



Hochschule Augsburg
Studiengang Wirtschaftsingenieur
Sommersemester 2016

Name:

Matrikelnr.:

Code:

Datum, Uhrzeit	04.06.2016, 9:00 Uhr
Semester	W-Ing. 2
Prüfer	Prof. Dr. Michael Kipp
Dauer	90 Minuten
Hilfsmittel	Laborrechner, Processing-Umgebung, Processing-Referenz (online)

Klausur zur Vorlesung **Grundlagen der Informatik und Programmieren**

Infos

- Diese Prüfung hat **5 Aufgaben auf 5 Seiten** - bitte prüfen!
- Tragen Sie oben **Ihren Namen und Ihre Matrikelnummer** ein
- Alle Aufgaben sind bezogen auf die Programmiersprache **Processing**
- Wichtig ist zunächst, dass jedes Programm lauffähig ist. Wenn Sie Teillösungen haben, die fehlerhaft sind, kommentieren Sie den fehlerhaften Teil aus und schreiben Sie z.B. "Bin hier nicht weitergekommen"
- Jeder Versuch, auf **unerlaubte Hilfsmittel (z.B. externe Webseiten)** oder auf **vorhandenen Code** auf Ihrem Laborrechner zurückzugreifen, wird als **Täuschungsversuch** gewertet

Bevor Sie beginnen...

1. Legen Sie ein Verzeichnis auf dem Desktop nach folgendem Schema an:

NachnameVorname_<handschriftlicher Code von oben>

Beispiel: HackerRudy_XY34

2. Innerhalb dieses Verzeichnis speichern Sie die Programme für die folgenden Aufgaben:

NachnameVorname_<Aufgabennummer>, also z.B.

HackerRudy_1

HackerRudy_2

HackerRudy_3

etc.

Aufgabe 1

Schreiben Sie zwei Funktionen: **betragSumme()** und **randomN()**.

betragSumme() bekommt einen Array von int-Werten und gibt die Summe aller *Beträge* zurück, d.h. negative Werte werden positiv gerechnet.

Zum Beispiel beim Array { 1, -5, 10, -3 } wird gerechnet 1 + 5 + 10 + 3, ergibt also 19. Sie dürfen **NICHT** auf die vorhandene Funktion `abs()` zurückgreifen!

randomN() bekommt eine ganze Zahl N übergeben und einen Array von float-Zahlen der Länge N zurückgeben, der mit Zufallszahlen zwischen 0 und 10 gefüllt ist. Zum Beispiel: bei `randomN(3)` bekomme ich einen float-Array mit drei Zufallszahlen zurück.

Testen Sie Ihre Funktionen mit folgendem Code. **Verwenden Sie keine globalen Variablen!**

```
void setup() {  
  int[] a = {1, -5, 10, -3};  
  int[] b = {-5, -3, -5};  
  println(betragSumme(a));  
  println(betragSumme(b));  
  println();  
  
  println(randomN(2));  
  println();  
  println(randomN(3));  
}
```

Es sollte ausgegeben werden:

19

13

(die folgenden Zahlen sind natürlich nur Beispiele, da Zufallszahlen)

[0] 0.7408035

[1] 4.857767

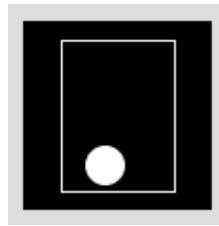
[0] 4.02894

[1] 0.35072148

[2] 9.653801

Aufgabe 2

Schreiben Sie ein Programm mit einem 100x100 Fenster mit einem Ball (Durchmesser 20) in einem Rechteck (Position 20, 10 – Breite 60, Höhe 80). **Tipp:** Verwenden Sie `noFill()` für das Rechteck.



Animieren Sie den Ball so, dass er von den Wänden des *gezeichneten* Rechtecks abprallt. Startpunkt ist 50, 50. Geschwindigkeit zufällig im Bereich -3 bis 3 in beide Richtungen. Verwenden Sie **float** für Position und Geschwindigkeit! Der Ball sollte das Rechteck nie verlassen. Achten Sie darauf, dass der Ball *korrekt* abprallt, d.h. er berührt die Seiten, aber übertritt sie nicht.

