

Einführung in die wissenschaftliche Programmierung

Übungsblatt 1

1) Python als Taschenrechner

Benutzen Sie Python um eine Temperatur von Grad Fahrenheit nach Grad Celsius umzuwandeln. Sie benötigen dazu die Formel

$$C = \frac{5}{9} (F - 32).$$

- i) Legen Sie eine Variable `F` mit dem Wert 47 an.
- ii) Wandeln Sie dies in Grad Celsius um und speichern Sie das Ergebnis in `C`. Geben Sie dann `C` mit der `print` Funktion aus.

2) Die math Bibliothek

Python bietet eine große Anzahl an Bibliotheken. In dieser Aufgabe wollen wir einige Funktionen und Konstanten der `math` Bibliothek verwenden. Importieren Sie dazu die Bibliothek mit `import math`.

a) Kreisfläche

Geben sei der Radius in der Variable `r`. Berechnen Sie die Kreisfläche. Sie benötigen dazu die Konstante π welche Sie in `math.pi` finden. Speichern Sie das Ergebnis in `A` und geben Sie es mit `print` aus.

b) Quadratische Gleichung

Es soll die quadratische Gleichung

$$ax^2 + bx + c = 0, \quad a, b, c \in \mathbb{R}, a \neq 0$$

gelöst werden. Es gilt

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

Benutzen Sie die Bibliotheksfunktion `math.sqrt` um die Formel auszuwerten, und geben Sie die Lösung für $a = 0.2, b = 2$ und $c = 1$ aus. Legen Sie hierzu zuerst Variablen `a`, `b` und `c` mit den angegebenen Werten an, und speichern Sie das Ergebnis in `x1` und `x2`.

3) Aggregatzustand von Wasser

In Aufgabe 1 haben Sie ein Programm geschrieben, welches eine Temperatur F von Grad Fahrenheit in Grad Celsius umrechnet. Nehmen Sie nun an, es handelt sich dabei um die Temperatur von Wasser. Erweitern Sie dann das Programm so, dass noch zusätzlich zur Temperatur der Aggregatzustand ausgegeben wird, d.h. "gasförmig", "flüssig" oder "fest".

4) Monte Carlo Methode: Approximation von π

Mit der Hilfe von Zufallszahlen soll die Kreiszahl π approximiert werden. Man erzeugt dazu Zufallszahlen aus $[0, 1]^2$ (Einheitsquadrat) und zählt wieviele Punkte davon im Viertelkreis mit Mittelpunkt $(0, 0)$ und Radius $r = 1$ liegen. Das Verhältnis der Anzahl der Punkte im Kreis zur Anzahl der Punkte insgesamt ergibt eine Näherung von $\pi/4$. Implementieren Sie diese Methode auf zwei Arten:

- i) Mit einer `for`-Schleife, wo Sie bestimmen wieviele Punkte insgesamt verwendet werden soll.
- ii) Mit einer `while`-Schleife, wo Sie solange neue Punkte erzeugen, bis eine vorgegebene Genauigkeit ϵ erreicht ist. Verwenden Sie die eingebaute Konstante `math.pi` als Referenzwert.

Hinweis: Verwenden Sie die `random` Bibliothek für die Erzeugung von Zufallszahlen. Nach `import random`, können Sie mit `x = random.random()` eine Zufallszahl zwischen $[0, 1]$ erzeugen.