

Aufgaben

1. Aufgabe

(Bereich 1, 5 Punkte)

Schreibe ein Programm, das für eine zufällig, i.e. via `randint` bestimmte Zahl zwischen 1 und 100 die Teiler in eine Liste schreibt und Zahl und Liste ausgibt.

2. Aufgabe

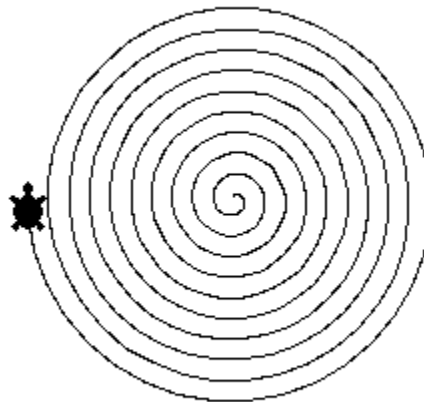
(Bereich 1, 5 Punkte)

Schreibe ein Programm "Countdown", das einen Zähler von 10 an um 1 abnehmend rückwärts zählen lässt (inkl. Bildschirmausgabe) und bei 0 das Kommando "Start" ausgibt.

3. Aufgabe

(Bereich 2, 7.5 Punkte)

Schreibe ein Programm, das eine kreisförmige Spirale zeichnet, s. Bild:



4. Aufgabe

(Bereich 1, 5 Punkte)

Schreibe ein Programm, das beliebig viele (zwischen 1 und 10) einstellige Zahlen in eine Liste schreibt und dann aus ihnen den Durchschnitt bildet. Die Liste musst du mit `randint` füllen (list comprehension, du brauchst `randint` für die einstelligen Zahlen und für die Anzahl). Ausgabe bitte mit Liste, Summe und Durchschnitt.

5. Aufgabe

(Bereich 2, 7.5 Punkte)

Erläutere die Ausgabe folgenden Codes, indem du für jeden Schleifendurchlauf die Inhalte von `my_list` aufschreibst und die Funktion von `i` erklärst:

```
my_list = [1, 10, 100, 1000]
```

```
for i in range(4):
```

```
    my_list[i] = 2*my_list[i]
```

```
    print(my_list)
```

Zweitens: Erläutere den Unterschied zwischen `2*my_list[i]` und `my_list[i*2]`, indem du die Wirkung von letzterem benennst.

6. Aufgabe

(Bereich 3, 10 Punkte)

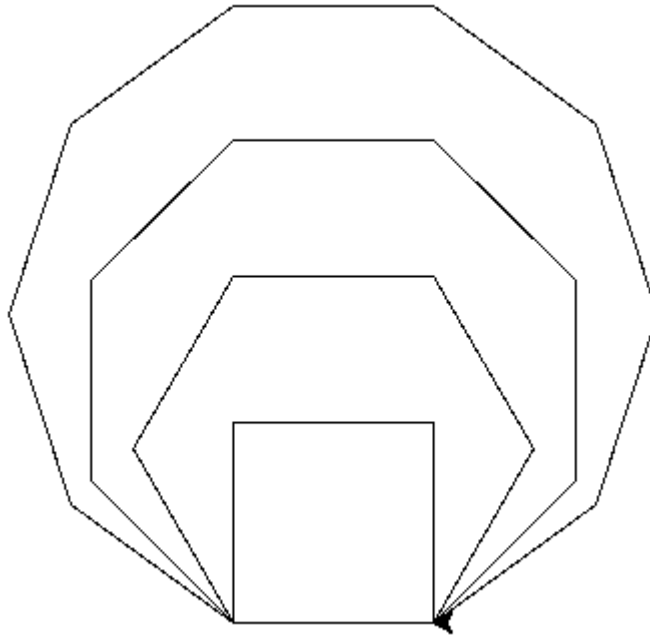
Belege 2 Variablen mit je einer Zufallszahl zwischen 1 und 100. Schreibe dann eine Funktion `ggT`, die von beiden Zahlen den größten gemeinsamen Teiler ermittelt.

7. Aufgabe

(Bereich 2-3, 10 Punkte)

Schreibe eine Funktion, die ein Polygon zeichnet, dessen Eckenzahl sich um jeweils 2 erhöht, z.B. 4, 6, 8, 10, wie im Bild:

(Tipp: mit 2 Funktionen ist es am leichtesten. Winkel = $360 / \text{Zahl der Ecken}$)



8. Aufgabe (Zusatzaufgabe)

(Bereich 3, 10 Punkte)

Erstelle mit `randint` eine Liste, die eine Binärzahl mit 5 Stellen erzeugt (z.B. `liste=[01101]`). Schreibe dann eine Funktion, die die Liste z.B. mittels einer `for`-Schleife, die über das Hochzählen der Indizes die jeweils benötigten Listenwerte ermittelt, als Dezimalzahl ausgibt.

Hinweis: Du benötigst 2 Variablen, Potenz und Index, die du hochzählen musst. S. hier:

letzte Stelle = `liste[-1]*2**0`

vorletzte Stelle = `liste[-2]*2**1`

"" "" = `liste[-3]*2**2`

"" "" = `liste[-4]*2**3`

"" "" = `liste[-5]*2**4`