

Programmieren:

8/9

Szenario:
Auf dem Pausenhof behauptet einer deiner Mitschüler, dass Münzen beim Werfen häufiger auf dem „Kopf“ als auf der „Zahl“ landen. Du möchtest ihm mit einem Computerprogramm das Gegenteil beweisen. Hierfür muss in deinem Computerprogramm eine „Wiederholung“ eingebaut werden.

8/9

1. Du kennst verschiedene Arten der Wiederholung in einem Computerprogramm. Nenne zwei davon!

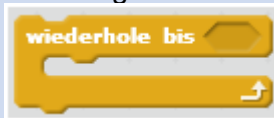
1

Mögliche Lösungen: *Zählwiederholung, bedingte Wiederholung, Endloswiederholung*

Bepunktung: je richtiger Nennung 0,5 Punkte

8/9

2. Eine mögliche Wiederholung beim Programmieren in „Scratch“ siehst du hier:

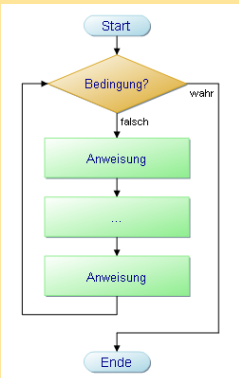


Skizziere dazu ein Flussdiagramm.

4

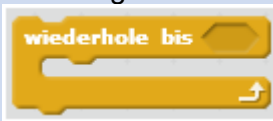
Mögliche Lösung:

Es können im Flussdiagramm eine oder mehrere Anweisungen (Sequenz, wie im Beispiel) vorhanden sein.



Bepunktung:
Start, Anweisung(en), Bedingung, Ende, wahr, falsch je richtiger Verwendung 0,5 Punkte
Verbindungspfeile 1 Punkt

2.* Eine mögliche Wiederholung beim Programmieren in „Scratch“ siehst du hier:



Skizziere dazu ein Flussdiagramm. Verwende dafür

- Verbindungspfeile (z.B. \longrightarrow)
- die Begriffe „wahr“ und „falsch“ sowie
- die Symbole

Alternative



Mögliche Lösung:

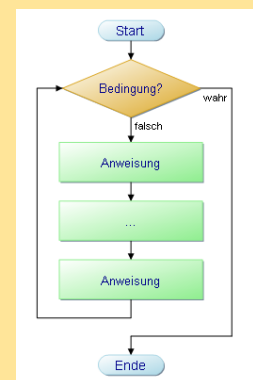
Es können im Flussdiagramm eine oder mehrere Anweisungen (Sequenz, wie im Beispiel) vorhanden sein.

Bepunktung:

Start, Anweisung(en), Bedingung, Ende, wahr, falsch je richtiger

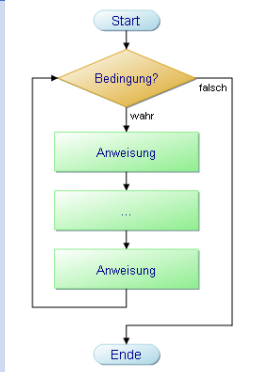
Verwendung 0,5 Punkte

Verbindungspfeile 1 Punkt

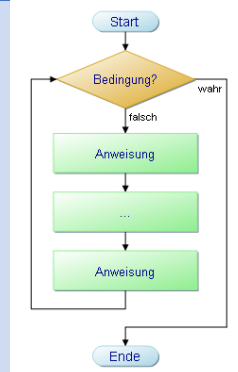


2.* Betrachte die beiden unterschiedlichen Flussdigramme.

Flussdiagramm 1



Flussdiagramm 2



Alternative

- Worin unterscheiden sie sich? Erkläre ausführlich.
- Beschreibe ein Flussdiagramm an einem Beispiel aus deinem Alltag.

Mögliche Lösung für Teilaufgabe a):

Die Begriffe „wahr“ und „falsch“ sind vertauscht. Diese (Wahrheits-)Werte (= boolsche Variablen) wirken sich unterschiedlich auf den weiteren Programmablauf aus.

Bepunktung für Teilaufgabe a):

Erkennen des Unterschieds bei Wahrheitswerten → 1 Punkt

Erklärung mit Fachbegriffen und Folge für Programmablauf → 1 Punkt

Mögliche Lösung für Teilaufgabe b):

Flussdiagramm 1: Frage „Soll ich etwas trinken?“

Start Frage: „Soll ich etwas trinken?“

Bedingung Hast du Durst?

Entscheidung – falsch Frage wird abgebrochen

- wahr: Gehe zum Kühlschrank – öffne Kühlschrank – hole Flasche aus Kühlschrank - fülle Glas - schließe Kühlschrank – trinke –
zurück zu Bedingung

Bepunktung für Teilaufgabe b):

Wählen einer passenden Situation zum gewählten Diagramm → 0,5 Punkte

Verwenden der entsprechenden Fachbegriffe in der sachlichen Darstellung der Situation → 1,5 Punkte



Die Katze lässt sich mit den Pfeiltasten steuern. Nun solltest du noch folgende weitere Aufgabe einprogrammieren:

Wenn die Katze die Maus berührt, sagt sie "Da bist du ja!" Ansonsten soll die Katze sagen: "Wo ist die Maus?"

Folgenden Algorithmus hast du hierzu programmiert:



1) Leider stellst du fest, dass der Algorithmus nicht funktioniert. Welchen der folgenden Bausteine brauchst du, damit der Algorithmus stimmt?

- a)  b)  c) 

2) Erkläre kurz, wie der Fehler zustande kam und warum er jetzt gelöst ist.

Lösung:

1) b) 1P

2) Erklärung:

aktuelle Programmierung prüft nur einmal für zwei Sekunden, ob Bedingung zutrifft 1P
 durch Hinzunahme des Befehls "wiederhole fortlaufend" prüft Scratch die Bedingung nicht nur einmal, sondern so lange der Algorithmus nicht vom Benutzer gestoppt wird 1P

Alternativszenario

8/9

Alternative * **Szenario:** Ein Mitschüler sitzt häufig gelangweilt zuhause und weiß nicht, was er mit seiner Freizeit anfangen soll. Du möchtest ihm helfen und schreibst ein Programm, das ihm, je nach Wetter, Vorschläge zur Freizeitgestaltung macht. Dafür muss er nur auf das entsprechende Wettersymbol klicken. Hierfür benötigst du in deinem Programm eine „Fallentscheidung“.

8/9

Alternative 1. * Du kennst zwei verschiedene Arten der Fallentscheidungen in einem Computerprogramm. Nenne diese!

1

Mögliche Lösungen: Einseitige Fallunterscheidung (auch: falls-Fall), zweiseitige Fallunterscheidung (auch: falls-sonst-Fall)

Bepunktung: je richtiger Nennung 0,5 Punkte

2.* Eine mögliche Fallentscheidung beim Programmieren in „Scratch“ siehst du hier:



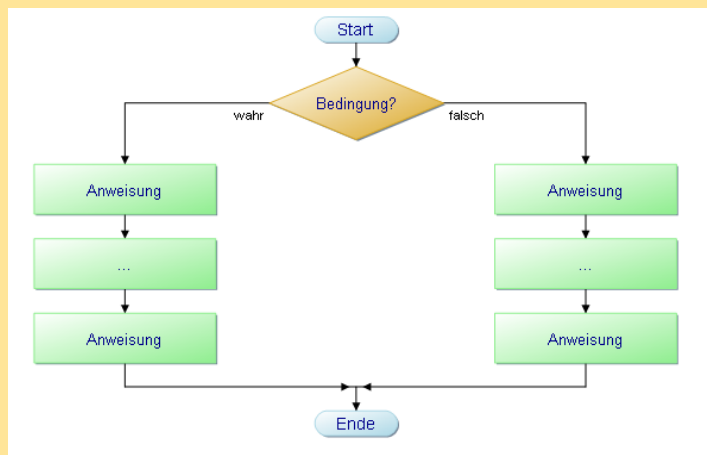
Alternative

Skizziere dazu ein Flussdiagramm.

Mögliche Lösung:

Es können im Flussdiagramm eine oder mehrere Anweisungen (Sequenz, wie im Beispiel) vorhanden sein.

Bepunktung:
Start, Anweisung(en), Bedingung, Ende, wahr, falsch je richtiger Verwendung 0,5 Punkte
Verbindungspfeile 1 Punkt



8/9

2.* Eine mögliche Fallentscheidung beim Programmieren in „Scratch“ siehst du hier:



Skizziere dazu ein Flussdiagramm. Verwende dafür

- Verbindungspfeile (z.B. \longrightarrow)
- die Begriffe „wahr“ und „falsch“ sowie
- die Symbole



Alternative

4

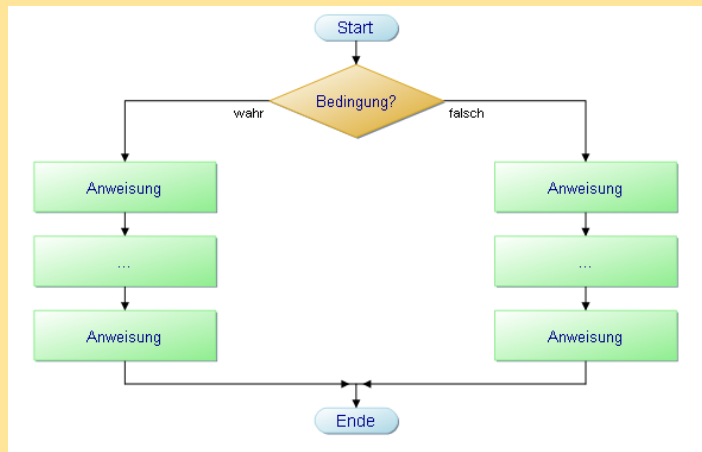
Mögliche Lösung:

Es können im Flussdiagramm eine oder mehrere Anweisungen (Sequenz, wie im Beispiel) vorhanden sein.

