

A2

Teil A: Divisionsmuster ausgeben

Gegeben eine Zahl n und zwei Zählvariablen i, j , die von 1 bis n hochzählen. Schreiben Sie eine Methode, die `*` ausgibt, wenn i durch j oder j durch i ohne Rest teilbar ist, ansonsten Leerzeichen.

Am Ende jeder Zeile soll die Zeilennummer ausgegeben werden.

Ausgaben ohne Zeilenumbruch erzeugen Sie in Ruby mit der Methode `print`

Für $n = 16$ sollte das Ergebnis wie folgt aussehen

```
* * * * * * * * * * * * * * 1
* * * * * * * * * * * * * 2
* * * * * * * * * * * * * 3
* * * * * * * * * * * * * 4
* * * * * * * * * * * * * 5
* * * * * * * * * * * * * 6
* * * * * * * * * * * * * 7
* * * * * * * * * * * * * 8
* * * * * * * * * * * * * 9
* * * * * * * * * * * * * 10
* * * * * * * * * * * * * 11
* * * * * * * * * * * * * 12
* * * * * * * * * * * * * 13
* * * * * * * * * * * * * 14
* * * * * * * * * * * * * 15
* * * * * * * * * * * * * 16
```

Teil B: Quadratische Gleichung berechnen

Lösung der quadratischen Gleichung: $f(x) = ax^2 + bx + c$

lautet
$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Lesen die Koeffizienten a, b, c von der Konsole (Methode `gets`) und berechnen Sie die Lösung der quadratischen Gleichung. Bestimmen Sie ob es eine ($b^2 - 4ac == 0$), zwei ($b^2 - 4ac > 0$) oder keine ($b^2 - 4ac < 0$) reelle Lösung gibt. Geben Sie diese Information und das ggf. das Ergebnis der Berechnung auf der Konsole aus.

Wenn die Eingaben nicht korrekt sind, d.h. a, b, c keine gültigen Zahlen sind, dann soll das Programm abbrechen.

Teil C: Approximation der Exponentialfunktion mit Fehlerbestimmung

Schreiben Sie eine Methode für die nachfolgende Näherungsformel der Exponentialfunktion:

$$e^x = \sum_{i=0}^n \frac{x^i}{i!}$$

Schreiben Sie auch eine Methode zur Berechnung der Fakultät.

Bewerten Sie Ihre Lösung: Wie effizient ist diese? Werden Berechnungen mehrfach durchgeführt? Wenn ja, beseitigen Sie Mehrfachberechnungen.

Schreiben Sie eine Methode, die die Anzahl der Schritte bestimmt, bis die Näherungsformel von dem Wert e^x kleiner einem vorgegebenen Fehler abweicht. Die Methode hat zwei Parameter, das x und den Wert für den zulässigen Fehler. Die Methode gibt die Anzahl der Schritte zurück. Zur Berechnung des Wertes e^x in Ruby verwenden Sie die Methode `exp` im Modul `Math`.

Teil D: Folge von Potenzen berechnen

Schreiben Sie eine Methode, die zu einer gegebenen Zahl x und einer oberen Grenze n, eine Folge von Potenzen berechnet und ausgibt solange die Potenz $\leq n$ ist.

Also die Folge: x^0, x^1, \dots, x^k

Denken Sie auch hier über eine effiziente Lösung nach.

Die Methode soll das k zurückgeben für das $x^k \leq n$ ist.

Teil E: Farbumwandlung

Für Farbdarstellungen sind unterschiedliche Formate bekannt. Das **RGB** Format für Bildschirme, Digitalkameras etc., das **CMYK** Format für den Druck von Büchern etc. **RGB** kodiert Farbanteile in Rot (R), Grün (G) und Blau (B) Werten in Bereich von 0-255. **CMYK** kodiert Farbanteile in Cyan (C), Magenta (M), Yellow (Y) und Black (K) Werten auf einer reellen Skala von 0.0-1.0.

Schreiben Sie eine Methode, die RGB-Werte in CMYK Werte umwandelt. Sind alle RGB Werte 0, dann sollen die CMY Werte 0.0 sein und der K Wert 1. Ansonsten verwenden Sie die folgenden Formeln zur Berechnung des Farbwertes:

$$w = \max((r/255), (g/255), (b/255))$$

$$c = (w - (r/255))/w$$

$$m = (w - (g/255))/w$$

$$y = (w - (b/255))/w$$

$$k = 1 - w$$

Teil E: Wochentag berechnen

Schreiben Sie eine Methode die aus Monat (m), Tag (d) und Jahr (y) den Wochentag (d_0) berechnet. Übergeben Sie 1 für Januar, 2 für Februar etc., 0 für Sonntag, 1 für Montag etc. Verwenden Sie die folgenden Formeln des gregorianischen Kalenders zur Berechnung.

$$y_0 = y - \frac{14 - m}{12}$$

$$x = y_0 - \frac{y_0}{4} - \frac{y_0}{100} + \frac{y_0}{400}$$

$$m_0 = m + 12 * \frac{14 - m}{12} - 2$$

$$d_0 = \left(d + x + \frac{31 * m_0}{12} \right) \% 7$$

Teil F: Prüfziffern berechnen

ISBN ist ein 10-stelliger Code zur Kennzeichnung von Büchern. Die Ziffer ganz rechts ist die Prüfziffer, die eindeutig aus den übrigen 9 Ziffern berechnet werden kann, aufgrund der Formel:

$$d_1 + 2 * d_2 + 3 * d_3 + 4 * d_4 + 5 * d_5 + 6 * d_6 + 7 * d_7 + 8 * d_8 + 9 * d_9 + 10 * d_{10} = n * 11$$

In der Formel entspricht d_i der i'ten Ziffer von rechts gelesen beginnend mit 1 (d_1 ist die Prüfziffer).

Schreiben Sie ein Programm, das eine 9-stellige Zeichenkette einliest und aus dieser Zeichenkette mit der oben stehenden Formel die Prüfziffer berechnet und die ISBN Nummer ausgibt.

Teil G: Würfeln

Schreiben Sie eine Methode, die 3 unterschiedliche Zahlen im Intervall $[1,6]$ berechnet und diese Zahlen ausgibt. Merken Sie sich in Variablen, $v1-v6$, wie häufig die einzelnen Zahlen gewürfelt wurden.

Schreiben Sie eine Methode, die das Würfeln einer Zahl simuliert. Die Methode gibt die Anzahl der Versuche zurück, bis die gesuchte Zahl gewürfelt wurde.

Schreiben Sie eine Methode, die das simulierte Würfeln einer Zahl n -mal wiederholt und am Ende die durchschnittliche Anzahl der Treffer zurückgibt.

Wie verhalten sich die Methoden, wenn diese mehrfach aufgerufen werden? Lässt sich dieses Verhalten ändern? Wenn ja, wie?

Teil F: Random Walk in 2D

Schreiben Sie eine Methode, die eine Zufallsbewegung eines Teilchens auf einem 2D Feld simuliert und schätzt, wie lange das Teilchen benötigt, um bis zum Rand des Feldes zu gelangen. Das Teilchen bewegt sich mit einer Wahrscheinlichkeit von $\frac{1}{4}$ in nördliche, südliche, westliche oder östliche Richtung unabhängig von den vorausgehenden Bewegungen. Der Methode wird der Parameter n übergeben, der ein $2n \times 2n$ Spielfeld beschreibt.

Zusatz freiwillig: Simulieren Sie den Random Walk in der Toolbox.