

# Mathematik

## differenziert

1-2025 Üben

Zeitschrift für die Grundschule | [www.mathematik-differenziert.de](http://www.mathematik-differenziert.de)

1-2025



•• **Arbeitsblätter**  
Einfach Online Editieren

## Üben

zielgerichtet – variationsreich – motivierend

Monster führen in die Kombinatorik ein

Fermi-Aufgaben üben das Schätzen und Modellieren

Forschervideos unterstützen das Lösen von Zahlengittern

# Links und rechts mal anders üben!

## Impulse zur Verzahnung mathematischer und informatischer Denkprozesse

Die Fähigkeit, sich real oder mental im Raum zurechtzufinden, lässt sich vielfältig im Geometrieunterricht in der Grundschule fördern. Neben Bewegungsaktivitäten und Brettspielen lassen sich auch digitale Lernumgebungen für Übungen zur räumlichen Orientierung nutzen. Ein Beispiel sind Programmieraktivitäten mit der App „Lightbot“: Sie regen zur Perspektivübernahme an und bieten Kommunikations- und Argumentationsanlässe. Richtungsbegriffe wie „links“ und „rechts“ werden einmal auf andere Art und Weise geübt.

Liebe Leserinnen und Leser,

leider ist die App Lightbot seit dem Druck dieses Heftes nicht mehr nutzbar. Eine Alternative bietet die App AlgoRun, die für Android und iOS kostenlos verfügbar ist. Sie ist vergleichbar, aber nicht identisch. So gibt es den Befehl „Springen“ nicht. Alle anderen Funktionen stehen zur Verfügung.

Wir bedauern diesen Umstand sehr.

Die Redaktion von Mathematik differenziert

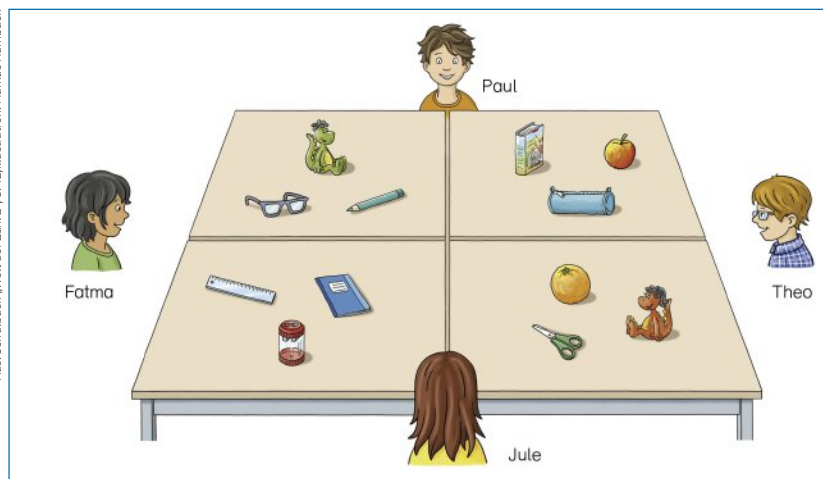
S. 70). Typische Aufgaben sind z. B. Ansichten auf Würfelgebäude oder auf Realobjekte, bei denen die Kinder von ihrer eigenen Perspektive abstrahieren und sich in unterschiedliche Perspektiven hineinendenken müssen (siehe Abb. 1). Was für die Kinder links zu sehen ist, kann in der Realität rechts sein.

Konkrete Handlungserfahrungen stellen eine wichtige Basis für die Entwicklung räumlicher Kompetenzen dar. Von besonderer Bedeutung sind jedoch Übungen zur Kopfgeometrie, die auf ein rein vorgestelltes Handeln abzielen (vgl. Franke/Reinhold 2016, S. 109). Entsprechende Übungen können sowohl analog als auch digital durchgeführt werden und unterstützen die Kinder bei der Perspektivübernahme.