Bepunktung:

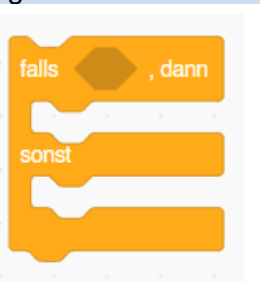
Start, Anweisung(en), Bedingung, Ende, wahr, falsch je richtiger Verwendung 0,5 Punkte

Verbindungspfeile 1 Punkt



8/9

3.* Ist diese Fallunterscheidung für das Programm zur Freizeitgestaltung geeignet? Begründe deine Entscheidung!



Alternative

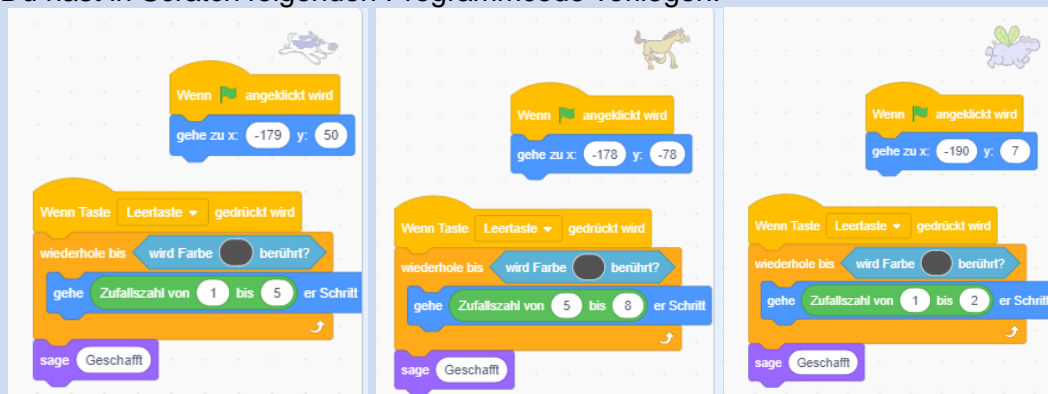
2

Mögliche Lösung:

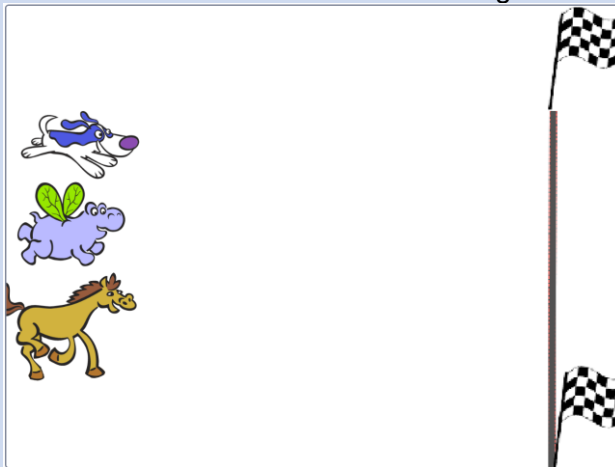
Die Fallentscheidung ist nicht geeignet. Es wird eine einseitige Fallunterscheidung benötigt. (1 Punkt)

Das Programm soll je nach angeklicktem Wettersymbol einen bestimmten Vorschlag machen. Wenn ein Wettersymbol nicht angeklickt wird, dann soll auch nichts passieren. (1 Punkt)

Du hast in Scratch folgenden Programmcode vorliegen.



Das Bühnenbild dazu ist wie unten abgebildet:



Erkläre ausführlich in eigenen Worten, wie dieses Programm funktioniert. Wer gewinnt höchstwahrscheinlich und wer verliert wohl dieses Rennen?

3,5

Lösung: (0,5 Pkt je Antwort)

Leertaste → alle rennen los

Unterschiedlich schnell, weil zufällig und unterschiedliche Zahlbereiche

Im Ziel sagen alle „Geschafft“

Leertaste immer wieder drücken

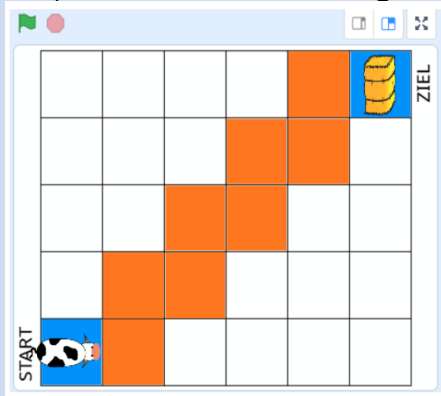
Pferd wird gewinnen, weil größter Zahlenbereich

Hippo verliert wohl, weil kleinster Zahlenbereich

Zurück / Neustart mit Fahne

Die Kuh soll auf dem orangenen Weg zum Heuballen gelangen. Hierfür wurde ein Programm erstellt, das allerdings einen Fehler hat. Zusätzlich könnte man das Programm vereinfachen.

- Verbessere den Fehler im vorhandenen Programm.
- Vereinfache das Programm und notiere dein Ergebnis als Pseudocode.



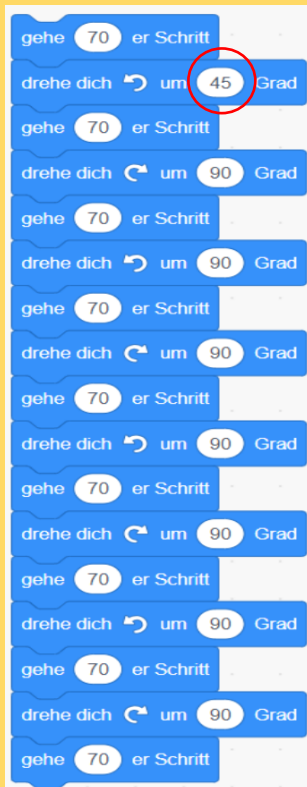
```

gehe 70 er Schritt
drehe dich um 45 Grad
gehe 70 er Schritt
drehe dich um 90 Grad
gehe 70 er Schritt
drehe dich um 90 Grad
gehe 70 er Schritt
drehe dich um 90 Grad
gehe 70 er Schritt
drehe dich um 90 Grad
gehe 70 er Schritt
drehe dich um 90 Grad
gehe 70 er Schritt
drehe dich um 90 Grad
gehe 70 er Schritt
drehe dich um 90 Grad
gehe 70 er Schritt

```

Lösung:

a) Es müssen 90 Grad sein. 1P.



b)

Wiederhole 4x

Gehe 70er Schritt

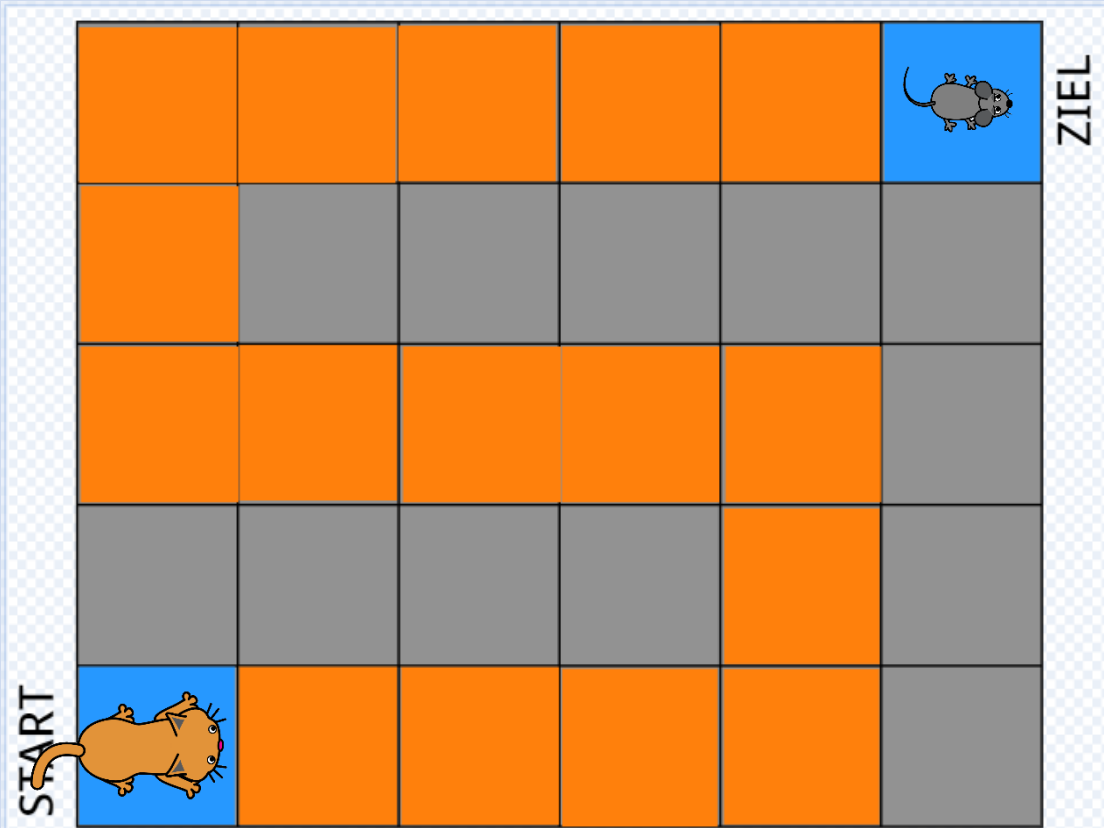
Drehe nach links um 90 Grad

Gehe 70er Schritt

Drehe nach recht um 90 Grad

Gehe 70er Schritt

- 1P. für das Erkennen von „Wiederhole 4x“**
- 1P. für die richtige Abfolge innerhalb der Wiederholung**
- 1P. für den letzten 70er Schritt nach der Wiederholung**



