Dr. Andreas Borchert
Norbert Heidenbluth
Abteilung Angewandte Informationsverarbeitung

Allgemeine Informatik II für Mathematiker/Wirtschaftsmathematiker (SS 2005)

Abgabetermin: 28. April 2005

Aufgabe 3: Die McCarthy-91-Funktion (3 Punkte)

Nachdem wir in der letzten Woche genug Bäumchen gepflanzt haben, nähern wir uns der Rekursionsthematik nun wieder etwas formaler. Aber auch auf diese Weise kann man mit Rekursionen Spaß haben. Glauben Sie nicht? Nun, dann schauen wir mal auf die folgende Funktion (**McCarthy-91-Funktion**):

Die McCarthy-91-Funktion ist für $n \in \mathbb{N}$ mit $n > 0$ wie folgt (rekursiv) definiert:

$$M(n) = \begin{cases} M(M(n+11)) & \text{für } n \leq 100 \\ n - 10 & \text{für } n > 100 \end{cases}$$

Aufgaben:

1. Versuchen Sie mal, "von Hand" (also auf dem Papier) einige Funktionswerte der McCarthy-Funktion zu ermitteln (sinnvollerweise natürlich für $n \leq 100$)!
2. Implementieren Sie diese Funktion in Oberon und lassen Sie sich die Funktionswerte für $n \in [1, 2, \ldots, 150]$ ausgeben.

Tip: Wenn Sie sich an Teilaufgabe 1 versucht haben, fällt es Ihnen unter Umständen leichter, zu bewerten, ob Ihr Programm aus Teilaufgabe 2 wohl das richtige macht ;-)

Aufgabe 4: Essen a la carte (7 Punkte)

Ein spannendes Gebiet der Mathematik ist die *Kombinatorik*. Die typische Fragestellung in diesem Fach beginnt mit "Wieviele Möglichkeiten gibt es...?". Aufgabe 4 führt uns nun in diesen Bereich der Mathematik.

Jeder kennt ja wahrscheinlich die Situation im Restaurant, daß eine Gruppe von Gästen eine Bestellung aufgegeben hat, aber – wenn das Essen kommt – niemand mehr weiß, wer was bestellt hat.

Nehmen wir an, ein (schwäbisches) Restaurant in der Gegend habe die folgenden Speisen auf der Karte:

- Schupfnudeln
- Maultaschen
- Kässpätzle
- Rostbraten
- Gaisburger Marsch

Angenommen, sechs Gäste an einem Tisch haben 2 mal Kässpätzle, einmal Schupfnudeln und drei mal die Maultaschen bestellt. Die Frage ist nun, auf wieviele Arten der Ober die Speisen auf die sechs Gäste verteilen kann (wenn jeder genau eine Speise bekommt und man davon ausgeht, daß gleiche Speisen nicht voneinander zu unterscheiden sind).

Diese Fragestellung soll diese Übungsaufgabe lösen:

Aufgabe:

Schreiben Sie ein Programm, das als Argument die Bestellung der Gäste enthält und daraufhin alle Möglichkeiten ausgibt, die Bestellung auf die Gäste zu verteilen.

Beispiel: Für die zuvor beschriebene Bestellung wäre der Aufruf des Programms wie folgt:

```
./Restaurant kkmsmm
```

("k" steht hier für Kässpätzle, "m" für Maultaschen, usw.)

Dieser Aufruf sollte dann zu folgender Ausgabe führen:

```
Serviermoeglichkeit # 1:
========================
Gast Nummer 1 bekommt: Schupfnudeln
Gast Nummer 2 bekommt: Maultaschen
Gast Nummer 3 bekommt: Maultaschen
Gast Nummer 4 bekommt: Maultaschen
Gast Nummer 5 bekommt: Kaesspaetzle
Gast Nummer 6 bekommt: Kaesspaetzle

Serviermoeglichkeit # 2:
========================
```

```
Gast Nummer 1 bekommt: Schupfnudeln
Gast Nummer 2 bekommt: Maultaschen
Gast Nummer 3 bekommt: Maultaschen
Gast Nummer 4 bekommt: Kaesspaetzle
Gast Nummer 5 bekommt: Maultaschen
Gast Nummer 6 bekommt: Kaesspaetzle

[... jede Menge andere Moeglichkeiten ...]

Serviermoeglichkeit #60:
========================
Gast Nummer 1 bekommt: Kaesspaetzle
Gast Nummer 2 bekommt: Kaesspaetzle
Gast Nummer 3 bekommt: Maultaschen
Gast Nummer 4 bekommt: Maultaschen
Gast Nummer 5 bekommt: Maultaschen
Gast Nummer 6 bekommt: Schupfnudeln
```

Hinweise:

- Selbstverständlich können / dürfen / müssen / sollten Sie in Ihrem Programm vom Prinzip der *Rekursionen* Gebrauch machen.
- Wie immer sollte Ihr Programm auch diesmal wieder robust gegen Falscheingaben sein, also gegen
  - die Eingabe von nicht existierenden Speisen(kürzeln) und
  - ein zu große oder zu kleine Bestellung.
- Die Anzahl der Gäste können Sie als Konstante vereinbaren, aber Ihr Programm sollte so konzipiert sein, daß es auch nach einer Änderung dieser Konstanten problemlos funktioniert.
- Bevor Sie zu Programmieren beginnen, ist es ratsam, sich erst Gedanken über die benötigte Datenstruktur zu machen.

Viel Erfolg!