### Analoges und digitales Üben

Alle Bundesländer haben sich 2016 auf die Umsetzung der Strategie „Bildung in der digitalen Welt“ (vgl. KMK 2016) geeinigt. Konkret heißt es in den Bildungsstandards Mathematik: „Zu den informatischen Kompetenzen, die in nicht-informatischen Kontexten im Mathematikunterricht in der Primarstufe gefördert werden können, zählen insbesondere [...] das Verstehen und Anwenden von Algorithmen, von symbolischer und formaler Sprache“ (KMK 2022, S. 8). Plötzlich müssen wir uns als Lehrkräfte in der Welt der Informatik orientieren. Häufig entsteht das Gefühl, man müsse „noch mehr“ in dem ohnehin schon vollen Schuljahr unterbringen. Umso erleichternder ist es doch, wenn man bei genauer Betrachtung feststellt, dass gar nicht „noch mehr“, sondern nur „anders“ gearbeitet werden kann.

Mit den hier vorgestellten Aufgaben kann eine Verzahnung von informatischer und mathematischer Kompetenzförderung gelingen. Beim Programmieren stehen uns durch die digitale Welt vielfältige Übungsmöglichkeiten zur Verfügung. Für eine davon – die App Lightbot – werden hier erprobte Ideen zum Einsatz im Unterricht vorgestellt.



•• Abb. 1: Die Kinder müssen unterschiedliche Perspektiven berücksichtigen (Abb. aus WdZ 2, S. 12).

von Jana Klusmeyer und Katharina Brinkmann

Die Entwicklung des räumlichen Vorstellungsvermögens ist ein zentrales Anliegen des Geometrieunterrichts in der Grundschule (KMK 2022, S. 16). Zentral sind dabei mentale Prozesse: Statt ausschließlich konkrete Handlungen durchzuführen, nehmen die Kinder vor allem gedankliche Manipulationen vor. Eine wichtige Teilkomponente des räumlichen Vorstellungsvermögens ist die „Räumliche Orientierung“, bei welcher es darum geht, „sich gedanklich in andere Perspektiven hineinzuversetzen“ (Franke/Reinhold 2016,



•• Abb. 2: Analog und digital: Die Kinder tauschen sich durch Erprobung in der analogen Welt über ihre Erfahrungen im digitalen Setting aus.

Bewusst wird der Blick auch auf analoge Übungen gerichtet, da auch hier bereits viele Chancen für das Verstehen und Anwenden von Algorithmen sowie symbolischer und formaler Sprache zu finden sind.

Die vorgestellten Übungen können flexibel eingesetzt werden: Denkbar ist eine kompakte Einheit in Klasse 3/4 als Block oder der Beginn mit den ersten analogen Übungen bereits in Klasse 1/2 und die Fortführung im digitalen Bereich in 3/4. Im Sinne des Spiralcurriculums werden konkrete Handlungserfahrungen vertieft, systematisch geordnet, genutzt und erweitert (KMK 2022, S. 16).

### Übung 1: Offline Coding mit menschlichen Robotern

Ganz analog, ganz konkret selbst erfahren, was Programmieren bedeutet: Die Kinder schlüpfen in die Rolle eines Roboters (vgl. auch Wagner/Rieseneder 2023). Auf einem Spielfeld aus Schaumstoffmatten oder Teppichfliesen schicken sie sich zunächst durch mündliche Befehle gegenseitig von einem Startpunkt zu einem bestimmten Ziel. Das Ziel kann durch eine laminierte Schatztruhe o. Ä. gekennzeichnet werden. Wichtig ist, dass in dieser Phase bereits eine eindeutige Programmiersprache eingeführt wird (siehe Kasten Formulierungshilfen). Diese beinhaltet bei dieser Übung die Richtungsbegriffe „rechts“, „links“ und „vorwärts“, weil sich diese mit der Programmiersprache der App *Lightbot* decken. Zudem muss ein eindeutiges Wort für das Zielfeld vereinbart werden, z. B. „Schatz“ oder „Ziel“.