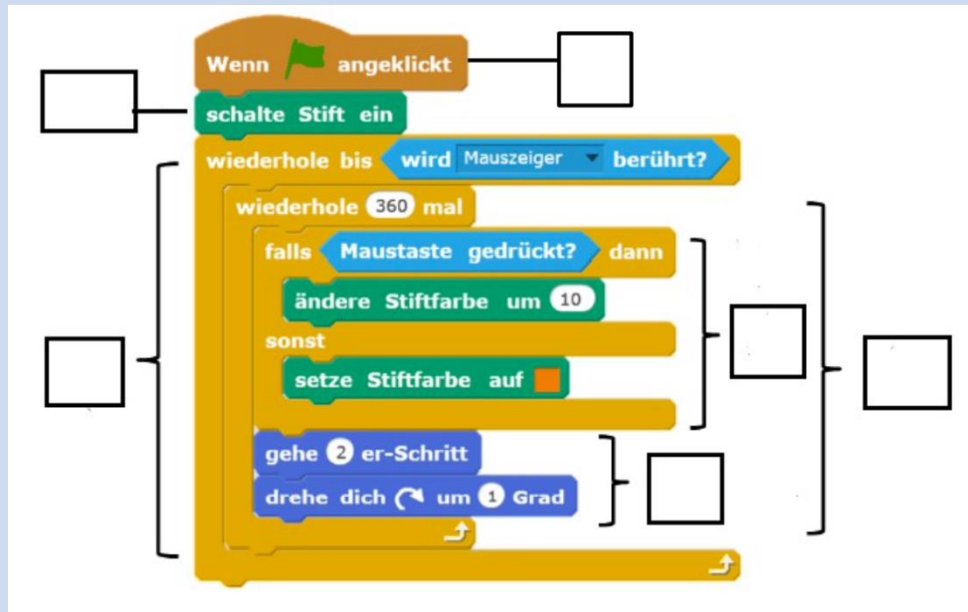
Die Katze soll auf dem vorgegebenen Weg zur Maus laufen. Erstelle einen Programmcode, der die Katze ans Ziel führt. Mehrfachnennungen sind möglich. Verwende nur die benötigten Befehle.

1		2		3	
4		5		6	
7		8		9	
10		11		12	
13		14		15	

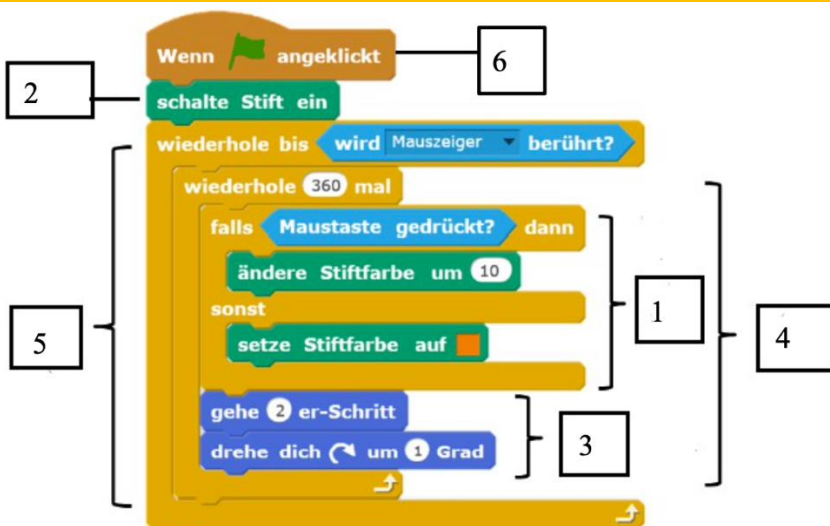
| | | ...

Ordne die Begriffe zu, indem du die Zahlen einsetzt.

- 1 **Bedingung**
- 2 Anweisung
- 3 Sequenz
- 4 Wiederholung mit fester Anzahl
- 5 Wiederholung mit Bedingung
- 6 Startanweisung



3



Lösung

Du siehst hier drei unterschiedliche Programmcodes für eine Digitaluhr:

A

B

C

```

Wenn Flagge angeklickt wird
  zeige Variable Stunde
  zeige Variable Minute
  zeige Variable Sekunde
  wiederhole fortlaufend
    setze Stunde auf Stunde im Moment
    setze Minute auf Minute im Moment
    setze Sekunde auf Sekunde im Moment
  
```

```

Wenn diese Figur angeklickt wird
  wiederhole fortlaufend
    zeige Variable Stunde
    setze Stunde auf Stunde im Moment
    zeige Variable Minute
    setze Minute auf Minute im Moment
    zeige Variable Sekunde
    setze Sekunde auf Sekunde im Moment
  
```

```

Wenn diese Figur angeklickt wird
  zeige Variable Stunde
  zeige Variable Minute
  zeige Variable Sekunde
  setze Vergleichsvariable auf Sekunde im Moment + 10
  wiederhole bis Vergleichsvariable = Sekunde
    setze Stunde auf Stunde im Moment
    setze Minute auf Minute im Moment
    setze Sekunde auf Sekunde im Moment
  verstecke Variable Stunde
  verstecke Variable Minute
  verstecke Variable Sekunde
  
```

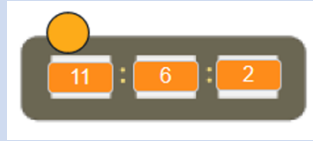
Kreuze in der Tabelle an, welche Aussagen jeweils zutreffen, es sind auch mehrere Kreuze pro Zeile möglich:

A B C Keins davon

- Beim Klicken auf das Startereignis läuft die Uhr los.**
- Die Anzeige verschwindet nach 45 Minuten.**
- Nach 10 Sekunden beginnt die Uhr zu laufen.**
- Um die Uhr zu starten, muss ich auf die Figur klicken.**
- Beim Klick auf die Fahne startet die Uhr.**



Tom hat eine Digitaluhr in Scratch programmiert. Diese zeigt die aktuelle Stunde, Minute und Sekunde an. Die Uhrzeit aktualisiert sich immer. Die Uhr startet, wenn du den orangenen Einschalter drückst.



Folgenden Programmcode hat Tom dazu geschrieben:



Allerdings, wenn Tom den orangenen Einschalter drückt passiert nichts.

Was hat er falsch gemacht?

Markiere den Fehler im Programmcode von Scratch.

Begründe, warum hier der Fehler ist?

Lösung:



Begründung:

0,5 Pkt.

0,5 Pkt.

Tom startet das Programm mit dem Fähnchen. Tom muss das Starterereignis auf den orangen Button ändern.

8.1

AF



Du willst in Scratch eine Digitaluhr programmieren. Folgende Anforderungen soll dein Programm erfüllen:

- Die Uhr soll über einen „Startknopf“ gestartet werden.
- Die Uhr zeigt die Stunde, Minuten und Sekunden an.
- Die Uhr soll die aktuelle Zeit anzeigen.

Aufgabe: Entwerfe für dieses Programm ein Flussdiagramm!

X

Y

XY



8.1

Du willst in Scratch eine Digitaluhr programmieren. Folgende Anforderungen soll dein Programm erfüllen:

- Die Uhr soll über einen „Startknopf“ gestartet werden.
- Die Uhr zeigt die Stunde, Minuten und Sekunden an.
- Die Uhr soll die aktuelle Zeit anzeigen.

Aufgabe: Entwerfe für dieses Programm ein Flussdiagramm!



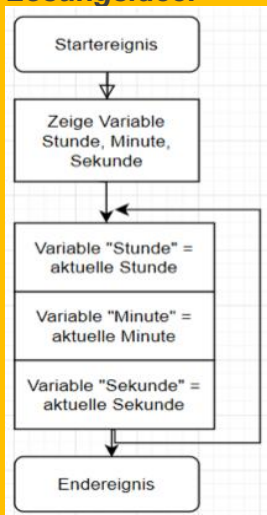
X

Y

XY



Lösungsidee:



Finde die Fehler!

„Frank the Tank“ soll mit Hilfe der Pfeiltasten steuerbar sein. Sobald er den Foodtruck berührt, soll er „Waffeln!“ sagen und sein Kostüm wechseln.



Folgender Code wurde programmiert:

```

Wenn angeklickt wird
  gehe zu x: -130 y: 7
  wechsele zu Kostüm frank-a
  wiederhole 10 mal
    falls wird Mauszeiger berührt? dann
      sage WAFFELN!!! für 2 Sekunden
      wechsele zu Kostüm frank-d

Wenn Taste Pfeil nach oben gedrückt wird
  setze Drehtyp auf links-rechts
  setze Richtung auf 0 Grad
  gehe 10 er Schritt
  pralle vom Rand ab

Wenn Taste Pfeil nach links gedrückt wird
  setze Drehtyp auf links-rechts
  setze Richtung auf 90 Grad
  gehe 10 er Schritt
  pralle vom Rand ab

Wenn Taste Pfeil nach unten gedrückt wird
  setze Drehtyp auf links-rechts
  setze Richtung auf 180 Grad
  gehe 10 er Schritt
  pralle vom Rand ab

Wenn Taste Pfeil nach rechts gedrückt wird
  setze Drehtyp auf links-rechts
  setze Richtung auf -90 Grad
  gehe 10 er Schritt
  pralle vom Rand ab
    
```

Kreise die Fehler ein!

