Übungsaufgaben Java 1

BlueJ Szenario: https://codeberg.org/qg-info-unterricht/bluej-uebungen01/archive/master.zip [1)]

Bearbeite die folgenden "Fingerübungen".

- Teste deine Methoden mit mehreren Beispieleingaben (wo möglich):
  - Überlege dir bei den Beispielen vor allem, solche, an denen du "Grenzfälle" testen kannst.
  - Überlege erst, welche Ausgabe du für eine bestimmte Eingabe erwartest und teste dann.
  - Halte deine Testfälle im Kommentarbereich deiner Methode fest.
- Mache dir bewusst, was man bei der Lösung beachten muss, vor allem wenn es nicht gleich klappt. Halte diese Erkenntnisse schriftlich fest.

A01 - Modulo selbst gemacht

Eine praktische Rechenoperation ist "Modulo" - sie gibt den Rest bei der Divisopn zweier ganzer Zahlen zurück:

```
7 mod 5 = 2 //  7 = 1 * 5 + 2 (5 geht einmal ganz in 7, dann bleibt ein Rest
von 2
167 mod 10 = 7  // 167 = 16 * 10 + 7 ("167/10 = 16 Rest 7")
```

Java hat dafür den Modulo Operator `%`. Erstelle eine Methode die den Rest der ganzzahligen Division zweier Integer Zahlen ausgibt. Benutze **nicht** den Modulo-Operator `%` arbeiten, sondern implementiere die Funktionalität selbst.

Tipp

Wenn zwei ganze Zahlen (int) in Java geteilt werden, dann wird auch eine ganze Zahl (int) als Ergebnis dargestellt. `10/3` ergibt also nicht 3.333333 als Ergebnis sondern 3, die Nachkommastellen werden also "abgeschnitten". Diesen Umstand kannst du dir zu Nutze machen.

A02 - Dreieckstausch

Manchmal möchte man die Werte zweier Variablen **vertauschen**, beispielsweise beim Sortieren.

Erstelle eine Methode, mit der zwei Zahlen vertauscht werden können. Die Parameter der Methode sollen die beiden Zahlen sein, die Ausgabe der Methode soll z.B. sein

```
a=7  b=16
a=16 b=7
```

Tipp

- Zunächst wird einer Hilfsvariablen der Wert von a zugewiesen.

- Danach kann a mit dem Wert von b überschrieben werden, da die Hilfsvariable ihren vorherigen Wert (von a) speichert.
- Danach bekommt b den Wert der Hilfsvariablen zugewiesen.

Der Tausch findet also "im Dreieck" über die Hilfsvariable statt.

Zusatz: Findest du eine Möglichkeit, ohne Hilfsvariable auszukommmen? Da muss man ein wenig rechnen…

A03 - Pyramidenvolumen

Schreibe eine Methode, um das Volumen einer Pyramide mit qudratischem Boden zu berechnen (V = 1/3 * Grundfläche * Höhe). Welche Parameter benötigt deine Methode?

Tipp

Wenn immer `0.0` rauskommt - denk an die Modulo-Aufgabe: Was passiert bei der Division zweier ganzer Zahlen mit den Nachkommastellen?

Recherche mit Notiz: Welche primitiven Variablentypen kennt Java? Kannst du dir einen Grund vorstellen, warum man bei der Deklaration einer Variablen ihren Typ angeben muss?

A04 - Alterstest

Entscheidungen: "Wenn-dann-Entscheidungen" sind für die strukturierte Programmierung unerlässlich.

Erstelle eine Methode, welche überprüft, ob die Integervariable `alter` kleiner ist als 7. In diesem Fall wird "Geschäftsunfähig" ausgegeben. Andernfalls wird geprüft, ob das Alter kleiner ist als 18. In diesem Fall wird "Geschäftsfähigkeit beschränkt" ausgegeben. Andernfalls wird "unbeschränkte Geschäftsfähigkeit" ausgegeben.

Tipp

Verwende die Funktion `Math.random()`, die eine Zufallszahl zwischen 0 und 0.999… erzeugt. Durch Multiplikation mit 49 und Addition der Zahl 1 gecastet in einen Integerwert erhältst du eine zufällige Lottozahl.

A05 - Gerade oder ungerade?

Erstelle eine Methode, die prüft, ob eine eingegebene ganze Zahl gerade oder ungerade ist. Die Methode erzeugt unten stehende Bildschirmausgabe.

```
Die Zahl 51 ist ungerade
Die Zahl 200199212 ist gerade
```

Tipp

Verwende den Modulo Operator %

A06 - Schulnoten

Sehr viele Entscheidungen führen mitunter zu unübersichtlich verschachtelten "Wenn-Dann" Anweisungen. Viele Programmiersprachen haben für solche Fälle eine "case" oder "switch" Statement, das Entscheidungen mit vielen Möglichkeiten übersichtlicher gestalten kann.

Erstelle eine Methode, die eine Kommazahl als Parameter erhält und dann entsprechende der Rundungsregeln über die verbale Note entscheidet.

