



## Tipp

Die grundlegende Struktur des Algorithmus könnte so aussehen, wie rechts abgebildet.

Überlege zunächst, was die Bedingung ist.

Du findest sie im Menüpunkt „fühlen“. Sie ist ziemlich weit unten und hat eine sechseckige Form.



Füge die passende Bedingung hinter „wenn“ ein.



1a



## Tipp

Die grundlegende Struktur des Algorithmus könnte so aussehen, wie rechts abgebildet.

Überlege zunächst, was die Bedingung ist.

Du findest sie im Menüpunkt „fühlen“. Sie ist ziemlich weit unten und hat eine sechseckige Form.



Füge die passende Bedingung hinter „wenn“ ein.



1a



## Tipp

Die grundlegende Struktur des Algorithmus könnte so aussehen, wie rechts abgebildet.

Überlege zunächst, was die Bedingung ist.

Du findest sie im Menüpunkt „fühlen“. Sie ist ziemlich weit unten und hat eine sechseckige Form.



Füge die passende Bedingung hinter „wenn“ ein.



1a



## Tipp

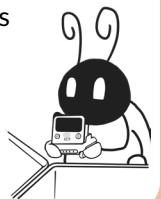


Hier siehst Du die Bedingung („Sieht Hindernisse voraus?“). Überlege Dir nun die Verzweigung:

Beginne mit „sonst“. Was soll der Codey Rocky machen, wenn er kein Hindernis sieht?

Was soll er machen, wenn er ein Hindernis sieht?

Füge die Anweisungen passend ein.



1b

**ddi!**

**1b**

6.3b Hindernissen ausweichen

**ddi!**

**1a**

6.3b Hindernissen ausweichen

**ddi!**

**1a**

6.3b Hindernissen ausweichen

**ddi!**

**1a**

6.3b Hindernissen ausweichen



## Tipp



Hier siehst Du die Bedingung („Sieht Hindernisse voraus?“). Überlege Dir nun die Verzweigung:

Beginne mit „sonst“. Was soll der Codey Rocky machen, wenn er kein Hindernis sieht?

Was soll er machen, wenn er ein Hindernis sieht?

Füge die Anweisungen passend ein.



## Tipp

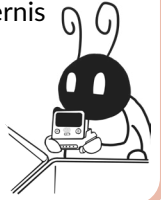


Hier siehst Du die Bedingung („Sieht Hindernisse voraus?“). Überlege Dir nun die Verzweigung:

Beginne mit „sonst“. Was soll der Codey Rocky machen, wenn er kein Hindernis sieht?

Was soll er machen, wenn er ein Hindernis sieht?

Füge die Anweisungen passend ein.



1b

1b



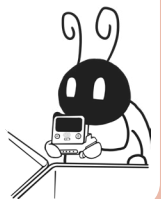
## Tipp

Versuche zunächst, die Zahlen in Deinem bisherigen Code so zu verändern, dass auch größere Hindernisse umfahren werden können oder Hindernisse früher erkannt werden.



## Tipp

Versuche zunächst, die Zahlen in Deinem bisherigen Code so zu verändern, dass auch größere Hindernisse umfahren werden können oder Hindernisse früher erkannt werden.



2a

2a

**ddi!**

**2a**

3b Hindernissen ausweichen

**ddi!**

**1b**

6.3b Hindernissen ausweichen

**ddi!**

**2a**

6.3b Hindernissen ausweichen

**ddi!**

**1b**

6.3b Hindernissen ausweichen



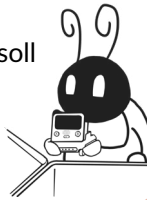
# Tipp

```

wenn  Sieht Hindernisse voraus? , dann
  um 90 Grad nach links drehen
  fahre vorwärts mit 50 % Leistung für 1 sek
  um 90 Grad nach rechts drehen
  fahre vorwärts mit 50 % Leistung für 1 sek
  um 90 Grad nach rechts drehen
  fahre vorwärts mit 50 % Leistung für 1 sek
  um 90 Grad nach links drehen
sonst
  fahre vorwärts mit 20 % Leistung für 0.5 sek

```

Die grundlegende Struktur des Algorithmus von vorhin reicht nun nicht mehr aus. Überlege Dir, an welchen Stellen der Codey wieder auf Hindernisse prüfen soll und füge weitere „wenn, dann“ Verzweigungen ein.