3

Fehler: wiederhole 10 mal = wiederhole fortlaufend; wird Mauszeiger berührt? = wird Foodtruck berührt?; Wenn Taste „Pfeiltaste nach links“ gedrückt wird (doppelt) = Wenn Taste „Pfeiltaste nach rechts“ gedrückt wird

Erhöhtes Anforderungsniveau (3) möglich:
Fehler markieren und berichtigen.

Ordne den möglichen Variableninhalten einen Variablentyp zu. Verbinde diese mit einem Pfeil:

Variableninhalt:

- +365
- wahr
- „Apfel“
- 45-23
- falsch
- „Hallo Welt!“

Variablentyp:

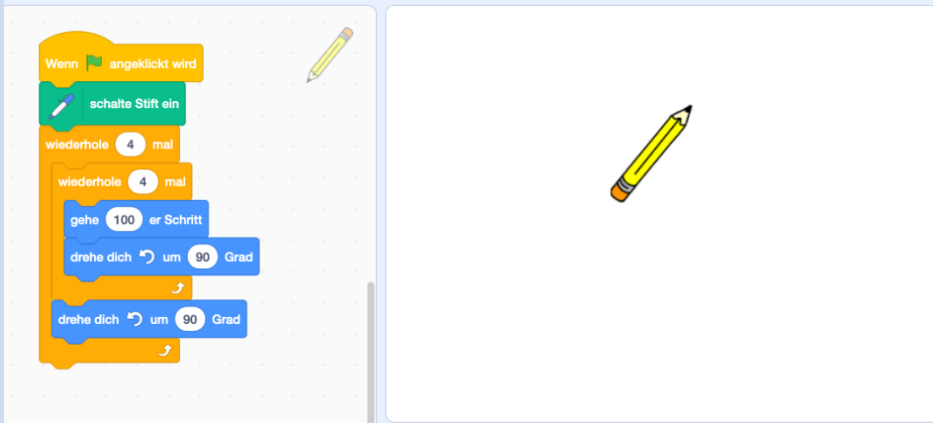
- Integer
- Boolean
- String

3

Integer (+365; 45-23); Boolean (wahr; falsch); String („Apfel“; „Hallo Welt!“)

Du erhältst folgendes Programm in der Programmiersprache „Scratch“. Beschreibe den Ablauf der Programmierschritte nach dem Start des Programms in der

5



richtigen Reihenfolge in eigenen Worten.

5

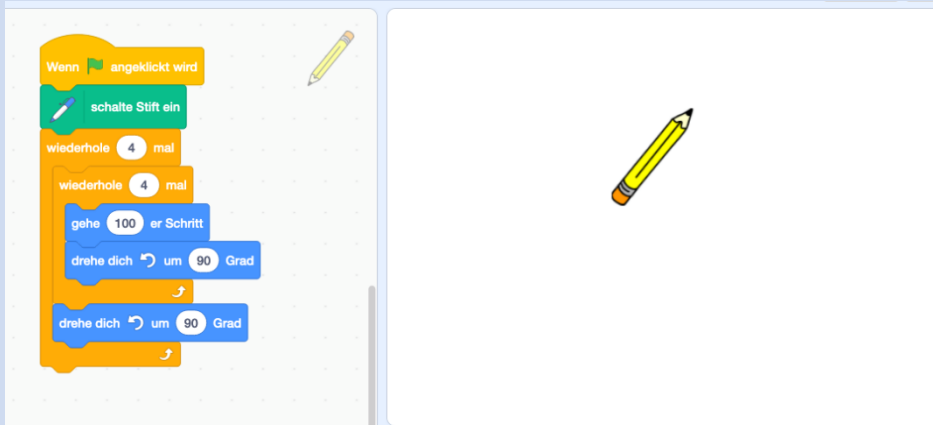
Mögliche Lösung:

Bepunktung:

je richtigem Programmierschritt 1 Punkt; für die richtige Reihenfolge 1 Punkt

- **Der Stift wird eingeschaltet.**
- **Die verschachtelte (innere) Wiederholung zeichnet ein Quadrat (mit 100 px Kantenlänge).**
- **Die Zeichenrichtung wird um 90 Grad gegen den Uhrzeigersinn gedreht.**
- **Die äußere Wiederholung wiederholt diesen Vorgang noch drei weitere Male (insgesamt 4mal).**

Du erhältst folgendes Programm in der Programmiersprache „Scratch“.
Sortiere den Ablauf der Programmierschritte nach dem Start des Programms in der



richtigen
Reihenfolge
mit den
Zahlen 1 bis
4.

Die Zeichenrichtung wird um 90 Grad gegen den Uhrzeigersinn gedreht.

Der Stift wird eingeschaltet.

Die äußere Wiederholung wiederholt diesen Vorgang noch drei weitere Male (insgesamt 4mal).

Die verschachtelte (innere) Wiederholung zeichnet ein Quadrat (mit 100 px Kantenlänge).

Mögliche Lösung:

Bepunktung:
je richtiger Zahl 0,5 Punkt

3 **Die Zeichenrichtung wird um 90 Grad gegen den Uhrzeigersinn gedreht.**

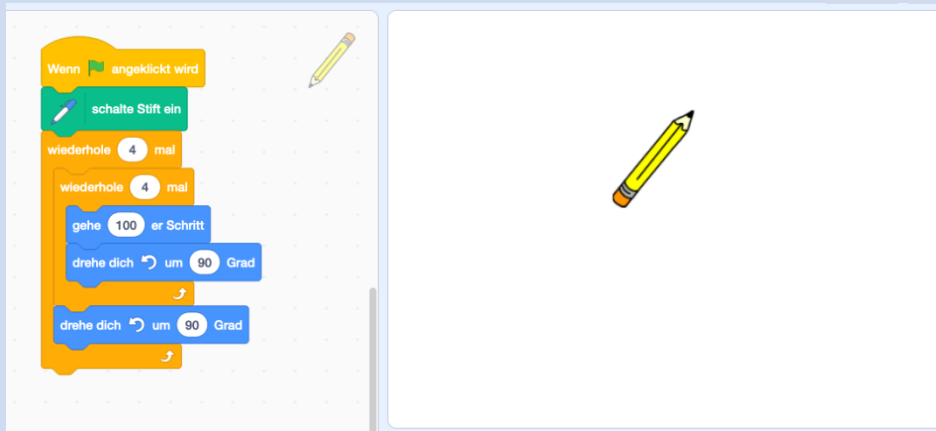
1 **Der Stift wird eingeschaltet.**

4 **Die äußere Wiederholung wiederholt diesen Vorgang noch drei weitere Male (insgesamt 4mal).**

2 **Die verschachtelte (innere) Wiederholung zeichnet ein Quadrat (mit 100 px Kantenlänge).**

Du erhältst folgendes Programm in der Programmiersprache „Scratch“. Skizziere die nach dem Programmstart vom Stift gezeichnete Figur.

3



3

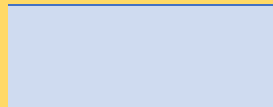
Mögliche Lösung:

Bepunktung:

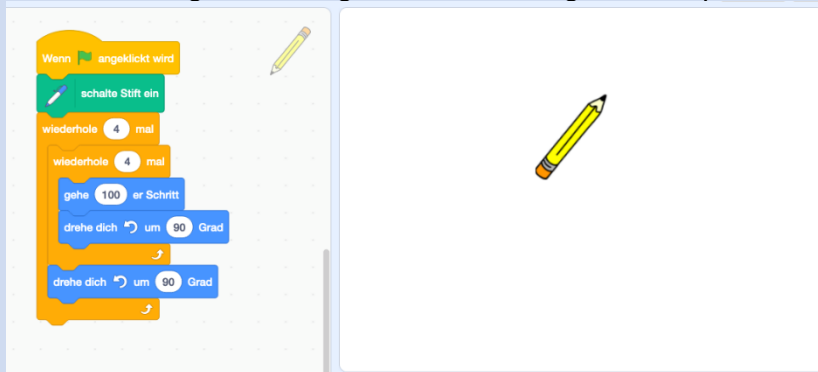
SuS erkennt Quadrat, 1 Punkt

SuS erkennt die 4fache Wiederholung, 1 Punkt

SuS erkennt die Anordnung der vier Quadrate, 1 Punkt



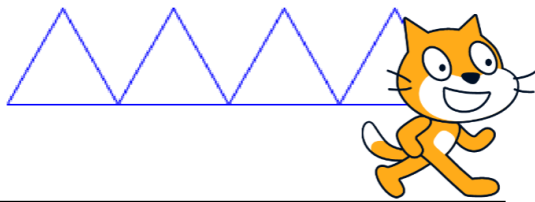
Du erhältst folgendes Programm in der Programmiersprache „Scratch“.



Welche von den vier Figuren zeichnet der Stift? Kreuze die richtige Figur an!

Mögliche Lösung:
Lösung 2 ist richtig, 1 Punkt

Scratch soll dieses Bild mit Dreiecken malen. Programmiert wurde die hier rechts stehende Sequenz. Leider entsteht beim Anklicken der grünen Fahne nicht das gewünschte Bild. Kannst du alleine durch Betrachten der Sequenz sagen, warum nicht?



Wähle den passenden Befehlsblock und markiere die Stelle in der Sequenz, wo dieser eingefügt werden muss.

- gehe zu Zufallsposition
- schalte Stift aus
- gehe 60 er Schritt

```

Wenn grüne Fahne angeklickt wird
  gehe zu x: -132 y: -41
  setze Richtung auf 90 Grad
  schalte Stift ein
  wiederhole 4 mal
    wiederhole 3 mal
      gehe 60 er Schritt
      drehe dich um 120 Grad
    
```

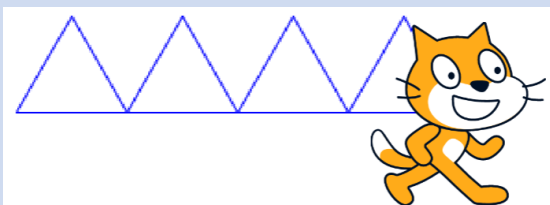
2

Lösung:

Befehlsblock: „Gehe 60er Schritt“

Unterhalb der 3-er-Schleife

Scratch soll dieses Bild mit Dreiecken malen. Programmiert wurde die hier rechts stehende Sequenz. Leider entsteht beim Anklicken der grünen Fahne nicht das gewünschte Bild. Kannst du alleine durch Betrachten der Sequenz sagen, warum nicht?



