

8. Übungsblatt: Programmierpraktikum I (WS 2003/04)

Abgabe: 15. Dezember 2003

Bem. 1: Es sind wieder die Grundregeln ordentlichen Programmierens wie Einrückungen und Kommentare zu beachten. Außerdem sind alle Programme gegen fehlerhafte Benutzereingaben abzusichern!

Java: Samstagslotto

(Aufgabe1.java, 6 Punkte)

Schreibe eine Java-Applikation, die Lottozahlen für das Samstagslotto 6 aus 49 ermittelt. Dabei soll gelten:

- Es werden Zahlen zwischen 1 und 49 verwendet.
- Es werden 6 Lottozahlen und 1 Zusatzzahl gezogen.
- Keine Zahl darf doppelt vorkommen.
- Die 6 “normalen” Lottozahlen sollen aufsteigend geordnet ausgegeben werden, die Zusatzzahl wird separat ausgegeben.

Das Programm soll zur Speicherung der Lottozahlen ein Feld (Array) verwenden.

Eine Musterausgabe des Programmes könnte wie folgt aussehen:

Lottozahlen (6 aus 49):

1 - 3 - 12 - 13 - 27 - 46 - Zusatzzahl 30

Bem.: Um (Pseudo-)Zufallszahlen zu erhalten, muss zunächst ein Pseudozufallsgenerator angelegt werden:

```
java.util.Random rand = new java.util.Random();
```

Der Methodenaufruf `rand.nextInt(49)` gibt dann jedesmal eine Pseudozufallszahl zwischen 0 und 48 zurück.

Java: Sortieren

(Aufgabe2.java, 8 Punkte)

Das Sortierproblem ist eine der grundlegenden Aufgabenstellungen der Informatik. Wir wollen einen einfachen Algorithmus zur Lösung des Sortierproblems betrachten: Das Bubblesort-Verfahren.

Schreibe dazu eine Klasse **Aufgabe2**, die ein Feld **feld** mit 100 Integer-Zahlen und die folgenden Methoden enthält:

- eine Methode **main** zur Steuerung des Programmablaufes,
- eine Methode **erzeugen**, die **feld** mit 100 zufällig ermittelten Integer-Zahlen zwischen 1 und 1000 vollschreibt.
- eine Methode **sortieren**, die **feld** nach dem unten beschriebenen Verfahren sortiert,
- eine Methode **vertauschen**, die zwei Elemente von **feld** vertauscht, und
- eine Methode **ausgeben**, die den Feldinhalt ausgibt, und zwar jeweils 10 Zahlen pro Ausgabezeile.

Das eigentliche Bubblesort-Verfahren funktioniert (in Pseudocode) wie folgt:

```
for i=feldgroesse-2 downto 0
  for j=0 to i
    if(feld[j] > feld[j+1])
      vertausche(feld[j],feld[j+1])
```

Java: Rekursion

(Aufgabe3.java, 6 Punkte)

Der Binomialkoeffizient $\binom{n}{k}$ ist definiert wie folgt:

$$\binom{n}{k} = \frac{n!}{(n-k)! k!}$$

Eine direkte Berechnung über diese Formel ist aber für gewöhnlich zu aufwändig, da das Zwischenergebnis $n!$ schnell zu groß wird. Stattdessen soll für diese Aufgabe die Rekursion

$$\binom{n}{k} = \frac{n-k+1}{k} \cdot \binom{n}{k-1}$$

mit $\binom{n}{0} = 1$ verwendet werden.

Schreibe ein Programm, das zwei positive ganze Zahlen n und k von der Konsole einliest und den Wert $\binom{n}{k}$ zurückgibt. Zur Berechnung soll eine (rekursive) Methode mit der Signatur

```
long binomial(int n, int k)
```

verwendet werden.