

# LU04.A03: Münzenspiel

## Spielregeln



Lösen Sie die Aufgabe mit [BlockPy](#) oder [Codingrooms](#)

Laden Sie einen Screenshot ihres Blockly-Programms hoch.

1. Zu Beginn legen die Spieler einige Münzen (10 - 30) auf den Tisch.
2. Zwei Spieler nehmen nun abwechselungsweise eine, zwei oder drei Münzen vom Tisch.
3. Gewonnen hat der Spieler, der zuletzt Münzen vom Tisch genommen hat.

## Ablauf

1. Erzeuge eine Zufallszahl für die Anzahl Münzen
2. Lege den aktuellen Spieler = 1 fest
3. Solange Münzen auf dem Tisch liegen
  - 3.1. Zeige die Anzahl Münzen an
  - 3.2. Der aktuelle Spieler gibt ein, wieviele Münzen er nimmt
  - 3.3. Subtrahier die Anzahl Münzen vom Stapel
  - 3.4. Lege den anderen Spieler als aktuellen Spieler fest
4. Gib den Sieger aus

## Auftrag

Schreiben Sie ein Programm, um das Spiel zu spielen.

## Ablauf

Willkommen zum Münzenspiel  
Es sind noch 11 Münzen auf dem Stapel  
Spieler 1 nimm 1-3 Münzen > 3

Es sind noch 8 Münzen auf dem Stapel  
Spieler 2 nimm 1-3 Münzen > 7  
Du darst höchstens 3 Münzen nehmen  
Es sind noch 8 Münzen auf dem Stapel  
Spieler 2 nimm 1-3 Münzen > 2

Es sind noch 6 Münzen auf dem Stapel  
Spieler 1 nimm 1-3 Münzen > 0  
Du musst mindestens 1 Münze nehmen

Es sind noch 6 Münzen auf dem Stapel  
Spieler 1 nimm 1-3 Münzen > 3

Es sind noch 3 Münzen auf dem Stapel  
Spieler 2 nimm 1-3 Münzen > 3

Spieler 1 hat verloren  
Bis bald

## Hinweise

### Zufällige Anzahl Münzen

Zu Beginn des Programms wird eine zufällige Anzahl Münzen generiert. Dies geschieht mit den Befehlen

```
import random  
coin_total = random.randint(11,20)
```

1. `import random` lädt die Bibliothek zum Erzeugen von Zufallszahlen.
2. `random.randint(n, m)` erzeugt eine Zufallszahl zwischen `n` und `m`.

Sie brauchen das Ergebnis dieses Befehls nur in einer Variable zu speichern.

### Aktueller Spieler

Die Spieler 1 und 2 ziehen abwechselungsweise. Für die Anzeige des jeweils aktuellen Spielers, benötigen wir eine Variable. Diese Variable erhält am Anfang den Wert 1, also ist Spieler **1** am Zug.

Hat der Spieler eine gültige Anzahl eingegeben, so wechselt der Spieler. Dazu rechnen wir `Aktueller Spieler = 3 - Aktueller Spieler`.

- Ist der aktuelle Spieler = 1, so ist das Resultat der Rechnung 2.
- Ist der aktuelle Spieler = 2, so ist das Resultat der Rechnung 1.

Das Resultat wird in der Variable für den aktuellen Spieler gespeichert.

## Vorgehen

### 1. Variablen

1. Bestimmen Sie alle benötigten Variablen mit Bezeichner (Name).
2. Geben Sie jeder Variable einen Initialwert.

## 2. Ein- / Ausgaben

Realisieren Sie alle Ein- und Ausgaben

## 3. Selektionen und Iterationen

Realisieren Sie schrittweise die Selektionen und Iterationen.

## 4. Ausprobieren

Spielen Sie das Spiel gegen sich selber oder einen Kollegen. Prüfen Sie auch, wie das Programm auf ungültige Züge (z.B. 0 Münzen, 5 Münzen) reagiert.

Zeigt das Programm den Sieger korrekt an?

---

[M319-D1F, M319-F3G](#)



© Marcel Suter, Kevin Maurizi

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

<https://wiki.bzz.ch/modul/archiv/m319python/learningunits/lu04/aufgaben/muenzenspiel>

Last update: **2024/03/28 14:07**