## Auf einen Blick

**Klassenstufe:** 1–4

**Zeit:** etwa 6 Schulstunden

**Lerngelegenheiten:**

- Richtungsbegriffe
- räumliches Vorstellungsvermögen
- Perspektivübernahme
- Kommunizieren und Argumentieren
- algorithmisches Denken

**Materialseiten:**

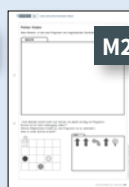
- M1** Spielfeld
- M2** Fehler finden
- M3** Forschungsaufträge
- M4** Programmablauf nachvollziehen



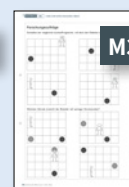
**Materialseiten  
downloaden oder  
online bearbeiten!  
Infos auf Seite 51**



S. 28



S. 29



S. 30



S. 31

**Zusätzlich:**

- App *Lightbot* (für Android/iOS kostenlos verfügbar und offline nutzbar)
- Teppichfliesen, Schaumstoffmatten oder Straßenkreide
- DIN-A4-Magnetafeln
- Magnetfolie selbstklebend für die Symbolkarten
- Foldback-Klammern für die Roboterspielfiguren
- Materialboxen, z. B. Schraubenkästchen für Symbolkarten und Spielfiguren

## Formulierungshilfen

*Beim Programmieren sehen wir uns mit einer ganzen Reihe (englischer) Fachbegriffe konfrontiert. Erfahrungsgemäß finden die Kinder das sehr motivierend!*

*Programmiersprache = eindeutige Befehle oder Symbole, die der Roboter versteht*

*Programm = Zusammenstellung mehrerer Kommandos zum Erreichen eines Ziels*

*main = Hauptbereich des Programms (mit begrenztem Platz für Befehle)*

*coding = Übersetzen der Handlungsschritte in Programmiersprache*

*bug = ein Fehler im Programm (übersetzt übrigens „Ungeziefer“)*

*debugging = den Fehler im Programm beheben*

Foto: Jana Klusmeyer/Katharina Brinkmann



•• **Abb. 3: Wie geht's weiter? Die Kinder diskutieren beim Offline Coding ihre konkreten Handlungserfahrungen.**

### Perspektivübernahme anregen

Spannend wird es, wenn mehrere Kinder um das Spielfeld verteilt stehen und so unterschiedliche Perspektiven auf den Weg des Roboters einnehmen. Hier wird die räumliche Orientierung geschult; die Kinder müssen sich mental in die Perspektive des Roboters hineinversetzen. Erfahrungsgemäß entstehen in dieser Phase interessante Stolperstellen, über die diskutiert wird (siehe Abb. 3): *Wo ist eigentlich links? Was passiert, wenn der Roboter mir gegenübersteht? Darf der Roboter mitdenken und sich nach rechts drehen, obwohl das Kind links gesagt hat?*

Als Reflexionsanlass sollte nach einigen Durchgängen über die gesammelten Erfahrungen gesprochen werden. Der Austausch in der Gruppe regt dazu an, eigene Annahmen zu reflektieren und neue Strategien zu erproben. Als Lehrkraft erhalten wir Einsicht in die Denkprozesse der Kinder.

Damit das Roboterkind zunehmend lernt, auf die Anweisungen der programmierenden Kinder zu vertrauen, kann der Schatz bei weiteren Übungen unter einer Teppichfliese versteckt werden.

### Differenzierung: Schreiben und Lesen in Programmiersprache

Der Schwierigkeitsgrad erhöht sich deutlich, wenn sich die Kinder im nächsten Schritt den kompletten Weg gedanklich vorstellen müssen. Wie bei einem echten Programm auch, wird nun der Weg im Vorfeld codiert. Dazu wird zu jedem Begriff das entsprechende Symbol eingeführt (siehe M1). Die Symbole orientieren sich an denen der App *Lightbot*, um die spätere Übertragung zu erleichtern. Mit Magnetfolie können diese Symbolkärtchen auf Magnettafeln fixiert werden. Das Roboterkind verlässt den Raum, während die anderen Kinder den Schatz verstecken. Nun tauchen wir richtig ein in die Welt des Programmierens: Die Kinder laufen den Weg mental ab, planen die einzelnen Schritte und müssen ein vollständiges Programm mit den Symbolkärtchen „schreiben“. Durch die Arbeit in Kleingruppen werden die Richtungsbegriffe angewendet und es wird über die Perspektive diskutiert. Das Roboterkind darf das Programm dann lesen und ablaufen.

Im Optimalfall entstehen hier Fehler in der Planung. Für uns die ideale Lernchance: Der Fachbegriff „Bug“ (siehe Kasten Formulierungshilfe) wird eingeführt, die Kinder begeben sich gemeinsam in den „Debugging“-Prozess, wie echte Informatikerinnen und Informatiker.