Nenne den fehlenden Befehlsblock und markiere die Stelle in der Sequenz, wo dieser eingefügt werden muss.

```

Wenn grüne Fahne angeklickt wird
  gehe zu x: -132 y: -41
  setze Richtung auf 90 Grad
  schalte Stift ein
  wiederhole 4 mal
    wiederhole 3 mal
      gehe 60 er Schritt
      drehe dich um 120 Grad
    
```

2

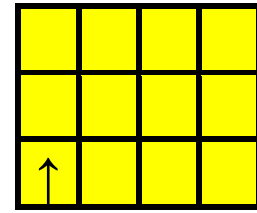
Lösung:

Befehlsblock: „Gehe 60er Schritt“ oder „Ändere x um 60“

Unterhalb der 3-er-Schleife

Auf einem Feld mit 12 Fliesen sollst du mit den Befehlen „Schritt“, „Drehe 90° rechts“ und „Drehe 90° links“ einmal alle Fliesen berühren.

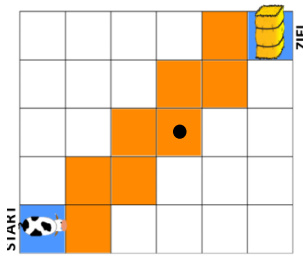
Schreibe den passenden Programm-Code auf. Starte unten links in Pfeilrichtung.



Lösung:

Für jedes berührte Feld einen Punkt. Bei doppelten oder extrem langer und umständlicher Programmierung gibt es Abzüge.

Franzi programmiert mit Scratch. Ihre Kuh Elsa soll automatisch vom Start erstmal bis zur Fliese mit dem Punkt laufen. Was hat Franzi falsch gemacht? Verbessere ihren Programm-Code so, dass es funktioniert.



```

Wenn [ ] angeklickt wird
  gehe 25 er Schritt
  drehe dich um 90 Grad
  gehe 70 er Schritt
  drehe dich um 15 Grad
  gehe 700 er Schritt
  drehe dich um 90 Grad
  gehe 70 er Schritt
  drehe dich um 900 Grad
  gehe 70 er Schritt
  
```

```

Wenn [ ] angeklickt wird
  gehe 25 er Schritt
  drehe dich um 90 Grad
  gehe 70 er Schritt
  drehe dich um 15 Grad
  gehe 700 er Schritt
  drehe dich um 90 Grad
  gehe 70 er Schritt
  drehe dich um 900 Grad
  gehe 70 er Schritt
  
```


Szenario

8/9

Szenario: Ein Ball soll in einer rechteckigen Bewegung um den Bildschirm rollen. Damit eine Rollbewegung entsteht, soll der Ball sich alle paar Schritte drehen.



8/9

1. Das folgende Programm funktioniert nicht. Beschreibe in zwei oder drei Sätzen, woran dies liegt.

```

Wenn  angeklickt
  gehe zu x: -214 y: -155
  wiederhole 2 mal
    setze Richtung auf 90 Grad
    wiederhole bis wird Rand berührt?
      ändere x um 5
      drehe dich um 15 Grad
    ändere x um -10
    setze Richtung auf 0 Grad
    wiederhole bis wird Rand berührt?
      ändere y um 5
      drehe dich um 15 Grad
    ändere y um -5
  
```

1

Lösung:

Die Schleife „Wiederhole 2 mal“ umschließt die Bewegung nach rechts und nach oben. Da diese wiederholt wird, wenn der Ball oben rechts in der Ecke ist, geht der Ball von dort nicht nach links und dann nach unten, sondern aus der rechten oberen Ecke geht er nochmal nach rechts und nach oben.

Bepunktung: 1 Punkt für die Nennung Wiederholungsschleife, 1 weiteren Punkt für die Erkenntnis, dass der Ball nicht nach links und nach unten rollt, sondern weiter nach rechts und nach oben.

8/9

2. Die Steuerung ist noch unvollständig. Notiere die zwei Anweisungen, wie du die „Pfeil nach unten“-Taste programmieren würdest.

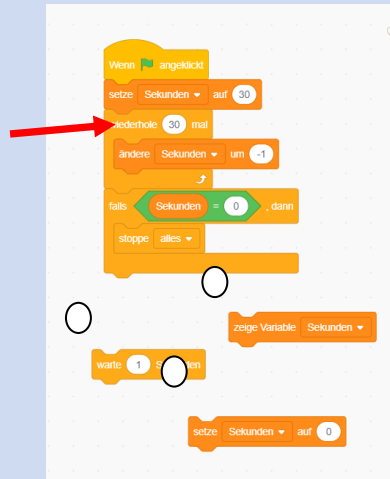
The image shows four Scratch code blocks arranged in a 2x2 grid. Each block starts with a yellow 'Wenn Taste gedrückt' (When key pressed) block. The top-left block is for the 'Pfeil nach oben' (Up arrow) key, followed by a blue 'ändere y um 10' (Change y by 10) block and a blue 'drehe dich um 15 Grad' (Turn 15 degrees) block. The top-right block is for the 'Pfeil nach links' (Left arrow) key, followed by a blue 'ändere x um 10' (Change x by 10) block and a blue 'drehe dich um 15 Grad' (Turn 15 degrees) block. The bottom-left block is for the 'Pfeil nach unten' (Down arrow) key, followed by a blue 'ändere y um 10' (Change y by 10) block and a blue 'drehe dich um 15 Grad' (Turn 15 degrees) block. The bottom-right block is for the 'Pfeil nach rechts' (Right arrow) key, followed by a blue 'ändere x um 10' (Change x by 10) block and a blue 'drehe dich um 15 Grad' (Turn 15 degrees) block.

1

Lösung: „ändere y um -10“ und „drehe dich (links oder rechts, beides kann erlaubt werden) um 15 Grad“

Bepunktung: 1 Punkt für „ändere y“, 1 Punkt für „um -10“, 1 Punkt für „drehe dich“

Max möchte ein Spiel programmieren. Dazu „bastelt“ er sich einen Timer, der insgesamt 30 Sekunden dauern soll. Jetzt ist er schon fast fertig, aber welche der drei Anweisungen muss er jetzt noch **hier** einfügen. (Kreuze an)

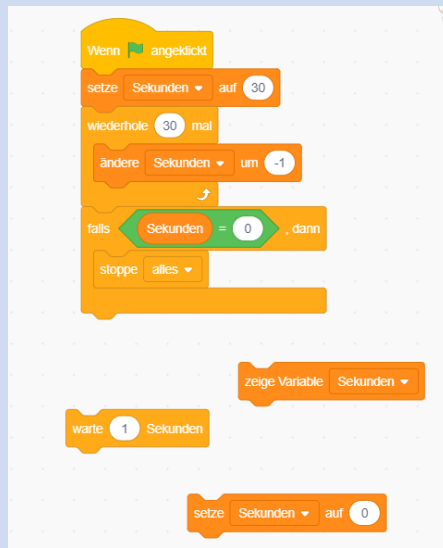


Lösung:   warte 1 Sekunden

Auf die richtig angekreuzte Anweisung gibt es einen Punkt. Bei mehr als einer angekreuzten Antworten gibt es keinen Punkt.

8

Max möchte ein Spiel programmieren. Dazu „bastelt“ er sich einen Timer, der insgesamt 30 Sekunden dauern soll. Jetzt ist er schon fast fertig, aber welche der drei Anweisungen muss er jetzt noch einfügen. Markiere auch die richtige Position.



1

1

2

Lösung:



**Auf die richtig gewählte Anweisung gibt es einen Punkt.
Auf die richtige Position der Anweisung gibt es einen weiteren Punkt.**

Max möchte ein Spiel programmieren. Dazu „bastelt“ er sich einen Timer, der insgesamt 30 Sekunden dauern soll. Beim Ausprobieren merkt er, dass der Timer nicht geht. Welchen Fehler hat er gemacht? Erkläre und hilf ihm dabei, den Timer richtig zu programmieren.

```

Wenn [ ] angeklickt
  setze Sekunden auf 30
  wiederhole 30 mal
    ändere Sekunden um -1
  falls Sekunden = 0, dann
    stoppe alles
  
```

1

1

1

3

Lösung:

```

Wenn [ ] angeklickt
  setze Sekunden auf 30
  erhöhe Sekunden um 1
  warte 1 Sekunden
  ändere Sekunden um -1
  falls Sekunden = 0, dann
    stoppe alles
  
```

Eine Anweisung vergessen. (1 Punkt)

Die richtige Anweisung gefunden. (1 Punkt)

Auf die richtige Position der Anweisung gibt es einen weiteren Punkt. (1Punkt)

Szenario

8/9

Szenario: In einem Spiel soll ein Ball mit den Pfeiltasten in alle vier Richtungen auf dem Bildschirm gesteuert werden.

8/9

1. Im folgenden Programmcode steckt ein Fehler.
- Welche Anweisung ist falsch? Umkreise diese.
 - Wie müsste sie richtig heißen? Schreibe deine Verbesserung daneben.

