Prof. Dr. Stefan Brass
M.Sc. Mario Wenzel

Martin-Luther-Universität Halle-Wittenberg
Institut für Informatik
Wintersemester 2018/19

Objektorientierte Programmierung: Hausaufgabenblatt 2

Abgabe: 05.11.2018, 11:00

Das Ziel dieser Übungsblattes ist es, dass Sie die Ausführung eines kleinen Programms auf Papier simulieren können, sowie in der Lage sind, Ausdrücke und Anweisungen auf syntaktischer Ebene zu unterscheiden.

Hausaufgabe 1a: (8 Punkte)

Bitte beachten Sie: die Einsendung der Aufgabe erfolgt über die Übungsplattform.

Das folgende Java-Programm (ohne `=>`) implementiert einen bekannten Algorithmus. Vollziehen Sie den Programmablauf nach und **geben Sie den Zustand der Variablen** `x` und `y` an den markierten Stellen an, also *vor dem ersten* Schleifendurchlauf und *am Ende jedes* Schleifendurchlaufs. Geben Sie auch die Ausgabe des Programms an.

Wiederholen Sie dies für die Belegungen `x=11,y=7` und `x=4,y=128` (also geänderten Zuweisungen in den Zeilen 3 und 4). Zur Kontrolle sind einige Belegungen bereits vorgegeben.

Beantworten Sie zusätzlich folgende Fragen: Was stellen Sie über die jeweils letzte Belegung vor der Ausgabe fest? Geben Sie eine Belegung für `x` und `y` an, bei der die Belegungstabelle nur eine Zeile (zzgl. Ausgabe) hat. Welcher Algorithmus wird hier implementiert?

```
1  class Ecld {
2    public static void main(String[] args) {
3      int x = 54;
4      int y = 24;
5
6  =>  while (y > 0) {
7        int swap = x % y;
8        x = y;
9        y = swap;
10 =>  }
11     System.out.println(x);
12   }
13 }
```


x	y
54	24
	0
Ausgabe:	——

x	y
11	
	3
Ausgabe: 1	——

x	y
128	
Ausgabe:	——

Hausaufgabe 1b: (4 Punkte)

Bitte beachten Sie: die Einsendung der Aufgabe erfolgt über die Übungsplattform.

Ordnen Sie die folgenden Zeichenketten der Gruppe “Wertausdrücke” oder “Anweisungen” zu (Häkchen). Wenn die Zeichenkette keiner von beiden Gruppen zuzuordnen ist, begründen Sie dies kurz. Sie können davon ausgehen, dass alle unbekannten Bezeichner weiter oben im Programm definiert worden sind.

	Zeichenkette	Anweisung	Ausdruck	Beides nicht (Begründung)
1	`x = 7;`			
2	`6          *          7`			
3	`pluto`			
4	`7.0 + 3x`			
5	`"pluto"`			
6	`System.out.println(x+4;)`			
7	`x == 7`			
8	`{{{{{int z;}}}}}`			

Hausaufgabe 2: (4 Punkte)

Bitte beachten Sie: die Einsendung der Aufgabe erfolgt nur im YAPEX.
Open exercises via code → Freigabecode: `5connc4ce61h-6387` (Progressbar_Lueckentext)
Implementieren Sie ein Programm, das eine Ganzzahl zwischen (einschließlich) 0 und 100 einliest (auf Eingaben außerhalb dieses Bereich brauchen Sie nicht prüfen - die Eingabe repräsentiert eine Prozentzahl) und anschließend einen Fortschrittsbalken ausgibt, der 10 Zeichen breit ist. Dabei soll für jede ganze 10% der Eingabe ein `|` ausgegeben werden, und die restlichen Zeichen mit `-` aufgefüllt werden.

Hier sehen Sie einige Beispiel-Ein- und Ausgaben:

<table>
<tr><th>Eingabe</th><th>Ausgabe</th></tr>
<tr><td>0</td><td><code>----------</code></td></tr>
<tr><td>25</td><td><code>||--------</code></td></tr>
<tr><td>50</td><td><code>|||||-----</code></td></tr>
<tr><td>99</td><td><code>|||||||||-</code></td></tr>
<tr><td>100</td><td><code>||||||||||</code></td></tr>
</table>

Das folgende Programm ist eine gültige Lösung, bei der alle Variablenbezeichner und Zeichenketten durch `#` und alle Schlüsselwörter durch `%` ersetzt worden sind. Sie können diese Vorlage nutzen, müssen dies jedoch nicht.

```
% java.util.Scanner;

% Progress {
        % % % main(String[] #) {
                Scanner # = % Scanner(System.in);

                % progress = #.nextInt() / 10;
                % # = 10 - #;

                %(# > 0) {
                        System.out.print(#);
                        # = # - 1;
                }
                %(# > 0) {
                        System.out.print(#);
                        # = # - 1;
                }
        }
}
```