### Übung 2: Brettspiel – Programmiersprache, Debugging und Effizienz üben

Als weiterführende und vertiefende Übung bietet es sich an, den menschlichen Roboter und den konkret ablaufbaren Weg durch ein Brettspiel zu ersetzen. Dabei wird die mentale Perspektivübernahme angeregt. Die Kinder dürfen das Material (siehe M1) zunächst jeweils in Partnerarbeit erproben, indem sie Schätze (z. B. Muggelsteine) positionieren und die Roboter-

Spielfigur den Weg dorthin entsprechend mit den Symbolkärtchen codieren (siehe Abb. 4). Aufgaben zum Lesen von Programmen (siehe M4) können zur gezielten Vertiefung anregen.

In diesem Lernsetting kann problemorientiert das Debugging geübt werden, indem die Kinder fehlerhafte Programme reparieren (siehe M2): *Tims Roboter kommt nicht zum Schatz. Es steckt ein Bug im Programm. Kannst du ihm beim Debugging helfen?*

Eine weitere Erkenntnis, die für die Weiterarbeit im digitalen Raum wichtig wird, ist die Effizienz eines Programms. Es ist sinnvoll – und wie wir in der digitalen Welt später merken werden – sogar notwendig, ein Programm möglichst kurz zu schreiben. In Übung M3 werden die Kinder deshalb dazu angeregt, Wege mental zu vergleichen, sich für den kürzeren zu entscheiden und diese Entscheidung dann konkret mit der Spielfigur zu überprüfen.



Foto: Jana Klusmeyer/Katharina Brinkmann

•• **Abb. 4: Vertiefung: Zu zweit bearbeiten und erfinden die Kinder Problemstellungen.**

### Differenzierung: Möglichkeiten finden

Im Sinne des entdeckenden Übens können die Kinder Möglichkeiten für Zielfelder finden. *Tim hat genau drei Kommandos für sein Programm genutzt und damit den kürzesten Weg zu seinem Ziel gefunden. Auf welchen Feldern könnte der Schatz sein?*

Für diese Übungsphase sind der Kreativität keine Grenzen gesetzt. Die Kinder könnten sich auch eigene Problemstellungen überlegen und bearbeiten.

### Übung 3: Programmieren in einer digitalen Lernumgebung

Das erworbene Wissen wird mit der App *Lightbot* im digitalen Raum vertiefend geübt. Reizvoll für die Kinder ist der Einsatz von Tablets (siehe Abb. 5). Aus didaktischer Sicht bietet die App *Lightbot* den Vorteil, dass sie sich auf die wesentlichen Aspekte des Programmierens konzentriert. Optisch reizarm liegt der Fokus auf dem Coding als solches, was die Kinder bereits aus dem analogen Lernsetting kennen. Das Ziel bleibt das gleiche: *Programmiere den Weg des Roboters so, dass er die Zielfelder erreicht.* In der App werden, anders als bisher, die Zielfelder beleuchtet (daher der Name *Lightbot*).



•• **Abb. 5: Die App Lightbot regt zur Perspektivübernahme und zum algorithmischen Denken an.**

Im Bereich „Grundlagen“ wird Level für Level der Schwierigkeitsgrad erhöht. Die Wege werden länger und anspruchsvoller. Das neue Kommando „springen“ kommt hinzu (wobei dieses auch auf dem Spielfeld in Übung 2 denkbar wäre, wenn Bauklötze zur Verfügung stehen).

### Differenzierung: Komplexität mentaler Prozesse variieren

Für Kinder mit weniger ausgeprägter Vorstellungskraft können in der App *Lightbot* Zwischenschritte flexibel überprüft werden. Durch Drücken der Play-Taste läuft der Roboter den bis dahin programmierten Weg ab. Kommandos können verändert oder hinzugefügt werden, bis der Roboter an seinem Ziel ankommt. Die Kinder können über den Schwierigkeitsgrad selbst entscheiden, indem sie überlegen, wie viele Schritte sie mental planen können, bevor sie das Programm erproben.

