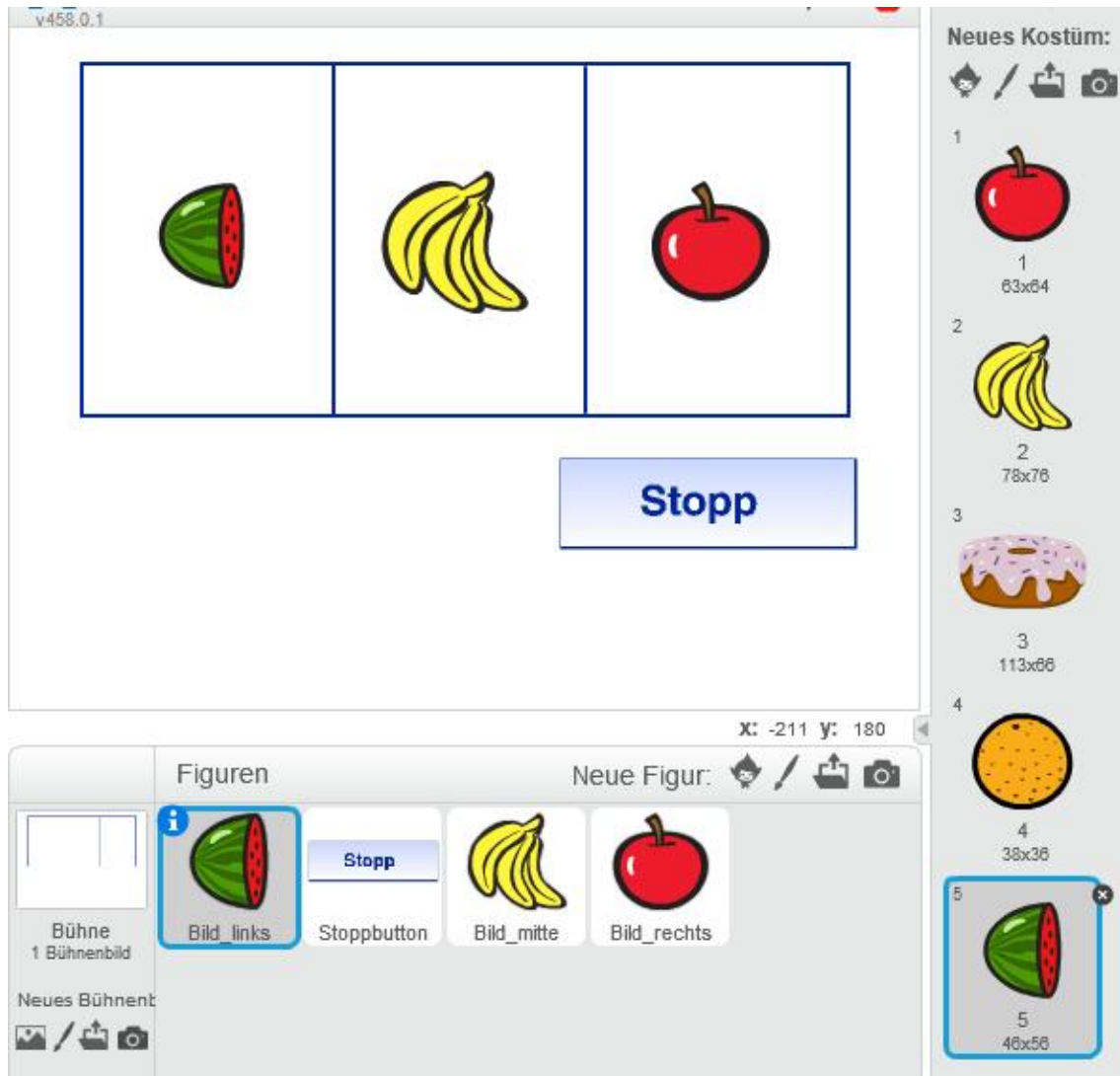




Einarmiger Bandit

Unter dem Begriff „einarmiger Bandit“ versteht man einen Spielautomaten, bei dem man einen Hebel auf einer Seite nach unten drückt (daher „einarmig“). Dadurch drehen sich drei Rollen nebeneinander und können durch einen Knopf gestoppt werden. Wenn drei gleiche Bilder nebeneinander erscheinen, hat man gewonnen. Bei manchen Automaten gibt es auch schon für zwei gleiche Bilder einen Preis.

In Scratch könnte der Automat etwa so aussehen:



Jedes der drei Bilder besitzt fünf Kostüme, die einfach durchnummeriert werden (siehe Abbildung).