The image shows four Scratch code blocks arranged in a 2x2 grid. Each block starts with 'Wenn Taste' followed by a dropdown menu and 'gedrückt'.
Top-left: 'Pfeil nach oben' dropdown. Contains two blue blocks: 'ändere y um 10' and 'drehe dich um 15 Grad'.
Top-right: 'Pfeil nach links' dropdown. Contains two blue blocks: 'ändere x um 10' and 'drehe dich um 15 Grad'.
Bottom-left: 'Pfeil nach unten' dropdown. The block is empty.
Bottom-right: 'Pfeil nach rechts' dropdown. Contains two blue blocks: 'ändere x um 10' and 'drehe dich um 15 Grad'.

1

Lösung:

- Die Anweisung „ändere x um 10“ unter der Anweisung „Pfeil nach links“ ist falsch.
- Die Anweisung müsste heißen: „ändere x um -10“.

Bepunktung: 1 Punkt für die Nennung der richtigen Anweisung, 1 Punkt für die korrekte Verbesserung der Anweisung.

2. Die Steuerung ist noch unvollständig. Notiere die zwei Anweisungen, wie du die „Pfeil nach unten“-Taste programmieren würdest.

The image shows four Scratch code blocks arranged in a 2x2 grid. Each block is a 'When Key Pressed' event block. The top-left block is for the 'Up' arrow key, with 'change y by 10' and 'rotate 15 degrees' blocks. The top-right block is for the 'Left' arrow key, with 'change x by 10' and 'rotate 15 degrees' blocks. The bottom-left block is for the 'Down' arrow key, which is currently empty. The bottom-right block is for the 'Right' arrow key, with 'change x by 10' and 'rotate 15 degrees' blocks.

1

Lösung: „ändere y um -10“ und „drehe dich (links oder rechts, beides kann erlaubt werden) um 15 Grad“

Bepunktung: 1 Punkt für „ändere y“, 1 Punkt für „um -10“, 1 Punkt für „drehe dich“

Szenario

8/9

Szenario: Du hast im Internet ein Scratch-Programm gefunden, das zwei Zufallszahlen addieren kann. Leider funktioniert es nicht.

8/9

1. Im folgenden Programmcode stecken zwei Fehler.
- Eine Anweisung ist falsch. Umkreise diese.
 - Was muss an dieser Anweisung verbessert werden? Schreibe die Verbesserung daneben.
 - Eine andere Anweisung ist an der falschen Stelle. Kennzeichne mit einem Pfeil, zwischen welche zwei Anweisungen sie hingehört, damit das Programm funktioniert.

```

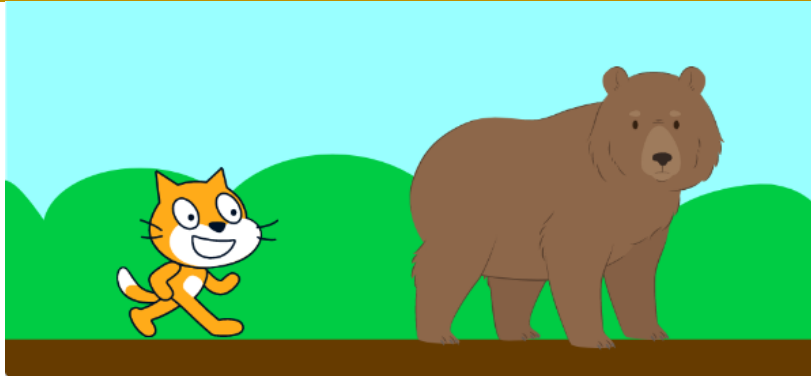
Wenn angeklickt
  setze Zahl 1 auf 0
  setze Zahl 2 auf 0
  sage Ich addiere für dich zwei Zahlen! für 2 Sekunden
  setze Zahl 1 auf Zufallszahl von 1 bis 10
  sage Zahl 1 für 2 Sekunden
  warte 1 Sekunden
  sage Zahl 2 für 2 Sekunden
  warte 1 Sekunden
  setze Zahl 2 auf Zufallszahl von 1 bis 10
  setze Ergebnis auf Zahl 1 + Zahl 2
  sage verbinde Das Ergebnis ist und Ausgabe für 2 Sekunden
    
```

1

Lösung:

- Die Variable „Ausgabe“ in der letzten Anweisung ist falsch.
- Die Variable müsste „Ergebnis“ heißen.
- „Setze Zahl2 auf Zufallszahl ...“ ist an der falschen Stelle. Diese muss vor die Anweisung „Sage Zahl2 für 2 Sekunden“.

Bepunktung: 1 Punkt für jede Teilaufgabe.



Die Idee ist folgende: Die Katze soll sich, sobald sie den Bären berührt, umdrehen und in die entgegengesetzte Richtung laufen. Sobald sie den Rand berührt, kehrt ihr Mut zurück und sie kehrt wieder in Richtung Bären um.

```

Wenn [ ] angeklickt wird
wiederhole 10 mal
  gehe 10 er Schritt
  falls [ ] wird Rand berührt? , dann
    setze Richtung auf -90 Grad
  falls [ ] wird Baer berührt? , dann
    setze Richtung auf 90 Grad
  
```

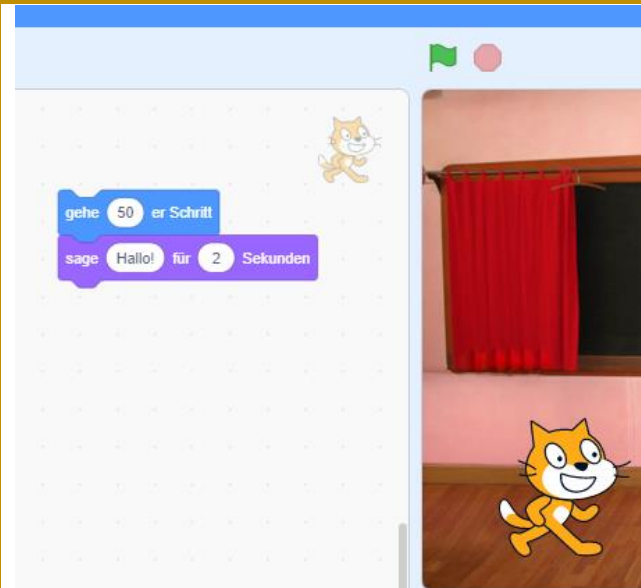
1. Leider funktioniert der Code nicht wie geplant. Finde die Fehler und schreibe die Korrektur direkt rechts daneben.
2. Der 10er-Schritt wird momentan 10-mal wiederholt, dann muss die Flagge wieder angeklickt werden. Ändere das Programm so, dass eine Endlosbewegung daraus wird.
3. Sobald die Leertaste gedrückt wird, soll der Bär unsichtbar werden. Mit welchem Befehl geht das? _____
4. Mit welchem Befehl kann man den Bären anschließend wieder sichtbar machen? _____
5. Ein Freund möchte gerne, dass die Bären-Berührungen gezählt werden. Notiere auf ein extra Blatt, wie dies realisiert werden könnte. Was muss bei Neustart des Programmes beachtet werden?

Lösung:

1. Die Gradzahlen sind vertauscht, d.h. die erste Richtungsangabe muss 90 Grad lauten, die zweite -90. (je 1 Punkt)
2. Aus „wiederhole 10 mal“ soll „wiederhole fortlaufend“ gemacht werden. (1 Punkt)
3. „verstecke dich“ (1 Punkt) 4. „zeige dich“ (1 Punkt)
5. Als Lösungsbestandteil sollten Variablen genannt werden, die bei jeder Berührung um eine hochgezählt werden. (1 Punkt für Nennung Stichwort Variable; 1 Punkt dafür, dass sie jeweils um 1 hochgezählt werden bei einer Bärenberührung); 1 Punkt, dass bei Neustart des Programmes die Variable wieder auf 0 gesetzt werden muss.

8

CI



The image shows a Scratch workspace. On the left is the code editor with two blocks: a blue 'move' block set to 50 steps and a purple 'say' block set to 'Hallo!' for 2 seconds. On the right is the stage view showing a red curtain and a cat sprite on a wooden floor. At the top of the stage, there is a green flag icon and a red circle icon.

Warum führt die Figur die programmierten Anweisungen nicht aus, wenn du auf die grüne Flagge klickst?

1

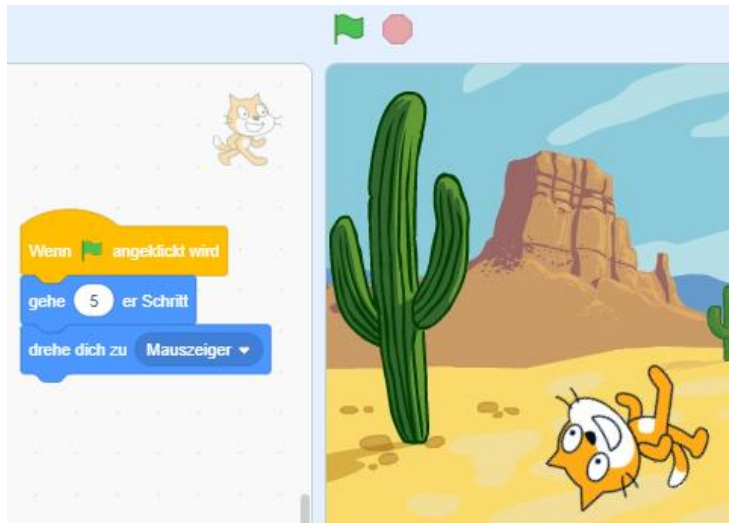
Lösung:

Es fehlt das auslösende Ereignis „Wenn grüne Flagge angeklickt wird“.

Die Figur soll sich beim Anklicken der grünen Flagge fortlaufend dem Mauszeiger folgen.

Du hast dazu diesen Algorithmus programmiert, stellst jedoch fest, dass er nicht funktioniert.

Beschreibe den Fehler und eine mögliche Lösung!