### Vertiefung: Prozeduren und Schleifen

Nach dem Bereich Grundlagen, der acht Level umfasst, bietet die App *Lightbot* zwei weitere kostenfreie Bereiche mit jeweils sechs Leveln an. Hier tauchen wir deutlich tiefer in die Informatik ein. Die Kinder haben die Chance, durch „Prozeduren“ und „Schleifen“ Programme effizienter zu gestalten. Erfahrungsgemäß lassen sich hier vor allem Kinder mit einem guten räumlichen Vorstellungsvermögen und einem Blick für Muster und Strukturen begeistern, die sich den Problemstellungen mit Ausdauer stellen.

### Fazit

Im Geometrieunterricht lässt sich die räumliche Orientierung und die dabei zentrale Perspektivübernahme durch vielfältige Aktivitäten fördern: Konkrete Bewegungsaktivitäten werden durch Brettspiele und durch Übungen mit einer App ergänzt. Ähnliche Übungen finden analog sowie digital statt und unterstützen über die Wiederholungen die Entwicklung des räumlichen Vorstellungsvermögens.

Durch Programmieraktivitäten entsteht zudem eine Schnittstelle von Informatik und Mathematik. Es erge-

ben sich vielfältige analoge und digitale Übungsanlässe zur Schulung inhaltsbezogener- und prozessbezogener mathematischer Kompetenzen. Insbesondere beim digitalen Üben stehen durch die Funktionen der App Differenzierungsmöglichkeiten zur Verfügung. Im Sinne des Spiralcurriculums können Kinder aller Jahrgangsstufen der Grundschule durch konkrete Handlungserfahrungen ihre Fähigkeiten vertiefen und auch im digitalen Raum nutzen. ■

### Literatur

Franke, M./Reinhold, S.: *Didaktik der Geometrie*. Berlin, Heidelberg 2016

Sekretariat der Kultusministerkonferenz: *Bildung in der digitalen Welt - Strategie der Kultusministerkonferenz*. Berlin 2016

Sekretariat der Kultusministerkonferenz: *Bildungsstandards für das Fach Mathematik – Primarbereich*. Berlin 2004 (i. d. F. vom 23.06.2022)

Wagner, W./Rieseneder, M.: *Spielerisch Programmieren lernen*. In: *Mathematik differenziert*, Heft 2-2023, S. 26–31

### Die Autorinnen

Foto: Privat



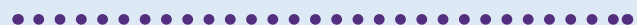
**Jana Klusmeyer** ist Grundschullehrerin und abgeordnete Lehrkraft im Institut der Didaktik für Mathematik an der Universität Bielefeld.

Foto: Privat



**Katharina Brinkmann** ist Grundschullehrerin und Mitglied des Kompetenzteams der Stadt Bielefeld im Bereich Medien-erziehung.

