

Übungsblatt 7: Objektorientierte Programmierung

Ausgabe: 29.11.2013

Abgabe: 06.12.2013

Aufgabe 1: Konsolenparameter und Dateien (4 Punkte)

Schreiben Sie ein Programm, welches überprüfen soll, ob gegebene Dateien im aktuellen Ordner vorhanden sind. Dazu gibt der Nutzer bei dem Aufruf des Programms (mittels Konsolenparameter) an, wie die Datei (ohne Endung) heißt und für welche Endungen er sich interessiert (es wird mindestens eine Endung angegeben, es können aber beliebig viele folgen). Anschließend soll Ihr Programm den jeweils vollständigen Dateinamen mit zugehörigem absoluten Pfad ausgeben und jeweils angeben, ob diese Datei existiert oder nicht.

Der folgende Aufruf

```
java Dateien Dateien class java temp
```

könnte somit die folgende Ausgabe bewirken:

```
/home/schiele/Dokumente/Dateien.class existiert!  
/home/schiele/Dokumente/Dateien.java existiert!  
/home/schiele/Dokumente/Dateien.temp existiert nicht!
```

Aufgabe 2: Caesar-Verschlüsselung (8 Punkte)

Schreiben Sie ein Programm, welches als Konsolenparameter den zu verschlüsselnden Klartext und eine ganze Zahl übergeben bekommt. Die ganze Zahl gibt an, um wie viele Stellen die Buchstaben im Alphabet zyklisch verschoben werden sollen. Der Klartext kann beliebige Zeichen enthalten, allerdings sollen nur die Klein- und Großbuchstaben (ohne Umlaute und ß) zyklisch verschoben werden!

```
java Caesar 'Hallo Welt!' 1  
Ibmmp Xfmu!  
java Caesar 'Hallo Welt!' 53  
Ibmmp Xfmu!  
java Caesar 'Hallo Welt!' -1  
Gzkkn Vdks!  
java Caesar 'Hallo Welt!' -27  
Gzkkn Vdks!
```

1. *Hinweis:* Damit das Programm weiß, dass es sich bei `Hallo Welt!` um den ersten Konsolenparameter handelt, muss diese Zeichenkette in Anführungszeichen gesetzt werden.
2. *Hinweis:* Sie dürfen davon ausgehen, dass die Buchstaben im von Java verwendeten Unicode immer hintereinanderweg stehen. Der z. B. Bereich der Großbuchstaben liegt somit zwischen 'A' und 'Z'.

Aufgabe 3: Operatoren (keine Abgabe)

Was gibt das folgende Programm aus? Begründen Sie!

```
1 public class Ausgabe
2 {
3     public static void main(String[] args)
4     {
5         int i = 12;
6         int j = 5;
7         double x = 7.6;
8
9         i = j++ * --i;
10        System.out.println("i:␣" + i);
11        j = (int) x / 3 * 2;
12        System.out.println("j:␣" + j);
13        x = 4 / 3 * 12.5 % 5;
14        System.out.println("x:␣" + x);
15    }
16 }
```

Kleine Rätselfrage: Wo ist der Fehler? (keine Abgabe)

Das folgende Programm

```
1 public class Fehler
2 {
3     public static void main(String[] args)
4     {
5         if(args.length < 1)
6             return;
7         int zahl = Integer.parseInt(args[0]);
8         int ziffer;
9         while(zahl > 0)
10            ziffer = zahl % 10;
11            zahl = zahl / 10;
12            System.out.println("Ziffer␣" + ziffer);
13    }
14 }
```

liefert die Fehlermeldung:

```
Fehler.java:12: variable ziffer might not have been initialized
    System.out.println("Ziffer " + ziffer);
```

Erklären Sie wodurch die Fehlermeldung zustande kommt und wie der Fehler behoben werden kann!