

Übungsblatt 3

Abgabedatum: 07.11.2021

Die Abgabe Ihrer Lösungen erfolgt vor Ablauf der Abgabefrist digital über die Moodle-Plattform. Erstellen Sie dazu ein PDF-Dokument, das die Lösungen Ihrer schriftlichen Aufgaben enthält. Laden Sie dieses PDF-Dokument und den erarbeiteten Java-Code (.java-Dateien) mit den in den Aufgaben vorgegebenen Namen bei Moodle hoch. Bitte laden Sie die Dateien einzeln hoch, Dateiarhive (z.B. .zip-Dateien) werden nicht akzeptiert.

Sie können maximal **(6 Punkte)** mit diesem Übungsblatt erreichen.

Aufgabe 1 (Vergleiche)

1 Punkt

1. Schreiben Sie eine Anwendung `Namen.java`, welche zwei Namen einliest und den längeren Namen gefolgt von den Worten „hat mehr Buchstaben als“ gefolgt vom kürzeren Namen ausgibt. Sind beide Namen gleich lang, so soll der erste Name gefolgt von „hat gleich viele Buchstaben wie“ gefolgt vom zweiten Namen ausgegeben werden. Auf einem String kann die Methode `length()` ausgeführt werden, die die Anzahl der Zeichen des Strings wiedergibt.
2. Schreiben Sie eine Anwendung `Teilbar.java`, welche drei Zahlen a, b, c einliest und ausgibt, ob die erste Zahl a durch die zweite Zahl b oder die dritte Zahl c oder durch beide Zahlen b und c teilbar ist.
3. Schreiben Sie ein Java-Programm (`Karten.java`), dass eine zufällige Mau-Mau Karte textuell in der Konsole ausgibt. Es genügt dabei die Farbe (Karo, Herz, Pik, Kreuz) als Text und den Wert (7, 8, 9, 10, B, D, K, A) in einem geeignetem Format auszugeben. Verwenden Sie bei der Implementierung die `switch`-Anweisung. Die Ausgaben könnten zum Beispiel wie folgt aussehen:
 - Herz K
 - Kreuz 7
 - Pik 10

Aufgabe 2 (Schleifen)

2 Punkte

1. Schreiben Sie ein Programm (`Palindrom.java`), das einen Text einliest und ausgibt ob dieser ein Palindrom ist. Ein Palindrom ist eine Folge, die identisch mit ihrer Umkehrung ist.
2. Schreiben Sie ein Java-Programm (`Vokale.java`), welches die Anzahl der Vokale, der Konsonanten und der übrigen Zeichen in einem eingegebenen Wort ermittelt und ausgibt. Beachten Sie, dass eine Zeichenkette `str` mit `str.toLowerCase()` in Kleinbuchstaben konvertiert und auf ihr i -tes Zeichen mit `str.charAt(i)` zugegriffen werden kann.

3. Schreiben Sie eine Anwendung `Raute.java`, welche ein Rautenmuster aus `#` ausgibt. Der Nutzer soll zuvor die Breite b der Raute eingeben können. Bevor eine Raute ausgegeben wird, soll getestet werden ob b eine gerade Zahl ist, ist dass der Fall, so soll das Programm mit der Textausgabe „Die Raute kann nur für ungerade Eingaben gezeichnet werden.“ ausgegeben werden. Beachten Sie auch die folgenden Beispiel für die Eingabe 5 und 7.

```
5      7
  #      #
 ###    ###
#####  #####
  ###  #####
   #   #####
      ###
        #
```

4. Schreiben Sie ein Java-Programm (`Hexadezimal.java`), welches durch Verwendung der ganzzahligen Division `/` sowie der Modulo-Operation `%` eine einzugebende positive ganze Zahl in ihre hexadezimale Darstellung umwandelt und ausgibt.