## Stolpersteine



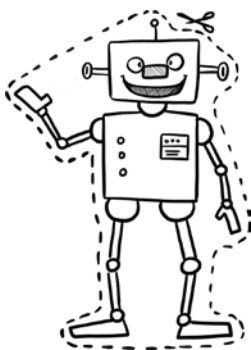
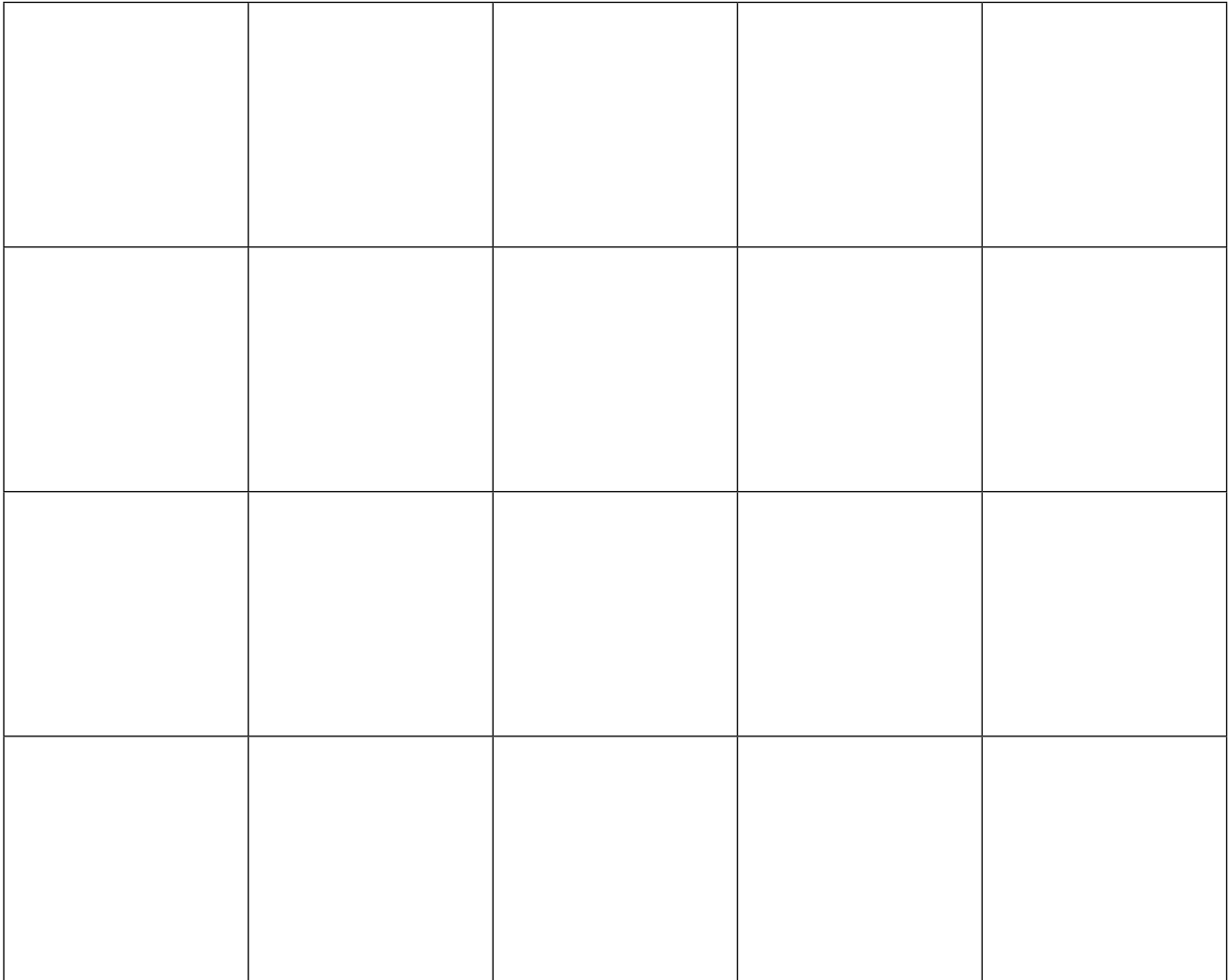
*Das digitale Angebot von Apps, Bots und Software zum Programmieren kann mitunter überfordernd wirken. Wie immer gilt es, eine Passung zwischen Lernziel und Umsetzung zu finden.*

*Wenn die Kinder beispielsweise im räumlichen Vorstellungsvermögen geschult werden sollen, muss das gewählte Medium dazu anregen, mehrere Schritte des Programms mental durchzugehen, statt Schritt für Schritt zu codieren.*

*Die Perspektivübernahme kann am besten geschult werden, wenn der Roboter sich entsprechend authentisch durch den Raum bewegt. Das ist nicht bei allen Apps der Fall!*

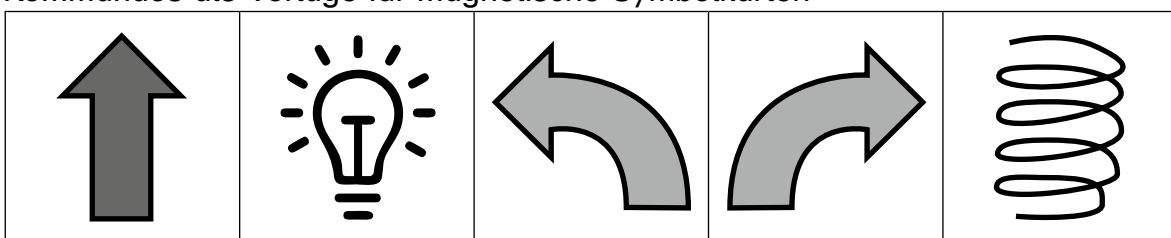
*Tablets und Bots motivieren sicherlich immer – aber der mathematikdidaktische Fokus ist nicht zwangsläufig gegeben!*

