Lebewesen erheblich unterscheiden. Sie werden im Winter geboren und leben dann maximal 5 Jahre. Im Sommer verbinden sie sich, soweit möglich, in Dreiergruppen, die sich im Winter wieder aufspalten, wobei ein neues Lebewesen entsteht (d.h. aus drei alten Lebewesen sind drei alte plus ein neues Lebewesen entstanden).
Wir nehmen an, dass im Winter des Jahres 0 nur drei neugeborene Lebewesen vorhanden sind. Wieviele Lebewesen können nach Ablauf von `n` Jahren im Frühling vorhanden sein, wenn man annimmt, dass alle Lebewesen die volle Lebensdauer erreichen und sich maximal vermehren.
Schreiben Sie eine rekursive Modula-3 Funktion `AnzahlLebewesen`, die n als Parameter erhält und die gesuchte Zahl liefert. Hinweis: Dazu ist es sinnvoll, eine eigene Funktion zu realisieren, welche die neugeborenen Lebewesen eines Jahres bestimmt.