

Programmieren und Software-Engineering I

Übung 6

Name: _____ Klasse: _____ Datum: _____

Lernziele:

- Erstellen von einfachen Ablaufdiagrammen (mit Verzweigungen und Schleifen)
- Einfache Ablaufdiagramme in Java-Code umwandeln
- Testdokumentation
- Modulo-Operator

Aufgabe 1: Zahlenausgabe (Ex_06_01_Numbers)

Erstelle ein Ablaufdiagramm und ein Java-Programm, welches die ungeraden Zahlen zwischen 1 und 100 ausgibt. Es sollen dabei immer genau 8 Zahlen in einer Zeile stehen. Anschließend wird die Summe der geraden und ungeraden Zahlen ausgegeben.

Beispiel:

1	3	5	7	9	11	13	15
17	19	21	...				
...							
Summe der geraden Zahlen:				2550			
Summe der ungeraden Zahlen:				2500			

Aufgabe 2: Note (Ex_06_02_Mark)

Erstelle ein Ablaufdiagramm und ein Java-Programm, welches die Note für eine eingegebene Punktezahl ausgibt. Beachte dabei auch ungültige Eingaben.

Beispiel:

Eingabe: 86
Ausgabe: 86 Punkte ergeben ein Gut.

Eingabe: 64
Ausgabe: 64 Punkte ergeben ein Befriedigend.

...

Eingabe: 123
Ausgabe: Ungültige Punkteanzahl.

Testdokumentation

Beschreibung wie das Programm getestet wurde.

Testfall	punkte			Ausgabe erwartet	Ausgabe tatsächlich

Aufgabe 3: Werte tauschen (EX_06_03_Change)

Erstelle ein Java-Programm, bei welchem zu Beginn zwei Variablen mit Werten belegt werden. Falls keine aufsteigende Sortierung gegeben ist, sollen die Werte vertauscht werden.

Beispiel:

```
int min = 37;  
int max = 23;
```

...

Ausgabe:

Minimum:	23
Maximum:	37

Programmieren und Software-Engineering I

Übung 6

Aufgabe 4: Primzahl (EX_06_04_Prime)

Erstelle ein Java-Programm welches für eine gegebene Zahl prüft, ob es sich um eine **Primzahl** handelt.

Lösungshinweis: Eine Primzahl ist eine Zahl, die nur durch 1 und sich selbst teilbar ist. Es ist Aufgabe des Programms das zu prüfen.

Beispiel: 37 ist eine Primzahl.
...
12345 ist keine Primzahl

Testdokumentation

Beschreibung wie das Programm getestet wurde.

Testfall	Zahl			Ausgabe erwartet	Ausgabe tatsächlich

Aufgabe 5: Zahlen raten (Ex_06_05_Guess)

Erstelle ein Ablaufdiagramm für folgendes einfache Spiel:

Zu Beginn wird eine Zufallszahl zwischen 1 und 100 festgelegt. Der Benutzer versucht nun, diese Zufallszahl mit maximal 7 Versuchen zu erraten. Nach jedem Tipp bekommt er einen Hinweis, ob die gesuchte Zahl größer oder kleiner ist.

Falls er die Zahl erraten kann, bekommt er eine entsprechende Gewinnmeldung mit der Anzahl der benötigten Versuche. Im anderen Fall bekommt er die Meldung, dass er verloren hat und wie die richtige Zahl gelautet hätte.

Beispiel:

Tipp: 63	Ausgabe: Die gesuchte Zahl ist größer.
Tipp: 80	Ausgabe: Die gesuchte Zahl ist kleiner.
Tipp: 74	Ausgabe: Die gesuchte Zahl ist kleiner.
Tipp: 70	Ausgabe: Die gesuchte Zahl ist größer.
Tipp: 72	Ausgabe: Gratuliere, Du hast die Zahl mit 5 Versuchen erraten!!!
Tipp: 40	Ausgabe: Die gesuchte Zahl ist größer.
Tipp: 60	Ausgabe: Die gesuchte Zahl ist größer.
...	
Tipp: 92	Ausgabe: Leider auch beim 7.Versuch falsch. Die gesuchte Zahl wäre 94 gewesen.

Aufgabe 6: Eingabecheck (Ex_06_06_Inputcheck)

Erstelle ein Java-Programm, welches prüft, ob eine Zahl im vorgegebenen Bereich liegt. Andernfalls wird eine entsprechende Meldung ausgegeben.

Beispiel:

```
lower = 15;
upper = 47;
number = 42;
```

Ausgabe: Die Zahl 42 liegt im vorgegebenen Bereich zwischen 15 und 47.

```
lower = 30;
upper = 60;
number = 22;
```

Ausgabe: Die Zahl 22 ist zu klein für den vorgegebenen Bereich zwischen 30 und 60.

Programmieren und Software-Engineering I

Übung 6

Testdokumentation

Beschreibung wie das Programm getestet wurde.

Testfall	lower	upper	number	Ausgabe erwartet	Ausgabe tatsächlich

Aufgabe 7: Logische Ausdrücke

Überprüfe ob es sich bei der Zahl `myNum` um ...

... eine ungerade Zahl handelt.	
... eine dreistellige Zahl handelt.	
... eine negative, gerade Zahl handelt.	
... eine Zahl im Intervall [-50, 50] handelt.	
... eine Zahl im Intervall]0, 3967] handelt.	
... eine Zahl im Intervall]min, max[handelt.	
... eine 4stellige, ungerade Zahl handelt.	
... eine Zahl mit genau 2 Stellen handelt.	
... die gleiche Zahl wie <code>anotherNum</code> handelt.	

Aufgabe 8: Tutorial

Schau Dir die Kapitel 2, 3, 5, 6, 7, 9, 10 und 12 des Java-Tutorials an:

https://www.youtube.com/watch?v=rv_lmAy6-HE (Kapitel 2)

Aufgabe 9: Sonstiges

- Wiederholung vom Theorie-Unterricht (Mitschrift + Kopien)
- Genaues Ansehen der bisherigen Musterlösungen

WICHTIG

Alle Programme müssen einen Programmkopf (=Beschreibung) enthalten.

z.B.:

```

/*****
*      Name:      Max Mustermann
*      Hü:        4
*      Bsp:        2
*      Datum:      20.10.2015
*      Dateiname:  HUE_04_02_Dreieck.java
*      Beschreibung: Es wird für gegebene Seitenlängen a, b und c geprüft, ob es ein gleichseitiges, ein
                    gleichschenkeliges, ein rechtwinkeliges, ein sonstiges gültiges oder ein ungültiges
                    Dreieck ist.
*****/

public class HUE_04_02_Dreieck { ...

```