

Sommersemester 2018
Übungen zur Vorlesung
Objektorientierte Softwareentwicklung (BA-INF-024)
Aufgabenblatt 2 (20 Punkte)
Zu bearbeiten bis: 05.05.2018

Aufgabe 1 (*Imperative Programmierung in Java: Fakultätsfunktion - 4 Punkte*)

a) Implementieren Sie in Java eine rekursive Funktion, die zu einem Eingabeparameter n den Wert $n! = n \cdot (n-1) \cdot (n-2) \cdot \dots \cdot 1$ zurückgibt (*Fakultätsfunktion*). Ein- und Rückgabeparameter sollen den Typ `long` besitzen.

b) Die für nicht-negative ganze Zahlen n und k definierte Funktion

$$\binom{n}{k} := \begin{cases} \frac{n!}{k!(n-k)!} & \text{für } 0 \leq k \leq n \\ 0 & \text{für } 0 \leq n < k \end{cases}$$

(gesprochen „n über k“) heißt *Binomialkoeffizient*. Implementieren Sie die Binomialkoeffizienten-Funktion. Die Parameter sollen vom Typ `long` sein, ebenso wie der Rückgabewert.

Bemerkung: Es gilt

$$\binom{n}{k} = \frac{n(n-1)\dots(n-k+1)}{1 \cdot 2 \cdot \dots \cdot k}$$

c) Kombinieren Sie nun die beiden Funktionen zur Lösung des *modifizierten Lottoproblems*: Aus n Zahlen lassen sich – bei Berücksichtigung der Anordnung – k Zahlen ohne Zurücklegen auf

$$\binom{n}{k} k!$$

Arten auswählen.

Hinweis: Je nach implementation, können auch schon bei recht kleinen Zahlen fehlerhafte Ergebnisse auftreten. Woran liegt das?

Aufgabe 2 (*Genauigkeit - 2 Punkte*)

Schreiben Sie ein Programm, das die Zahlen 6.5f und 6.5 sowie die Zahlen 6.4f und 6.4 auf Gleichheit testet und das Ergebnis ausgibt. Erklären Sie die Ausgabe Ihres Programms.

Aufgabe 3 (*Implementation eines Algorithmus - 5 Punkte*)

Euklid von Alexandria war ein griechischer Mathematiker. Er lebte um ca. 300 v. Chr. in Alexandria. Er entwickelte einen Algorithmus um den größten gemeinsamen Teiler zweier natürlicher Zahlen zu finden. Dieser ist heutzutage bekannt unter dem Namen Euklidischer Algorithmus und funktioniert wie folgt:

```
Euklid(a, b)
  wenn a = 0 dann
    Ergebnis = b
  sonst
    solange b ≠ 0
      wenn a > b dann
        a = a - b
      sonst
        b = b - a
    ende sonst
```

```

        ende solange
    Ergebnis = a
    ende sonst
ende Euklid

```

- Implementieren Sie den Algorithmus.

Aufgabe 4 (Anwendung objektorientierter Konzepte - 3+3+3 Punkte)

In dieser Aufgabe sollen mehrere Klassen programmiert werden, deren Objekte miteinander agieren. Ziel ist es eine Klasse Schiff, Schifffahrtsamt und Werft zu programmieren.

a) Schreiben Sie eine Klasse Schiff, die folgende Bedingungen erfüllt:

- Ein Schiff hat einen Namen (String), ein Kennzeichen (String) und eine Länge (double).
- Die Sichtbarkeit der Variablen soll private sein.
- Der Konstruktor bekommt einen Namen sowie eine Länge übergeben, die dann den Variablen zugeordnet werden.
- Schreiben Sie eine Methode setKennzeichen(String kennzeichen) die dem Schiff ein, als Parameter übergebenes Kennzeichen, zuordnet.
- Ein Schiff soll eine Nachricht empfangen können. Schreiben Sie eine Methode empfangenNachricht(String msg), die die Nachricht, den Schiffsnamen und das Kennzeichen ausgibt.

b) Beim Schifffahrtsamt sollen die Schiffe registriert werden. Die Klasse soll bis zu 50 Schiffe mittels eines Arrays verwalten.

- Schreiben Sie eine Methode die als Parameter ein Schiff übergeben bekommt und dieses im Array speichert.
- Schreiben Sie eine Methode meldung(String nachricht, int dringlichkeit), die die Nachricht an alle Schiffe weitergibt, wenn die Dringlichkeit 1 ist.

Fügen Sie der Klasse folgende Methode hinzu:

```

public static String getKennzeichen() {
    java.util.Random rnd = new java.util.Random();
    StringBuilder sb = new StringBuilder();
    for(int i = 0; i < 6; i++) {
        sb.append((char)((rnd.nextInt(25)+65)));
    }
    return sb.toString();
}

```

c) Bei der Werft werden Schiffe produziert. Diese kümmert sich auch direkt um die Anmeldung des Schiffes beim Amt und um das Kennzeichen.

- Schreiben Sie eine Methode produziereSchiff, die als Parameter einen Namen, eine Länge sowie das Schifffahrtsamt übergeben bekommt, bei dem das Schiff registriert werden soll.
- Die Methode soll ein neues Objekt vom Typ Schiff erzeugen und es dann beim Amt anmelden. Anschließend soll die Methode sich beim Amt ein Kennzeichen generieren lassen und es dann dem Schiff zuweisen.

Testen Sie ihr Programm mit folgendem Code in der main-Methode:

```

Schifffahrtsamt amt=new Schifffahrtsamt();
Werft w1=new Werft();
w1.produziereSchiff("Gertrut", 12.5, amt);
Schiff sc1=new Schiff("Merlin", 45.9);
amt.registriereSchiff(sc1);
sc1.setKennzeichen(Schifffahrtsamt.getKennzeichen());
amt.meldung("Havarie im Rhein bei Rheinkilometer 591.", 1);
//Ausgabe Empfangen von Gertrut/IHVMXW: Havarie im Rhein bei Rheinkilometer 591.
//Empfangen von Merlin/VPBJQW: Havarie im Rhein bei Rheinkilometer 591.
amt.meldung("Sonnenschein bei Rheinkilometer 650", 0);

```