

Einführung in die Programmierung mit Skriptsprachen

V04: Erste Schritte in Python vom 08.11.2022

Aufgabenblatt 3

1. Spickzettel

Der Spickzettel 2 und 3 befinden sich im Ordner unter 2. Spickzettel.pdf und 3. Spickzettel.pdf.

2. Module in Python

Erklären Sie, welche Auswirkungen die unterschiedlichen Arten der Einbindung von Modulen haben, die in der Vorlesung gezeigt werden.

<code>import math</code>	Importiert den ganzen Inhalt des Moduls
<code>import math as mathematikmodul</code>	Importiert den ganzen Inhalt des Moduls und ändert den Namen des Moduls in meinem Programm.
<code>from math import</code>	Fehler, weil er nicht Weiß auf welchen Inhalt er zugreifen soll
<code>from math import sin</code>	Importiert aus dem Mathe Modul die Funktion sin
<code>from math import sin as sinus</code>	Importiert aus dem Mathe Modul die Funktion sin und ändert den Namen der Funktion

3. Pseudocode lesen

Implementieren Sie den Code nicht, sondern verstehen Sie ihn durch Lesen. $a[i]$ meint hier den i -ten Buchstaben von a . Der $+$ -Operator hängt hier zwei Strings hintereinander.

Welchen Output produziert das folgende (Pseudocode-) Programm?

```
a = "python"
for i = 1 to 3
  for j = 2 to 7
    a = a + a[j-1]
  print (a)
```

Ausgabe: pythonythonython

Teilaufgabe 3.2:

Wie müsste das Pseudocodeprogramm aussehen, wenn der erste Buchstabe $a[0]$ ist, nicht $a[1]$, d. h. man fängt bei 0 an zu zählen?

```
a = "python"
for i = 1 to 3
  for j = 1 to 7
    a = a + a[j-1]
  print (a)
```

Ausgabe: pythonpythonpython

4. Pseudocode schreiben, Schleifen

Entwickeln Sie in Pseudocode mit Schleifen ein Programm, das folgenden Output erzeugt:

```
sternchen = ""
for i 1 to 6
  for j 1 to i
    for k 1 to i
      sternchen = sternchen + "*"

print(sternchen)
sternchen = ""
```

Teilaufgabe 4.1:

In Python gibt es keine repeat-until-Schleife. Finden Sie ein Python-Äquivalent zu der folgenden Pseudocode-Schleife, d. h. die Funktionalität muss gleich bleiben:

```
input(a)
repeat
  print a
  a = a + 1
until a >= 10
print(a)
```

in Python Sprache:

```
a = int(input())
b = 0
while 10 != b:
  print(a)
  a = a + 1
  if a >= 10:
    b = 10
print(a)
```

Man trägt in der Konsole für a eine Zahl ein und die eingegebene Zahl wird ausgegeben und danach wird die vorherige Zahl in a plus 1 gerechnet.

Dann findet wieder eine Ausgabe statt. Das ganze passiert solange bis 10 erreicht ist

Dann am Ende gibt es eine Ausgabe noch von a .

Teilaufgabe 4.2:

Mit welchen Zahlen müssen Sie Ihr Programm testen, um festzustellen, ob die Funktionalität stimmt?

Antwort: 1 und 10 und 11.