## Spielfeld



Roboter zum Ausschneiden & Laminieren;  
als Spielfigur mit Foldback-Klammer stabilisieren

Kommandos als Vorlage für magnetische Symbolkarten

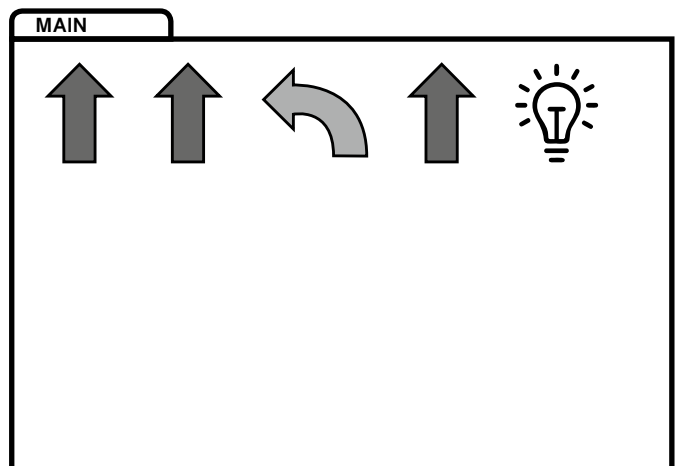
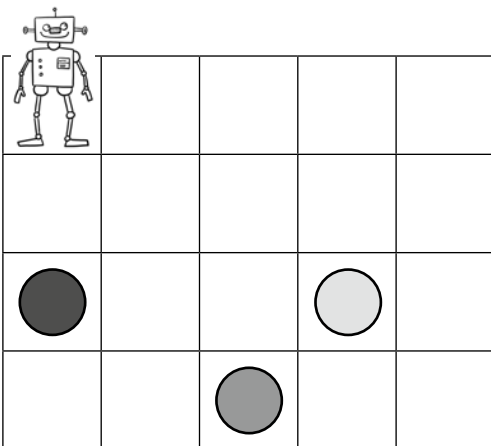


## Fehler finden

Main-Bereich, in den das Programm mit magnetischen Symbolkarten „geschrieben“ wird

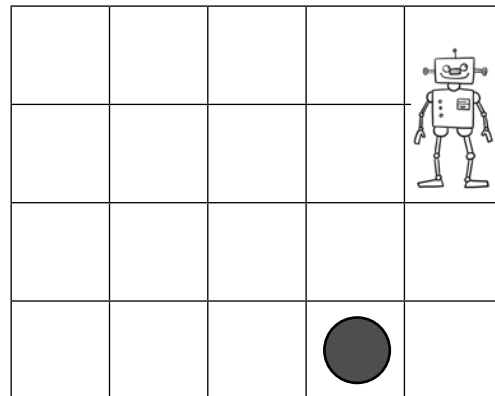
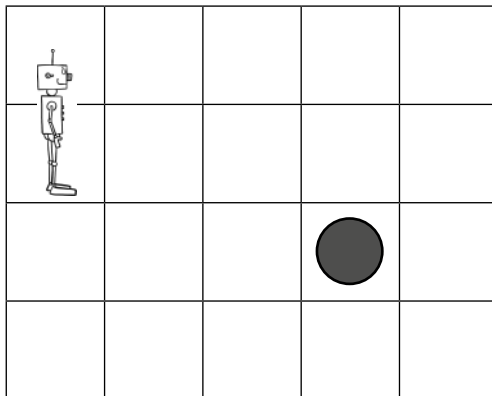
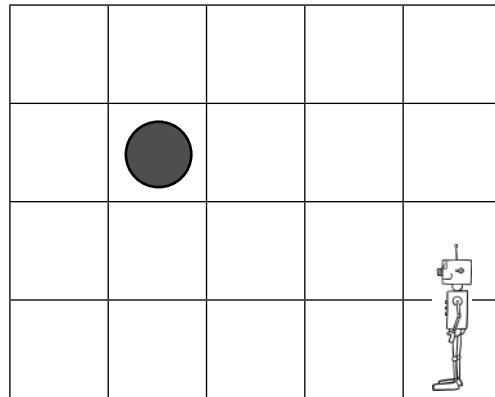
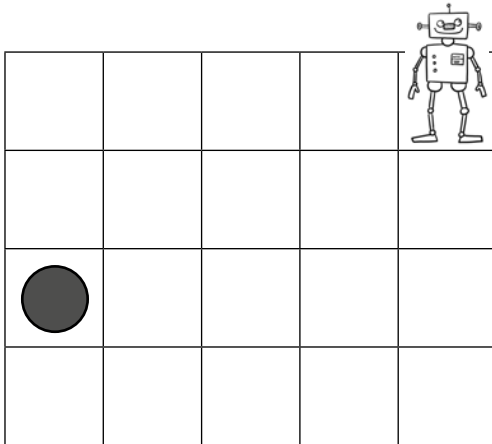
**MAIN**