Lösung:

Das Programm führt den Algorithmus genau **einmal** aus. Es fehlt die Anweisung „Wiederhole fortlaufend“.

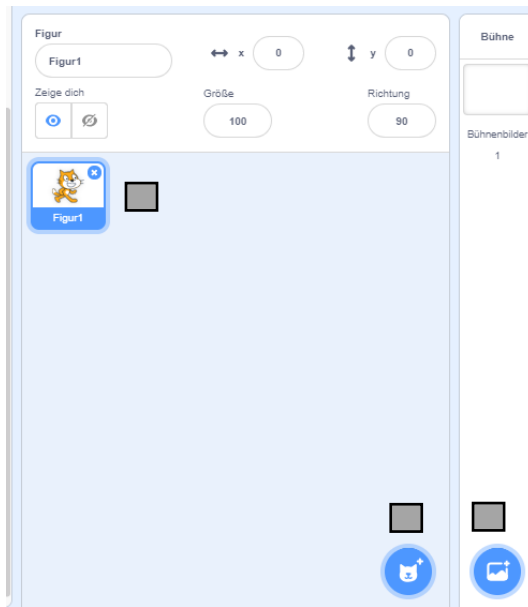
Fehlerbeschreibung: 1 Punkt

Lösung: 1 Punkt

8

ZO

Du willst eine neue Figur hinzufügen. Welche Schaltfläche nutzt du hierfür? Kreuze an!



1

Lösung:



9

FT

Erkläre den Fachbegriff Algorithmus.

2

Lösung:

Ein Algorithmus ist eine eindeutige **Handlungsvorschrift** zur Lösung eines Problems. Algorithmen bestehen aus vielen, wohldefinierten **Einzelschritten**. (je 1 Punkt für die hervorgehobenen Begriffe)

FT

Welche Regeln gelten in einem Algorithmus?

3

Lösung:

- Algorithmus besteht aus EINZELNEN endlichen Schritten:
- Im Algorithmus können Kontrollstrukturen verschachtelt werden.
- Ein Algorithmus hat genau einen Vorgänger und einen Nachfolger (nicht 1. und letzter)

FT

Was ist ein Klassendiagramm/Klassenbaum?

2

Lösung:

- Alle Klassen werden in Beziehung zueinander dargestellt.

FT

Was ist ein Attribut?
Was ist ein Attributwert?

2

Lösung:

- Eigenschaften einer Instanz (Linienstärke).
- Wert eines Attributes (3 mm).

FT

Was ist eine Klasse?

3

Lösung:

- (Bauplan)
- Beschreibt alle Eigenschaften und Ausführungen die möglich sind.
- Geometrische Grundkörper (Linie, Viereck, Ellipse, ...).

FT

Warum hilft es, wenn man sich zunächst über die „Objektorientierung“ eines Programms Gedanken macht?

2

Lösung:

- Die Grafische Oberfläche ändert sich, aber Objektorientierte Struktur von Vektorprogrammen bleibt gleich.
- Außerdem lässt sich das Wissen auf viele anderen Vektorprogramme übertragen.

FT

Was ist ein Attribut?
Was ist ein Attributwert?

2

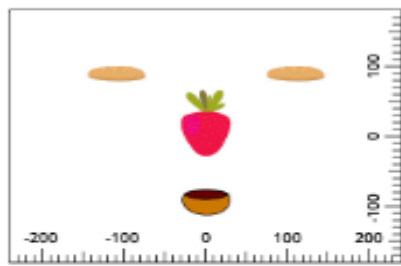
Lösung:

- Eigenschaften einer Instanz (Linienstärke).
- Wert eines Attributes (3 mm).

8/9

In den folgenden Aufgaben soll Scratch diese "Gesichter" zeichnen. Leider stimmt aber das Skript so nicht. Wo liegt der Fehler?

1.



Verwendete Kostüme:



```

Wenn angeklickt wird
  gehe zu x: -110 y: 90
  wechsele zu Kostüm: Brot
  hinterlasse Abdruck
  gehe zu x: 110 y: 90
  hinterlasse Abdruck
  gehe zu x: 0 y: 20
  wechsele zu Kostüm: Erdbeere
  hinterlasse Abdruck
  gehe zu x: -90 y: 0
  wechsele zu Kostüm: Schüssel
  hinterlasse Abdruck
    
```

2

Lösung: Beim letzten „Gehe-Zu“-Befehl wurden x und y verwechselt.

Hier das korrigierte Skript:

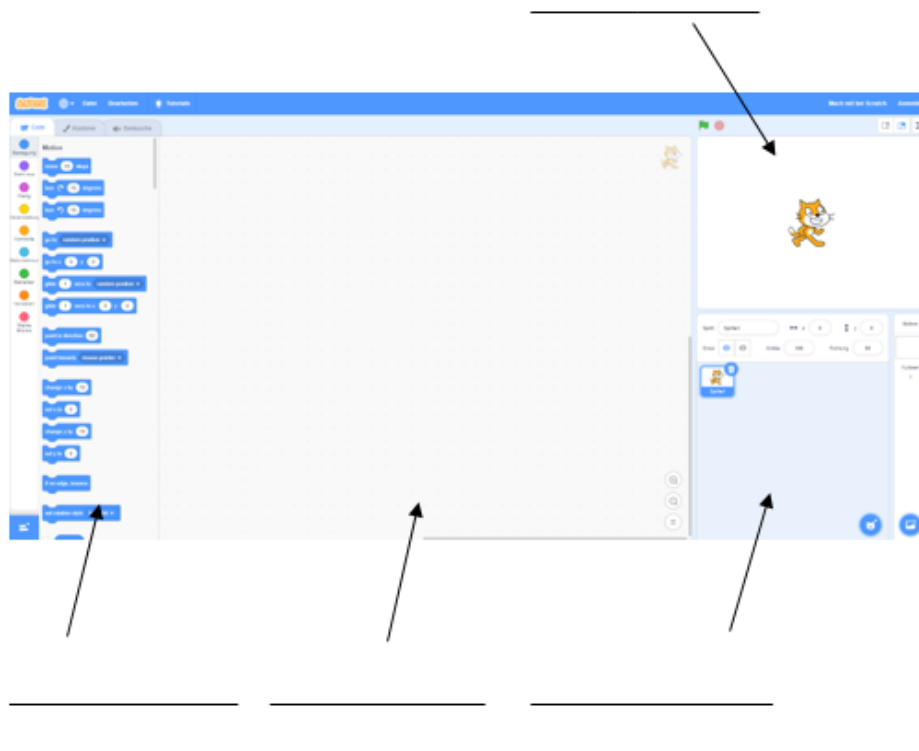
```

Wenn angeklickt wird
  gehe zu x: -110 y: 90
  wechsele zu Kostüm: Brot
  hinterlasse Abdruck
  gehe zu x: 110 y: 90
  hinterlasse Abdruck
  gehe zu x: 0 y: 20
  wechsele zu Kostüm: Erdbeere
  hinterlasse Abdruck
  gehe zu x: -90 y: 0
  wechsele zu Kostüm: Schüssel
  hinterlasse Abdruck
    
```

8/9

Ordne folgende Begriffe zu:

Objektliste, Blockpalette, Programmierbereich, Bühne



2

Lösung:

8/9

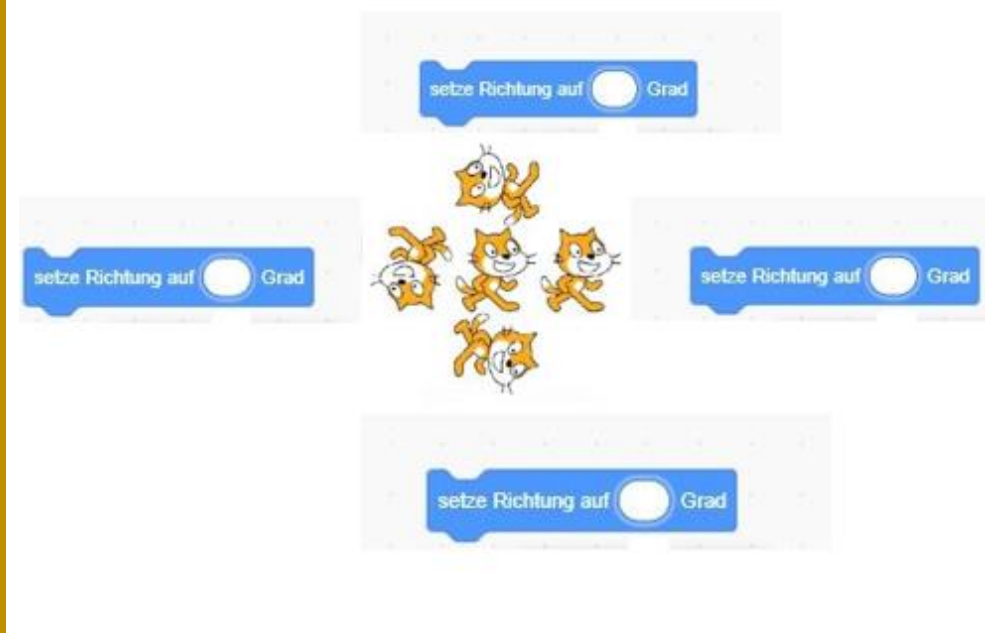
Wie kannst du die Katze Scratch dazu bringen, rückwärts zu laufen? Benutze nur den folgenden Befehl:



1

Lösung: Eintragen einer negativen Schrittzahl

Bestimme die jeweilige Richtung durch Einsetzen der richtigen Gradzahl:



The image shows a Scratch script with four 'setze Richtung auf' (set direction to) blocks. Each block has a circular input field for a degree value and the word 'Grad' next to it. The blocks are arranged around a central group of five orange cat sprites. The top block is positioned above the sprites, the left block is to the left, the right block is to the right, and the bottom block is below the sprites.

