

Arbeitsauftrag

1. Öffnen Sie die Webseite: inf-schule.de/6.1.2.1
2. Lesen und Bearbeiten Sie die **Lernstrecke 2: Wiederholungen**.
3. Bearbeiten Sie zu den jeweiligen Teilabschnitten, die Aufgaben auf dem Arbeitsblatt.



Hinweis: Das Dokument „Wissensspeicher“ auf inf-Schule.de soll nicht bearbeitet werden.
Ausschließlich, die Aufgaben auf diesem Arbeitsblatt sind zu bearbeiten und schriftlich festzuhalten!

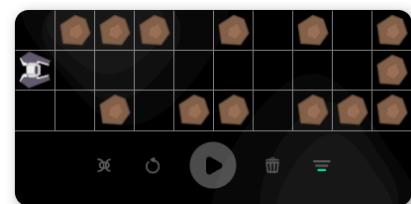
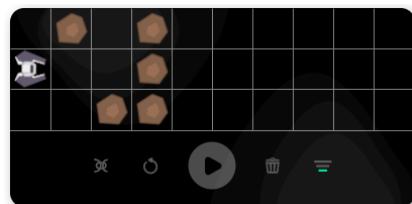
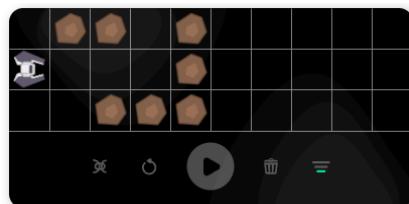
Aufgabe 0

Szenario 1	Szenario 2	Szenario 3
<pre>move() turnRight() move() putPow() turnLeft() turnLeft() move() turnRight() move() move() turnLeft() move() putPow() turnRight() turnRight() turnRight() move() turnLeft()</pre>	<pre>move() turnRight() move() putPow() turnLeft() turnLeft() move() turnRight() move() turnLeft() move() putPow() turnRight() turnRight() move() turnLeft()</pre>	<pre>move() turnRight() move() putPow() turnLeft() turnLeft() move() turnRight() move() move() move() move() move() move() move() turnLeft() move() putPow() turnRight() turnRight() move() turnLeft()</pre>

Aufgabe a) (Erkundung)

Schreiben Sie **ein** Programm, dass die Anforderungen der Missionszentrale für die drei gegebenen Szenarien, aus **Aufgabe 4 (Erkundung)** löst.

```
move()
turnRight()
move()
putPow()
turnLeft()
turnLeft()
move()
turnRight()
while not rockFront():
    move()
turnLeft()
move()
putPow()
turnRight()
turnRight()
move()
turnLeft()
```



Aufgabe b) (Erkundung)

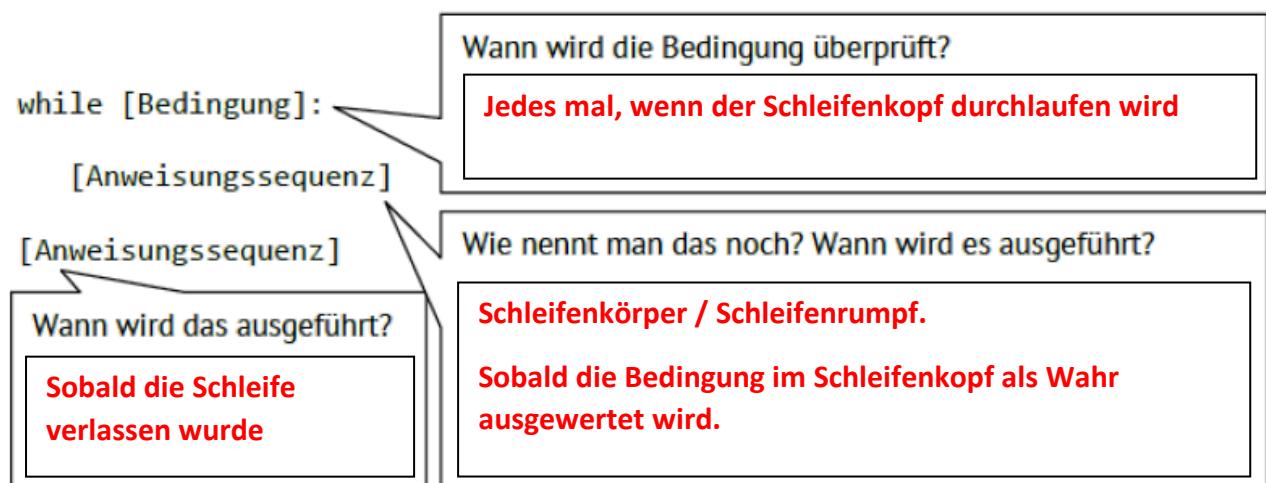
Formulieren Sie in eigenen Worten, welche **Vorteile** das Konzept „Wiederholung“ in der Programmierung bietet.