2b



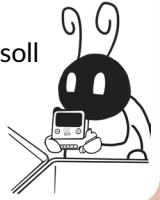
# Tipp

```

wenn  Sieht Hindernisse voraus? , dann
  um 90 Grad nach links drehen
  fahre vorwärts mit 50 % Leistung für 1 sek
  um 90 Grad nach rechts drehen
  fahre vorwärts mit 50 % Leistung für 1 sek
  um 90 Grad nach rechts drehen
  fahre vorwärts mit 50 % Leistung für 1 sek
  um 90 Grad nach links drehen
sonst
  fahre vorwärts mit 20 % Leistung für 0.5 sek

```

Die grundlegende Struktur des Algorithmus von vorhin reicht nun nicht mehr aus. Überlege Dir, an welchen Stellen der Codey wieder auf Hindernisse prüfen soll und füge weitere „wenn, dann“ Verzweigungen ein.



2b



# Tipp

T		(?) sieht Hindernisse voraus ?		F
		um 90° nach rechts drehen		
T		(?) sieht Hindernisse voraus ?		
		um 180° nach links drehen		Ausweichmanöver rechts um das Hindernis starten
T		(?) Hindernisse voraus ?		
um 90° nach links drehen		Ausweichmanöver links um das Hindernis starten		

2c



# Tipp

T		(?) sieht Hindernisse voraus ?		F
		um 90° nach rechts drehen		
T		(?) sieht Hindernisse voraus ?		
		um 180° nach links drehen		Ausweichmanöver rechts um das Hindernis starten
T		(?) Hindernisse voraus ?		
um 90° nach links drehen		Ausweichmanöver links um das Hindernis starten		

2c

**ddi!**

**2c**

6.3b Hindernissen ausweichen

**ddi!**

**2b**

6.3b Hindernissen ausweichen

**ddi!**

**2c**

6.3b Hindernissen ausweichen

**ddi!**

**2b**

6.3b Hindernissen ausweichen

?

## Tipp

Der Code besteht aus vier Teilen. Versuche zunächst, einen oder mehrere Teile ohne Hilfe zu programmieren!

1) Beim Start des Codey Rockys („Initialisierung“) werden die Werte der Variablen festgelegt. Hilfe findest Du dazu bei H1.

2) Der Algorithmus, den der Codey Rocky befolgen soll, wenn er auf einen anderen Codey Rocky (bzw. allgemein ein Hindernis) trifft: Hilfe findest Du dazu bei H2.

3) Der Algorithmus, den der Codey Rocky befolgen soll, wenn er eine Infrarotnachricht empfängt.

Hilfe findest Du dazu bei H3.

4) Den Teil, wenn der Codey Rocky kein Leben mehr hat.

Hilfe findest Du dazu bei H4.

Außerdem sollen die einzelnen Teile sinnvoll verknüpft werden. (H5)



A

?

## Tipp

Bei der „Initialisierung“ sollen zwei grundlegende Informationen festgelegt werden: Die Teamfarbe und die Anzahl der Leben zu Beginn.

Man könnte hier noch mehr festlegen bzw. mit Variablen beschreiben, aber dann wird die Zeit knapp.

Wenn der Codey Rocky startet, dann

- Soll die Variable „Leben“ auf 10 gesetzt werden. (Variable → Neue Variable → „Leben“ (oder „HP“, ...) → Nur für diese Figur → Haken setzen → Setze Leben auf 10)
- Soll der Codey „Team blau“ bzw. „Team rot“ anzeigen (Aussehen)



1a

?

## Tipp

Was passiert, wenn der Codey Rocky auf ein Hindernis trifft?

Zunächst brauchen wir eine Bedingung mit einer Verzweigung (Wenn... dann ... sonst ...)

Wenn er auf ein Hindernis trifft, dann...

Soll der Codey eine Infrarot-Nachricht senden (Inhalt: Farbe Deines Teams) und sich im Anschluss drehen, um eine Kollision zu vermeiden.

Sonst...

Soll er für eine kurze Zeit vorwärtsfahren. Hinweis: Je kürzer die Zeit, desto häufiger überprüft er, ob ein Hindernis im Weg ist.

Weitere Tipps: 2b



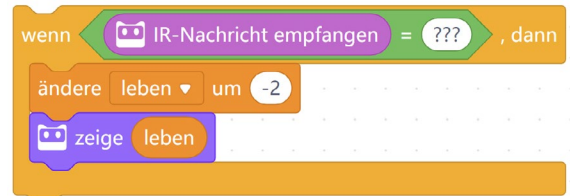
2a

?

