Übung 1 - Einstieg

Programmiervorkurs 2025

1 Python Installieren

Um die Aufgaben dieses Programmiervorkurses bearbeiten zu können, benötigt man eine (halbwegs aktuelle) Python-Installation.
Als Code-Editor ist jeder übliche Texteditor geeignet, es benötigt keine Entwicklungsumgebung (IDE), von welchen wir im Rahmen dieses Vorkurses eher abraten, um einen größeren Lerneffekt zu erzielen.
Im Moodle haben wir dafür die Anleitung ”Python Installieren für Dummies” hinterlegt.

Auf den Poolrechnern ist Python bereits installiert, dort muss der Befehl **python3** verwendet werden.

2 Theoriefragen

Bitte ordne folgende Aussagen als wahr oder falsch ein:

wahr	falsch	Frage
☐	☐	In Python ergibt der Ausdruck `5 / 2` den Wert `2,5`.
☐	☐	`5y` ist ein valider Variablenname.
☐	☐	`"2"+ 5` wird ohne Fehler ausgeführt.
☐	☐	`float(int(str(int("3"))))` ist ein valider Ausdruck.
☐	☐	Es macht einen Unterschied, ob ich Tabs oder Spaces für die Einrückung verwende.

3 Ausdrücke

3.1 Code-Verständnis

Überlege für jeden der folgenden Ausdrücke, ob sie korrekten Python-Code darstellen und von welchem Datentyp sie in diesem Fall sind.

- 2
- 10.0
- true
- ’Lorem’
- True
- ”ipsum”

3.2 Mathe 0 für Informatiker

Wenn Programme nur das zurückgeben würden, was man hinschreibt, hätte man noch lange kein vollständigen Computer. Deswegen möchte man auch Berechnungen durchführen können. Überlege dir, was die folgenden Ausdrücke ergeben, und für welche Rechnenoperation die verwendeten Symbole stehen. Beachte dabei die Präzedenz der einzelnen Operatoren. Lasse dir auch immer den Datentyp des Ergebnisses ausgeben.

- `16 + 26`

- `12 * 123 // 28 - 15`
- `5 * 3 + 6 * 6`
- `13.75 + 28.67`
- `2**6 + 2**3 + 2**1`
- `5.4 * 7.7`

4 Konvertierung

4.1 Grundlagen

Führe folgende Konvertierungen durch und notiere, was die Ergebnisse sind und welche davon nicht funktionieren:

1. 3.17 -> int
2. 'a' -> int
3. 5 -> float
4. '5' -> int
5. 3.1415 -> string
6. 5 -> bool
7. 'false' -> bool

4.2 Überkomplizierte Eingabe

Im Folgenden soll eine Ganzzahl unter Nutzung der Stringkonstante `"1"` und der Zahlenkonstante `0` sowie der Konvertierungsfunktionen `int()` und `str()` ausgewertet werden. Arithmetische und String-Operationen sind erlaubt, allerdings keine weiteren Konstanten wie z.B. 2 oder "19".
Beispiel: Mit `"1"+ "1"+ str(0)` bekommt man das Ergebnis `"110"`. Dieses kann wiederum in einen Integer umgewandelt werden, um damit weiter zu rechnen.
Kombiniere auf diese Weise die Konstanten, sodass sich am Ende die Zahl `219` ergibt.

5 Variablen

5.1 Fehlersuche

Gegeben ist folgendes Listing:

```
1 meinAlter = 21
2 meinalter = meinalter + 1
3 print(meinAlter)
```

Beim Ausführen der zweiten Zeile wirft Python einen Fehler. Warum? Verbessere das Listing.

5.2 Gültigkeit & Konvention

a) `matrikel_nummer`
b) `ÄhmKeineAhnung`
c) `2fancy4python`
d) `_meinAlter`
e) `richtig&gut`

f) `_2Euro`

g) `Variable2`

h) `noch_besser`

i) `#Falsch`

5.2.1 Welche der aufgelisteten Variablennamen sind gültige Variablennamen?

5.2.2 Welche entsprechen der Namenskonvention snake_case?

6 RSA (Bonus)

Solltest du bei dieser Aufgabe Fragen zur verwendeten Mathematik oder Schreibweise haben, frag uns gerne. Für eine RSA-Verschlüsselung werden zwei Primzahlen p, q benötigt. Deren Produkt definiert einen Ring $\mathbb{Z}_n$ mit $n = p \cdot q$. Das heißt, nach jeder Operation wir das Ergebnis mittels Modulo in das Intervall $[0, n)$ gebracht.

Wenn man nun etwas verschlüsseln/entschlüsseln möchte, benötigt man zunächst $N = (n) = (p-1) \cdot (q-1)$. Außerdem wird ein $0 < e < N$ gebraucht, mit dem der größte gemeinsamen Teiler von e und N 1 ist sowie ein $0 < d < N$ mit $e \cdot d \; mod \; N = 1$.

Da das hier keine Krypto-Vorlesung ist, soll die Berechnung dieser Werte übersprungen werden und es sind folgende Werte vorgegeben: $p = 3$, $q = 11$, $n = 33$, $N = 20$, $e = 3$, $d = 7$. Eine Nachricht m wird nun als $m^e \; mod \; n$ verschlüsselt, ein Chiffretext c als $c^d \; mod \; n$ entschlüsselt.

Verschlüssele mit den vorgegebenen (oder eigenen) Werten nun ein paar Zahlen ($0 \leq x < n$), und probiere, ob du sie wieder entschlüsseln kannst. Du kannst diese auch an euren Sitznachbarn weitergeben.