Lösung:

- 0 Grad = oben
- 90 Grad = rechts
- 180 Grad = unten
- 90 Grad = links

Zeichne zu den folgenden Skripten ein, wo sich Scratch nach Ausführen des jeweiligen Skripts befindet. Du kannst Scratch dabei durch ein Strichmännchen darstellen, aber Standort und Richtung müssen erkennbar sein! Bei jeder Teilaufgabe kannst du davon ausgehen, dass Scratch zu Beginn in der Mitte der Bühne steht und nach rechts (Richtung 90 Grad) schaut.

a)

b)

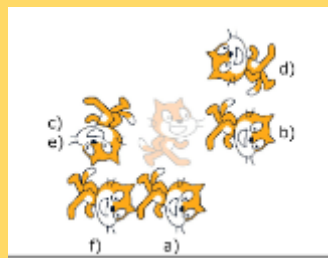
c)

d)

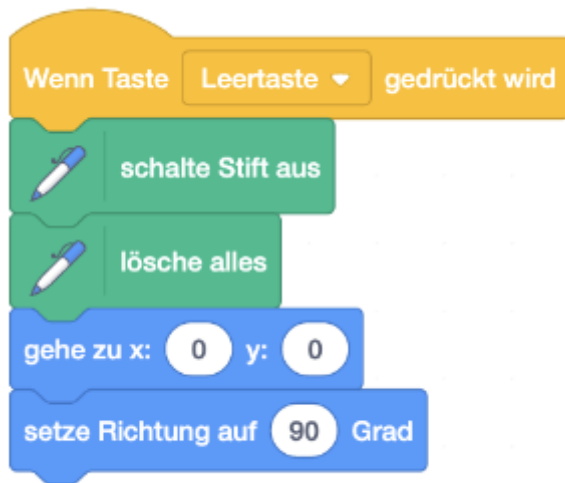
e)

f)

Lösung:



Ist die Reihenfolge in diesem „Aufräum-Skript“ egal?
Wenn nicht, worauf musst du achten?



2

Lösung:

Das Gehen muss nach dem Anheben des Stiftes oder vor dem Wegwischen der Malspuren erfolgen, da sonst beim Gehen zur Mitte noch eine Malspur entsteht, die nicht weggewischt wird. Die Reihenfolge ist also nicht egal, und das Skript wäre falsch.

Korrekt:



Schreibe für die folgenden Skripte auf, welche Werte die verwendeten Variablen am Ende des Skriptes haben. Stelle dazu am besten eine Tabelle auf mit allen Variablen und ihren Werten. Wenn der Wert einer Variable verändert wird, streichst du den alten Wert durch und schreibst den neuen Wert auf. Schreibe außerdem auf, was Scratch sagt!

1.



2.



2

Lösung:

1. Die Variable "meine Variable" hat den Wert 20 und Scratch sagt 20.

2. Der Wert der Variable "meine Variable" wird mit dem Wert von "zweite Variable" ersetzt. Somit haben am Ende beide Variablen den Wert 20.

Schreibe für die folgenden Skripte auf, welche Werte die verwendeten Variablen am Ende des Skriptes haben. Stelle dazu am besten eine Tabelle auf mit allen Variablen und ihren Werten. Wenn der Wert einer Variable verändert wird, streichst du den alten Wert durch und schreibst den neuen Wert auf. Schreibe außerdem auf, was Scratch sagt!

1.



2.



Lösung:

1. Auch hier wird zuerst der Wert der Variable mit dem Namen "meine Variable" mit dem Wert von "zweite Variable" ersetzt. Danach wird der Wert der Variable "zweite Variable" auf 30 gesetzt. Dies hat keinen Einfluss auf den Wert von "meine Variable". Somit hat am Ende "meine Variable" den Wert 20 und "zweite Variable" den Wert 30.
2. Diese Variablen wurden schlecht benannt. Der Computer führt die Befehle einfach genau so aus, wie sie dastehen, auch wenn wir zum Beispiel wissen, dass 100 die größere Zahl ist und 1 die kleinere. Beide Variablen haben am Schluss den Wert 100.

Schreibe für die folgenden Skripte auf, welche Werte die verwendeten Variablen am Ende des Skriptes haben. Stelle dazu am besten eine Tabelle auf mit allen Variablen und ihren Werten. Wenn der Wert einer Variable verändert wird, streichst du den alten Wert durch und schreibst den neuen Wert auf. Schreibe außerdem auf, was Scratch sagt!

1.

```

    Wenn [grünes Flag] angeklickt wird
      setze Zähler auf 0
      wiederhole 5 mal
        ändere Zähler um 1
      sage Zähler
  
```

2.

```

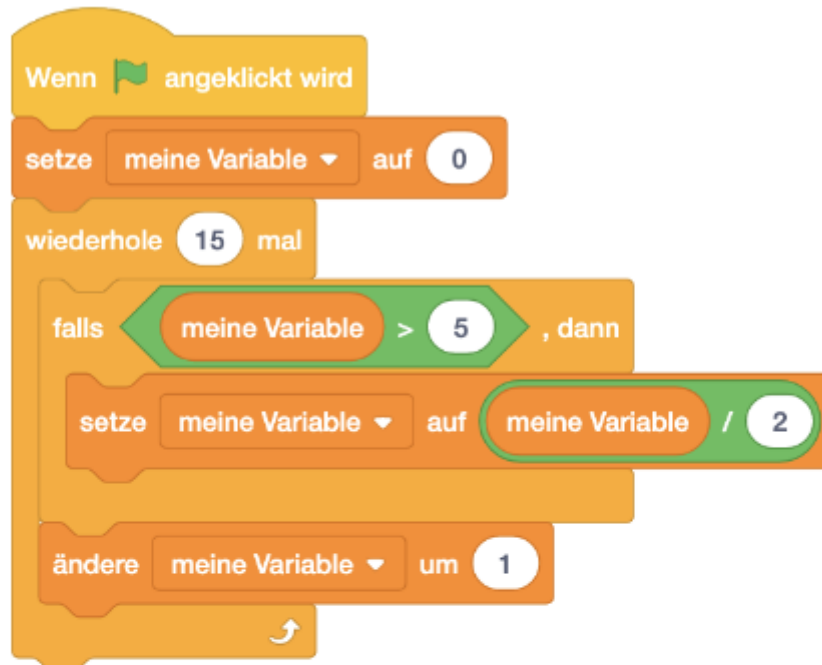
    Wenn [grünes Flag] angeklickt wird
      setze Zähler auf 0
      wiederhole 5 mal
        ändere Zähler um 1
        sage Zähler für 1 Sekunden
  
```

Lösung:

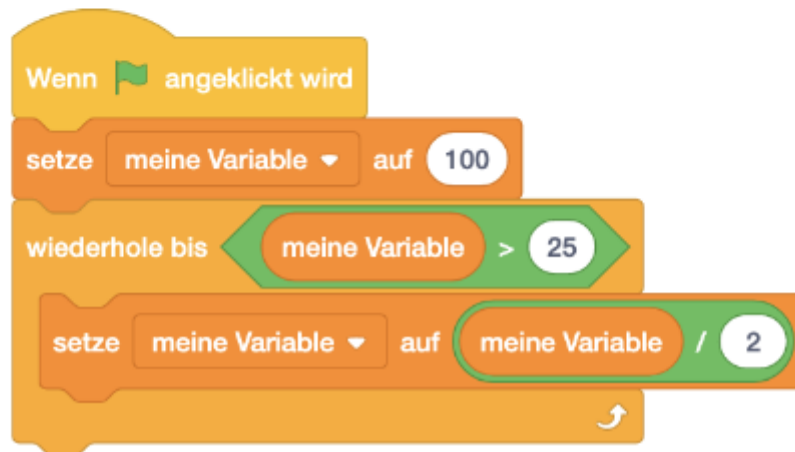
1. Die Variable "Zähler" hat am Schluss den Wert 5. Scratch sagt 5.
2. Die Variable "Zähler" hat am Schluss den Wert 5. Scratch sagt nacheinander 1,2, 3, 4 und 5.

Häufig will man mit Computern etwas ausrechnen. Welchen Wert hat die Variable am Ende des jeweiligen Programms?

1.



2.



4

Lösung:

- Die folgende Tabelle zeigt den Wert von "meine Variable" am Ende des jeweiligen Durchlaufes der Schleife.

Durchlauf:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Wert:	1	2	3	4	5	6	4	5	6	4	5	6	4	5	6

- Da die Bedingung bereits zu Beginn stimmt, wird die Schleife gar nicht ausgeführt und der Wert von "meine Variable" ist 100.

8/9

Im Versteckspiel „Gobo“ soll die Zeit nach 15 Sekunden ablaufen. Für die Programmierung stehen zwei Operatoren zur Verfügung. Welcher zählt die Zeit zurück?

The image shows a Scratch script on the left and a game screenshot on the right. The script starts with a 'Wenn angeklickt wird' block, followed by 'setze Zeit auf 15', a 'wiederhole fortlaufend' loop containing 'warte 1 Sekunden' and 'ändere Zeit um -1', and a 'stoppe alles' block. A red arrow points from the 'dann' slot of the 'falls' block to a table. The table has two rows: the first row contains a green arrow pointing left with 'Zeit < 1' and an empty cell; the second row contains a green arrow pointing right with 'Zeit > 1' and an empty cell. The screenshot shows a character in a cave with a 'Zeit' bar at the top.

1

Lösung:

The image shows the same Scratch script as above, but with a red 'X' in the empty cell of the first row of the table, indicating that the left-pointing arrow is incorrect. The screenshot on the right is identical to the one in the question.