Aufgabe 3

Schreiben Sie die Funktion `splitBoxSpalte()` mit den Parametern: *breite, hoehe, spalte*. Ihre Funktion sollte einen (Text-)Kasten der entsprechenden Höhe und Breite zeichnen, bestehend aus Punkten und Hash-Symbolen (#). Der Parameter **spalte** gibt an, bis zu welcher Spalte die Box mit #-Symbolen gefüllt sein soll – der Rest wird mit Punkten gefüllt. **WICHTIG:** Die Zählung von Spalte soll bei 1 beginnen (nicht 0). *Ihre Funktion soll natürlich mit beliebigen Werten funktionieren!*

Testen Sie Ihre Funktion mit folgendem Code:

```
void setup() {  
  splitBoxSpalte(10, 5, 6);  
}
```

Sie sollten folgenden Output sehen:



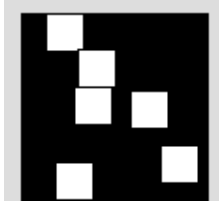
Hinweise: Achten Sie darauf, dass Sie Breite und Höhe nicht verwechseln. Verwenden Sie keine globalen Variablen. **Beginnen Sie, indem Sie ein Rechteck nur aus Punkten ausgeben und verfeinern Sie dann Ihr Programm.**

Aufgabe 4

Schreiben Sie eine Klasse **Floater**, die ein Quadrat (20x20) zeichnet und bewegt. Die Klasse hat 4 Eigenschaften:

x, y, Geschwindigkeit (alle float) und eine boolesche Variable **horizontal**.

Es gibt **genau einen Konstruktor** mit einem booleschen Parameter für die Eigenschaft **horizontal** (die anderen drei Eigenschaften werden zufällig gesetzt).



Schreiben Sie zwei **Methoden** fürs Zeichnen (render) und Updaten (update).

Je nach Wert von Eigenschaft **horizontal** soll das Quadrat horizontal (links-rechts) oder vertikal (hoch-runter) animiert werden und dabei von den Wänden abprallen (Quadrat sollte nicht in die Wand hineinschneiden!).

Im **Hauptprogramm** erzeugen Sie ein Fenster der Größe 100x100 mit schwarzem Hintergrund und sechs Floater-Objekten, die Sie **in einem Array** speichern. Bei der Hälfte der Objekte soll horizontal *true* sein, bei der andere Hälfte *false*.

Bei **Mausklick** soll **bei allen Objekten die Richtung umgekehrt** werden, also von horizontal auf vertikal oder von vertikal auf horizontal (Verwenden Sie dazu die boolesche Eigenschaft **horizontal**).

Ihr Programm sollte auch problemlos laufen, wenn Sie die Anzahl der Objekte auf 4 oder 12 stellen.

Aufgabe 5

Gegeben seien zwei gleich lange Arrays:

```
int[] a = {1, 2, 3, 4};  
int[] b = {10, 20, 30, 40};
```

Erzeugen Sie einen neuen Array **c**, der abwechselnd alle Wert von **a** und **b** enthält, also den ersten Wert von **a**, dann den ersten Wert von **b**, dann den zweiten Wert von **a** usw.

Ihre Ausgabe im Beispiel sollte so aussehen:

```
[0] 1  
[1] 10  
[2] 2  
[3] 20  
[4] 3  
[5] 30  
[6] 4  
[7] 40
```

*Ihre Funktion sollte für beliebige Werte (und beliebig viele Werte) für **a** und **b** funktionieren, solange die Länge von **a** und **b** gleich ist.*

Anweisungen zur Abgabe (ZIPen und Hochladen)

Stellen Sie von Ihrem Verzeichnis auf dem Desktop eine ZIP-Datei her. Das machen Sie mit einem Rechtsklick auf das Verzeichnis: dann auf **Senden an** und dort auf **ZIP-komprimierter Ordner**. Anschließend erscheint Ihr Verzeichnis als ZIP-Datei (Icon mit Reißverschluss).

Laden Sie diese ZIP-Datei in **Moodle** unter **ABGABE** hoch.

Stellen Sie sicher, dass Sie **Änderungen sichern** geklickt haben.

Denken Sie daran, diesen Prüfungsbogen abzugeben!!!