

4.2.5 Gemischte Übungen

Donnerstag, 20. April 2023 16:08

4.2.5. Gemischte Übungen

- Ein Würfel wird so lange geworfen, bis die Summe aller Würfe 100 oder mehr beträgt. Zum simulieren eines Würfels kannst du die Funktion `randrange()` aus dem Modul `random` benutzen.

- Bestimme mit einem Programm, wie viele Würfe dazu nötig sind.
- Schreibe ein Programm, welches mit Hilfe einer Liste zählt, wie oft eine bestimmte Augenzahl geworfen wurde. Am Ende soll die Liste mit `print()` ausgegeben werden.

```
from random import randint
all_thrown_dice = []
sum = 0
counter = 0
while sum < 100:
    counter += 1
    random_dice = randint(1,6)
    all_thrown_dice.append(random_dice)
    sum += random_dice
print("amount of needed dice: ",counter)
for y in range(1,7):
    print(f"y: {len([x for x in all_thrown_dice if x == y])}")
```

- Der Spieler hat 7-mal die Chance die vom Computer durch Zufall festgelegte Zahl zu erraten. Als Rückmeldung bekommt er nur:

- deine geratene Zahl ist zu groß
- deine geratene Zahl ist zu klein
- Gewonnen! Die geheime Zahl ist nicht mehr geheim

```
#2. Aufgabe
from random import randint
counter = 0
random_nr = randint(1,100)
guessed_nr = 0
while guessed_nr != random_nr:
    if counter == 7:
        print(f"you lost, the random number was {random_nr}")
        break
    guessed_nr = int(input("guess a number between 1 and 100(both included)"))
    counter += 1
if guessed_nr == random_nr:
    print("the random number is higher than your guessed number")
elif guessed_nr > random_nr:
    print("the random number is lower than your guessed number")
else:
    print(f"you won, the number was {random_nr}")
```

Wenn der Spieler gewonnen hat, kommt eine entsprechende Meldung. Hat der Spieler die Zahl nicht erraten, kommt als Nachricht: „Schade – verloren. Einfach nochmals probieren“.

3. Lotto die zweite



Erstellen Sie ein Lottozahlenspiel.

Der Nutzer soll 6 einzelne Zahlen nacheinander eingeben können. Prüfen Sie, ob die Zahlen im Bereich von 1-49 liegen. Weiterhin darf jede Zahl pro Tipp nur einmal eingegeben werden. Füllen Sie alle gültigen Zahlen in eine Liste.

Das Programm soll nun sechs Zufallszahlen ziehen und die beiden Listen miteinander vergleichen.

- Die Ausgabe soll angeben, wie viele Richtige getippt wurden.

4. Schreibe ein Programm, welches alle Primzahlen zwischen 1 und einer vom Benutzer gewählten oberen Grenze ausgibt.

Schon fertig? Setze die Struktogramme in Python um.

```
# 4. Aufgabe
end = int(input("upper border"))
number = 0
primes = []
while number < end:
    number += 1
    is_prime = False
    for i in range(2, number):
        if (number%i) == 0:
            break
    is_prime = True
    if is_prime:
        primes.append(number)
print(primes)
```