Z.B. ergibt die Note 2,5 die verbale Note: befriedigend. Ungültige Noten, also Noten außerhalb des Notenspektrums zwischen 1 und 6 haben die Bildschirmausgabe: "keine gültige Note" zur Folge.

Hinweis: Löse die Aufgabe zuerst mit verschachtelten if-Anweisungen, recherchiere dann nach dem *switch-case*-Statement und löse die Aufgabe damit.

A07 - Wertetabelle

Schleifen sind wichtige Sprachemlemente zur Wiederholung von Anweisungen. Wenn man zuvor schon weiß, wie oft eine Anweisung ausgeführt werden wird, kann eine Zählschleife (for) zum Einsatz kommen.

Erstelle eine Wertetabelle für die Funktion f(x)=x^2-6x-4 für das Intervall -2 bis 2 mit der Schrittweite 0.1 und gib sie auf der Konsole aus.

Was fällt dir dabei auf?

A08 - Schaltjahr

Überprüfe (ohne die Verwendung von Bilbliotheksfunktionen) ob ein gegebenes Jahr ein Schaltjahr ist.

Tipp

Schaltjahre können anhand der folgenden Regeln bestimmt werden:

- Regel 1: Alle 4 Jahre ist ein Schaltjahr (das bedeutet: es gibt einen 29. Februar)
- Regel 2: Alle 100 Jahre tritt Regel 1 außer Kraft.
- Regel 3: Regel 2 tritt alle 400 Jahre außer Kraft

A09 - Stellenzähler

Wenn man im Voraus nicht weiß, wie oft eine Anweisung wiederholt werden muss, bieten sich eine while-Schleife an.

Schreibe eine Methode, die die Stellen einer eingegebenen Ganzzahl größer als Null zählt:

```
Deine Eingabe: 21355746
Die Zahl hat 8 Stellen.
```

Tipp

Bei der Division zweier Ganzzahlen gibt Java nur den ganzzahligen Anteil zurück, das kannst du dir hier zu Nutze machen:

```
3547/10=354
354/10=35
```

A10 - Quersumme

Berechne die Quersumme einer eingegebenen Ganzzahl.

Tipp 1

Hier kannst du den Stellenzähler einfach erweitern. Wie kannst du die letzte Ziffer der Zahl herausfinden?

Tipp 2

```
12876 % 10 = 6
5 % 10 = 5
7625415261 & 10 = 1
```

A11 - Dualzahl in Dezimalzahl umrechnen

Erstelle eine Methode, die als Parameter eine Dualzahl entgegen nimmt und die zugehörige Dezimalzahl berechnet. Z.B.

```
Die Dualzahl 110101101 hat den Dezimalwert: 429
```

Tipp

Du kannst wieder den Code der vorigen Aufgaben "recyceln" und entsprechend erweitern.

Welche Probleme erkennst du bei dieser Aufgabenstellung - was sollte man sinnvollerweise implementieren, damit die Methode wirklich sinnvoll ist?

A12 - Zahlendreher

Schreibe eine Methode, die die Ziffernreihenfolge einer Ganzzahl umdreht. Z.B:

```
Du hast die Zahl 153536758 eingegeben!
Die umgedrehte Zahl lautet: 857635351
```

Tipp 1

Wenn du Aufgabe 09-11 verstanden hast und geschickt kombinierst, muss du nicht sehr viel neues erfinden.

Tipp 2

Wie kannst du herausfinden, welchen Stellenwert die Ziffer ganz rechts bei der Zahl 76321 hat? Wie oft muss die umgedrehte Zahl diesen Stellenwert haben?

Pi nach Leibnitz

Erstelle eine Methode, die mit der Formel: PI=(1-1/3+1/5-1/7+1/9-1/11+1/13-….+1/n)*4 einen Näherungswert für die Zahl Pi berechnet. Als Parameter soll die Anzahl der Iterationen (n) übergeben werden.

X-Block

Erstelle eine Methode, die mit zwei verschachtelten Schleifen folgende Bildschirmausgabe erzeugt:

```
x x x x x x x x x x
x x x x x x x x x x
x x x x x x x x x x
x x x x x x x x x x
x x x x x x x x x x
x x x x x x x x x x
x x x x x x x x x x
x x x x x x x x x x
x x x x x x x x x x
x x x x x x x x x x
```

Dreieck

Erstelle eine Methode, die mit zwei verschachtelten Schleifen folgende Bildschirmausgabe erzeugt:

```
X
XX
XXX
XXXX
XXXXX
XXXXXX
```

Lottozahlen

Erstelle eine Methode, die 6 unterschiedliche Lottozahlen zwischen 1 und 49 zufällig erzeugt und gibt diese auf dem Bildschirm ausgibt, z.B.:

```
22 16 45 29 3 37
```

Verwende zur Speicherung der Lottozahlen ein Array, um dir zu merken, welche Zahlen bereits gezogen wurden

---

Lösungsvorschläge ->

[1)]

git clone https://codeberg.org/qg-info-unterricht/bluej-uebungen01

From:
https://info-bw.de/ -

Permanent link:
https://info-bw.de/faecher:informatik:oberstufe:java:algorithmen:uebungen01:start?rev=1631538460

Last update: **13.09.2021 13:07**