- Reduzierung von Programmcode
- Übersichtlicher
- Ein Programmcode für mehrere Anwendungsfälle

Aufgabe c) Aufbau einer Schleife beschreiben (Strukturierung)

Lesen Sie sich den Teilabschnitt „**Strukturierung: Wiederholte Vorgänge**“ und **Aufgabe1** auf der Seite „**Strukturierung**“ aufmerksam durch und bearbeiten Sie anhand dieser Informationen die folgenden Aufgaben.

- I. Markieren Sie im Beispiel-Quelltext in Farbe, worauf man unbedingt achten muss. Beschreibe rechts, was wann ausgeführt wird



- II. Erklären Sie, wann und wie oft die einzelnen Anweisungssequenzen ausgeführt werden.

```
[Anweisungssequenz 1]
while [Bedingung]:
    [Anweisungssequenz 2]
    [Anweisungssequenz 3]
```

Anweisungssequenz 1 → einmal

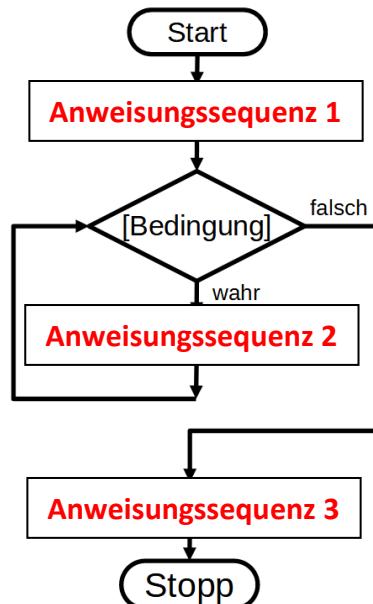
Anweisungssequenz 2 → Solange die while Bedingung = True

Anweisungssequenz 3 → einmal

Aufgabe d) Flussdiagramme für Schleifen (**Strukturierung**)

Das folgende Flussdiagramm stellt den schematischen Aufbau einer Schleife dar.

- a) Ordne die drei Anweisungssequenzen im Flussdiagramm den drei Anweisungssequenzen aus **Aufgabe c) II** zu: Welches ist Anweisungssequenz 1, welches 2, welches 3?

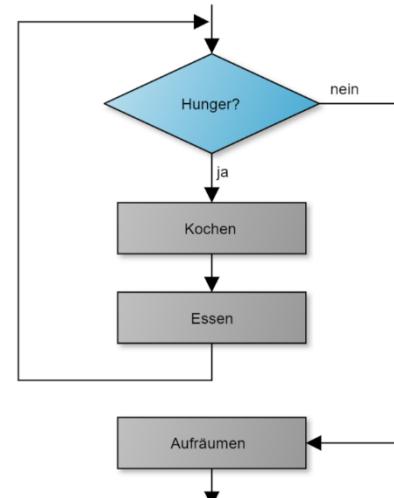


- b) Überlegen Sie sich:

- Wie oft wird ein Schleifenrumpf mindestens durchlaufen?
0-Mal, wenn der Schleifenkopf als false ausgewertet wird.
- Wie oft wird ein Schleifenrumpf maximal durchlaufen?
Unendlich, wenn der Schleifenkopf nie als false ausgewertet wird.