„Tims Roboter kommt nicht zum Schatz. Es steckt ein Bug im Programm.  
Kannst du ihm beim Debugging helfen?“  
Welche Möglichkeiten findest du, das Programm so zu verändern,  
dass er einen Schatz erreicht?

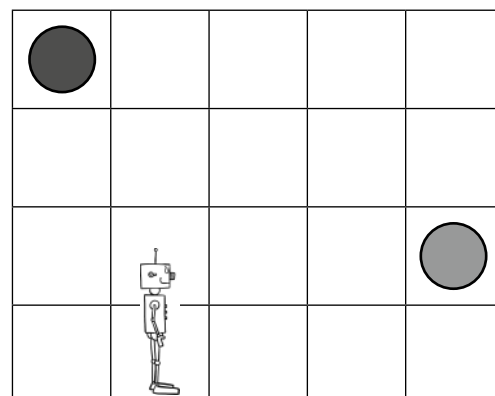
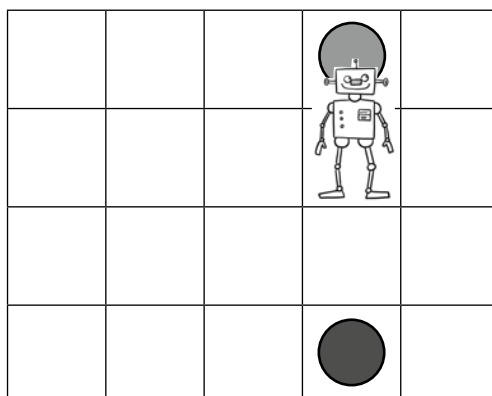
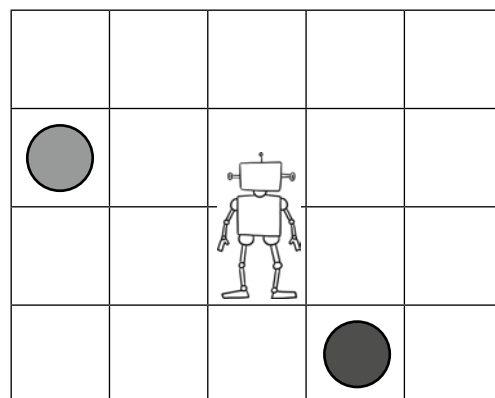
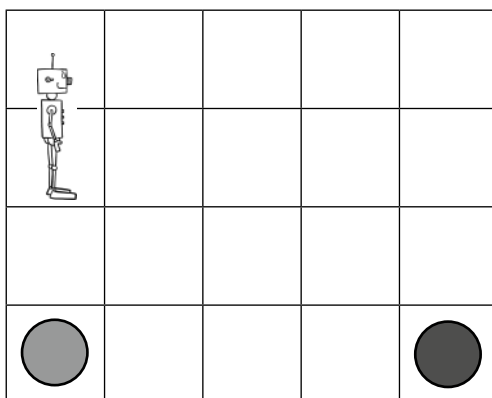


## Forschungsaufträge

Schreibe ein möglichst kurzes Programm, mit dem der Roboter den Schatz erreicht.



Welchen Schatz erreicht der Roboter mit weniger Kommandos?



## Programmablauf nachvollziehen

Welchen Schatz erreicht der Roboter, wenn er das Programm durchlaufen hat?