Beim Start des Programms (Klick auf die grüne Flagge) sollen die Bilder im Sekundenabstand zufällig wechseln. Wird der Stopp-Button geklickt, bleibt das aktuelle Kostüm sichtbar und es muss ausgewertet werden, ob der Spieler / die Spielerin etwas gewonnen hat.

Bei zwei gleichen soll es 10 Punkte, bei drei gleichen 100 Punkte geben.

Sind alle drei Bilder verschieden können auch Punkte abgezogen werden.



Auftrag für Expertinnen und Experten

1. *Gestalte die Bühne mit den drei Objekten und ihren fünf Kostümen und füge ein weiteres Objekt (Figur) hinzu, das einen Stopp-Button darstellen soll.*
2. *Überlege dir, was alles im Programm passieren muss und schreibe die Schritte auf.*
3. *Vergleiche deine Schritte mit einer Partnerin / einem Partner nach Wahl und besprecht eure Ideen. Entscheidet dann, wie ihr es versuchen wollt.*
4. *Programmiert euren eigenen „einarmigen Bandit“ Schritt für Schritt gemeinsam. Wechselt euch dazu alle zehn Minuten am PC ab.*

Platz für Ideen, Skizzen, Überlegungen:



Auftrag für alle anderen :)

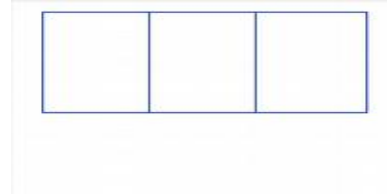
1. *Gestalte die Bühne mit den drei Objekten und ihren fünf Kostümen und füge ein weiteres Objekt (Figur) hinzu, das einen Stopp-Button darstellen soll.
(Hilfekarten 1 – 3)*
2. *Zuerst soll in jedem der drei Felder jede Sekunde ein neues Kostüm gewählt werden. Der Wechsel geht so lange vor sich, bis auf den Stopp-Button gedrückt wird. Überlege dir, welche Kontrollstruktur du hierfür brauchst und wie du ein zufälliges Kostüm auswählen lassen kannst.
(Hilfekarten 4 – 8)*
3. *Wenn sich die Bilder dauerhaft ändern, musst du den Stopp-Button programmieren. Beim Klick auf diesen Button soll zunächst alles stoppen. Es könnte dann ein weiterer Button erscheinen. Wenn man auf diesen klickt, müssen die Bilder verglichen werden. Überlege dir, welche Kontrollstruktur du hier brauchst und wie du das programmieren kannst.
(Hilfekarten 9 – 24)*



Einarmiger Bandit – Hilfekarten

Die Bühne.

1



Zeichne ein Rechteck und unterteile es mit zwei senkrechten Strichen.

Die Figuren (Objekte)

2

Gib den einzelnen Kostümen statt eines Namens eine Nummer.

Der Apfel heißt 1, die Bananen 2, usw.



Der Stopp-Button

3



Nutze ein Rechteck mit Text.

Welche Kontrollstruktur brauchst du für die Wiederholung?

4

Eine Schleife.



Welche Schleife bietet sich hier an?

5

Wir können hier eine Endlosschleife nutzen. Beim Klick auf den Stopp-Button wird dann alles gestoppt.

(Alternativ könntest du auch ohne Stopp-Button arbeiten: „Wiederhole bis Taste ... gedrückt“)

Wie kann man die Kostüme zufällig wechseln lassen?

6

wechsle zu Kostüm Zufallszahl von 1 bis 5

Der Wechsel soll jede Sekunde passieren.

7

warte 1 Sek.

Bisher sollte es etwa so aussehen:

8

```

Wenn [ ] angeklickt
wiederhole fortlaufend
  wechsle zu Kostüm Zufallszahl von 1 bis 5
  warte 1 Sek.
  
```



Welches Ereignis brauchst du für den Stopp-Button?

9



Was soll beim Klick auf den Stopp-Button passieren?

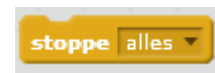
10

Die aktuellen Bilder sollen stehen bleiben und miteinander verglichen werden.

Sind zwei gleich, soll es 10 Punkte geben, sind alle drei gleich, sogar 100.

Wie schaffst du es, dass die Bilder stehen bleiben?

11



Jetzt fehlt noch die Auswertung. Wie könnte man das machen?

12

Um die Bilder vergleichen zu können, müssen wir jeweils wissen, welches Bild gerade aktuell ist.