## Tipp

Was passiert, wenn der Codey Rocky eine Infrarot Nachricht empfängt?

Er soll überprüfen, ob sie von einem Gegner kommt. Dann soll der Codey seine Anzahl der Leben um zwei reduzieren und die Anzahl der Leben anzeigen.



Die ??? musst Du sinnvoll ersetzen.

Für die Bedingung wenn braucht Ihr den grünen Baustein mit dem = in der Mitte (→ Operatoren)

Weitere Tipps: 3b



3a

**ddi!**

**1a**

6.4 Laserduell

**ddi!**

**Allg.**

6.4 Laserduell

**ddi!**

**3a**

6.4 Laserduell

**ddi!**

**2a**

6.4 Laserduell



?

## Tipp

Was passiert, wenn der Codey Rocky kein Leben mehr hat?

Zunächst brauchen wir eine Bedingung mit einer Folge (Wenn... dann ...)

Wenn die Variable Leben = 0 ist... (grüner Operator. Hinweis: Du musst die Variable nutzen!)

Dann

- Soll der Codey ein trauriges Gesicht machen
- Ein trauriges Geräusch machen
- Alle Bewegungen stoppen (Aktion)
- Alle Skripte stoppen (Steuerung)

Weitere Tipps: 4b



4a

?

## Tipp

Am besten teilst Du den Algorithmus auf zwei Ereignisse auf:

Beim Start des Codey Rockys wird initialisiert (siehe H1a)

Bei Drücken der Taste A sollen die anderen Teile umgesetzt werden:

Es gibt drei Bausteine, die alle fortlaufend wiederholt werden müssen:

Wenn der Codey ein Hindernis sieht, dann (siehe 2a)

Wenn der Codey eine IR-Nachricht empfängt (siehe 3a)

Wenn der Codey Rocky kein Leben mehr hat, dann (siehe 4a)



5a

?

## Tipp

Die Bedingung findest Du im Bereich „fühlen“:

Dann

- Sende IR-Nachricht (mit Deiner Teamfarbe), den Block findest Du bei „Infrarot“
- Um 90° nach rechts drehen (bei „Aktion“)

Sonst

- Fahre vorwärts mit 50% Leistung für 0.2 Sek („Aktion“)  
Je kürzer der Zeitraum ist, desto besser: Da der Codey schnell → er wieder überprüft, ob ein Hindernis im Weg ist.

Weitere Tipps: 2c



2b

?

## Tipp

wenn Codey startet

setze Leben auf 10

zeige Team



1b

**ddi!**

**1b**

6.4 Laserduell

**ddi!**

**5a**

6.4 Laserduell

**ddi!**

**2b**

6.4 Laserduell

**ddi!**

**4a**

6.4 Laserduell

?

## Tipp

```

wenn  , dann
  sende IR-Nachricht 
  um 90 Grad nach rechts drehen
sonst
  fahre vorwärts mit 100 % Leistung für 0.2 sek

```



2c

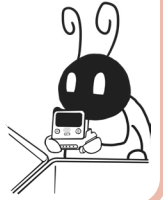
?

## Tipp

```

wenn  , dann
  ändere Leben um -2
  zeige Leben
  Sound abspielen ärgerlich bis zum Schluss

```



3b

?

## Tipp

```

wenn  , dann
  Sound abspielen traurig bis zum Schluss
  zeige Bild 
  Bewegung stoppen
  stopp alle / alles

```



4c

?

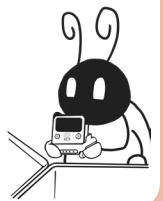
## Tipp

Für Bedingung brauchst Du den grünen Operator = ( oder noch besser <):  
 Hier musst Du überprüfen, ob die Variable „Leben“ (Du findest den orangefarbenen, ovalen Baustein bei „Variablen“) = 0 ist (besser <1!).

Dann

- Soll der Codey ein trauriges Gesicht machen
- Ein trauriges Geräusch machen
- Alle Bewegungen stoppen (Aktion)
- Alle Skripte stoppen (Steuerung)

Weitere Tipps: 4c



4b

**ddi!**

**3b**

6.4 Laserduell

**ddi!**

**2c**

6.4 Laserduell

**ddi!**

**4b**

6.4 Laserduell

**ddi!**

**4c**

6.4 Laserduell