- c) Erstellen Sie auf einem Blatt ein Flussdiagramm zu einem selbstgewählten Problem. Das Problem kann auch etwas aus dem Alltag sein. Sollte Ihnen keins einfallen, nehmen Sie eins aus den drei Szenarien von der Seite „**Erkundung**“. Rechts sehen Sie ein Beispiel für ein Flussdiagramm aus dem Alltag.

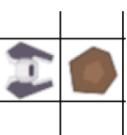
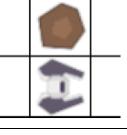
Quelle: <https://informatik.mygym.ch>



Aufgabe e) Argos Sensoren - Elementare Bedingungen (Fachkonzept)

Lesen Sie sich die Seite „**Fachkonzept**“ aufmerksam durch und bearbeiten Sie anhand dieser Informationen die folgenden Aufgaben.

- a) Die Argo kann mit Sensoren ihre Umgebung wahrnehmen. Links sehen Sie vier Situationen, die die Argo erkennen kann. Beschreibe die Bedeutung und nenne den entsprechenden Befehl.

Symbol	Bedeutung	Befehl
	<i>Ist vor mir ein Asteroid?</i>	<i>rockFront()</i>
	<i>Ist links von mir ein Asteroid?</i>	<i>rockLeft()</i>
	<i>Ist rechts von mir ein Asteroid?</i>	<i>rockRight()</i>
	<i>Fliege ich über einem PowerUp?</i>	<i>onPow()</i>

- b) Erklären Sie, wozu diese Bedingungen benötigt werden.

Die Steuerung komplexerer Abläufe (wie Fallunterscheidungen und Wiederholungen) erfolgt mit Hilfe von **Bedingungen**.

Entsprechend der Bedingung wird die Schleife anders ausgewertet

- b) Notieren Sie, wie in der Programmiersprache Python, Wiederholungen mit einer SOLANGE-Anweisung dargestellt wird.

while (Bedingung):

Anweisungssequenz

Aufgabe f) (Übung)

Programmieren Sie die Steuerung von Argo, wie in **Aufgabe 1 bis 4** gefordert.



Hinweis: Eine Ergebnissicherung auf dem Arbeitsblatt ist nicht notwendig.

Aufgabe 1

```
# gehe auf das erste Feld mit PowerUp
move()
# Solange auf einem PowerUp: aufheben und weiter gehen
while onPow():
    removePow()
    move()
```

Aufgabe 2

```
# Gehe solange Mauer
while rockRight():
    move()
# umdrehen und ein Schritt zurück
turnLeft()
turnLeft()
move()
```

Aufgabe 3

```
# fliege bis Mauerende
while rockLeft():
    move()
# nach oben fliegen
turnLeft()
move()
move()
turnLeft()
move()
# bis zum PowerUp fliegen
while not onPow():
    move()
# PowerUp entfernen und umdrehen
removePow()
turnLeft()
turnLeft()
```

Aufgabe 4

zum ersten Asteroid gehen

move()

solange rechts ein Asteroid:

while rockRight():

PowerUp legen

 turnLeft()

 move()

 putPow()

 turnRight()

 turnRight()

 move()

 turnLeft()

und weitergehen

 move()

am Ende umdrehen

turnLeft()

turnLeft()



Für die Schnellen!

Vertiefungsaufgabe: (Vertiefung und Vernetzung)

Bearbeiten Sie die Aufgaben 1 bis 3 auf der Seite „**Vertiefung und Vernetzung**“



Hinweis: Eine Ergebnissicherung auf dem Arbeitsblatt ist nicht notwendig.