Die Nummer des Kostüms könnten wir in einer Variable speichern.

Und die Werte dieser drei Variablen können wir dann vergleichen.



Deklaration der Variablen

13

Lege gleich vier neue Variable an:
punkte (für die Punkte bei Gewinn)
links, mitte und rechts (für die jeweils aktuelle Kostümnummer)

Übergabe der Werte an die Variablen

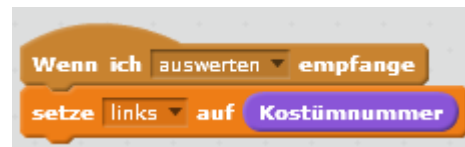
14

Am einfachsten ist es, wenn der Stopp-Button eine Nachricht „Auswerten“ an alle versendet, bevor er alles stoppt.

Dann können die drei Figuren (Objekte) beim Erhalten der Nachricht, die aktuelle Kostümnummer der Variable als Wert übergeben.

Der Code könnte jetzt so aussehen:

15



Das gleiche für die beiden anderen Figuren.

Der Code für den Stopp-Button könnte so aussehen:

16



Der Button versteckt sich gleich und taucht erst wieder auf, wenn neu gestartet wird.



Versuche die Auswertung für drei gleiche Bilder in eigenen Worten zu formulieren

17

„Falls links, mitte und rechts den gleichen Wert haben, erhöhe punkte um 100.“

Du brauchst noch eine neue Figur, den Auswerten-Button

18

Sein Kostüm könnte so aussehen:



Welche Kontrollstruktur brauchst du für die Auswertung?

19

Eine Verzweigung.

Wie lautet der Inhalt der ganzen Verzweigung in eigenen Worten?

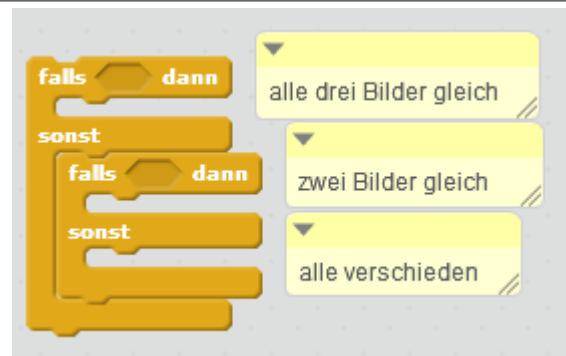
20

Falls alle drei Bilder gleich sind,
erhöhe die Punkte um 100,
sonst
falls zwei Bilder gleich sind,
erhöhe die Punkte um 10,
sonst tue nichts (oder ziehe später ab).



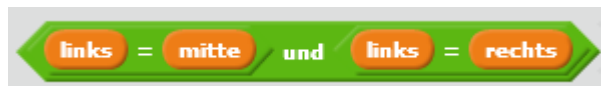
Wie sieht der Code der noch leeren Verzweigung aus?

21



Wie lautet die erste Bedingung?

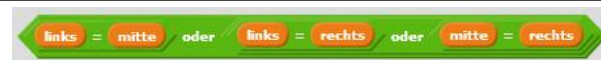
22



Überlege dir, warum diese Bedingung ausreicht, dass alle drei Bilder gleich sind.

Wie lautet die zweite Bedingung?

23

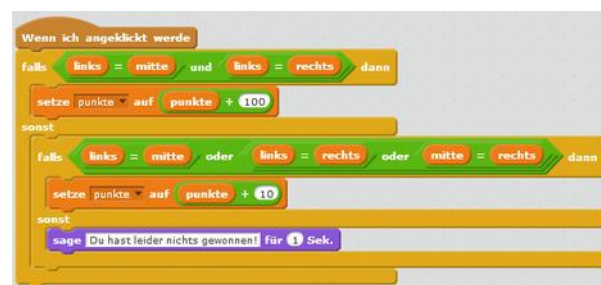


Da in der ersten Bedingung schon abgefragt wurde, ob alle drei gleich sind, reicht es hier aus, jeweils zwei miteinander zu vergleichen.

Das „oder“ verknüpft die drei einzelnen Bedingungen miteinander.

So könnte der Code des Auswerten-Buttons aussehen:

24



Scratch wird von der Lifelong-Kindergarten-Group am MIT-Media-Lab entwickelt. Siehe <http://scratch.mit.edu>.
Scratch ist lizenziert unter [CC BY-SA 2.0](https://creativecommons.org/licenses/by-sa/2.0/deed.en) (<https://creativecommons.org/licenses/by-sa/2.0/deed.en>).