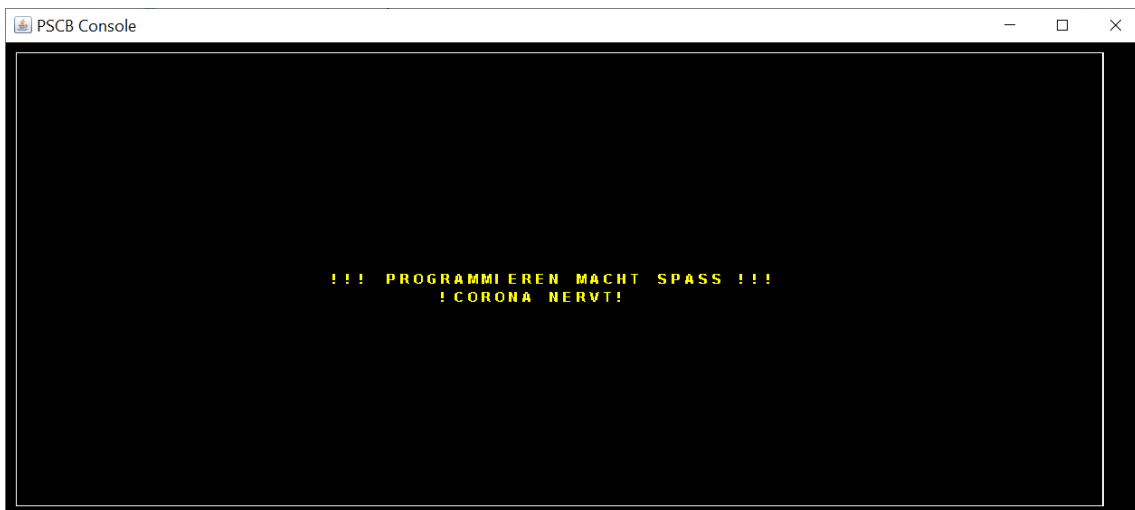
Aufgabe 3 (Laufband)

2 Punkte

1. Schreiben Sie eine Anwendung `Laufband.java`, welche die aus dem zweiten Aufgabenblatt bekannte `Console`-Klasse nutzt um ein animiertes Laufband mit dem Text „!!! PROGRAMMIEREN MACHT SPASS !!!“ und mittig darunter den Text „!CORONA NERVT!“ darzustellen.

Der Laufbandtext soll in der zwölften und dreizehnten Zeile dargestellt werden und sich abwechselnd zwischen dem rechten und linken Fensterrand hin und her bewegen. Wenn der Text den Rand berührt, soll sich die Textfarbe verändern. Orientieren Sie Sich an der Laufband-Beispielanimation aus der Vorgabe.

Sie können die Funktion `Console.wait()` nutzen, um zwischen den Animationsschritten geeignet lange Pausen einzufügen und so die Animationsgeschwindigkeit anzupassen.



2. Schreiben Sie eine zweite Anwendung `Spiegel.java`, welche ebenfalls die aus dem zweiten Aufgabenblatt bekannte `Console`-Klasse nutzt um ein „L“ am linken und ein „R“ rechten Rand starten zu lassen und diese solange aufeinanderzuwandern lässt, bis diese benachbart sind. Dann sollen das „L“ und das „R“ wieder auseinander wandern. Dies soll unendlich oft wiederholt werden.

Aufgabe 4 (Tarife)

1 Punkt

1. Schreiben sie eine Anwendung `Nachricht.java`, welche mit der Konsolenausgabe „Nachricht schreiben:“ beginnt, danach über die Konsole einen Text einliest und anschließend „XX Nachricht/en wurden versendet.“ ausgibt. Dabei wird XX mit der benötigten Anzahl von Teilnachrichten ersetzt. Eine Teilnachricht umfasst max. 15 Zeichen.

(Hinweis: Die Methode `.length()` gibt die Anzahl an Zeichen eines Strings wieder.)

2. Erweitern sie die Anwendung wie folgt:

- Vor dem schreiben der Nachricht ist einer von 3 Anbietern (incl. Preisinformationen) über die Konsole zu wählen. Die Auswahl des Anbieters erfolgt über die Eingabe einer Ziffer von 1-3. Die 3 zu wählenden Anbieter haben dabei unterschiedliche Preise für eine Teilnachricht, Sowie unterschiedliche einmalige Grundgebühren für das Versenden:

Anbieter 1: 12ct Grundgebühr, 6ct/Nachricht

Anbieter 2: 0ct Grundgebühr, 9ct/Nachricht

Anbieter 3: 55ct Grundgebühr, 0ct/Nachricht

- Nach dem schreiben der Nachricht wird neben der Anzahl an Teilnachrichten auch der Preis für die versendete Nachricht angezeigt.
3. Erweitern sie die Anwendung wie folgt: Bei der Anbieterauswahl erscheint ein neuer Auswahlpunkt „4. Komfortfunktion“. Diese Auswahl findet anhand der Anzahl zu versendender Nachrichten von selbst heraus, welcher Anbieter für den Anwender am kostengünstigsten ist und ergänzt die schlussendliche Ausgabe um die Information, welcher der 3 Anbieter verwendet wurde.