

## DIDAKTISCHER KOMMENTAR BASISMODUL

- BP – BASISMODUL TEIL 1 – WAS IST EIN PROGRAMM?
- BS – BASISMODUL TEIL 2 – WAS IST SCRATCH?
- BU – BASISMODUL TEIL 3 – UNTERSCHIEDUNGEN UND SCHLEIFEN
- BZ – BASISMODUL TEIL 4 – ZAHLEN UND ZÄHLEN

### DIDAKTISCHE GRUNDSÄTZE

Die vier Teile des **Basismoduls** führen in das algorithmische Denken über das Programmieren mit Scratch im Mathematikunterricht ein. Wenn die Kinder das Basismodul bearbeitet haben, lassen sich Mathe-Module in beliebiger Auswahl anschließen. Das Basismodul kann daher gut in Klasse 3 im zweiten Halbjahr bearbeitet werden, sodass sich in Klasse 4 ausgewählte Mathe-Module anschließen lassen. Je ein Teil des Basismoduls passt gut in eine Schulwoche, die Bearbeitungszeit beträgt etwa 1-3 Schulstunden.

Es hat sich als äußerst sinnvoll erwiesen, die Kinder grundsätzlich zu zweit daran arbeiten zu lassen. Daher sind alle Materialien für **Partnerarbeit** ausgelegt. Die **digitalen Aufgaben** können zwei Kinder gemeinsam an einem Tablet-PC oder einem Laptop/ PC bearbeiten. In der Erprobung wurde fast immer mit iPads gearbeitet, darauf sind die Materialien abgestimmt. Sollte ein anderes Betriebssystem oder eine andere Endgeräteform benutzt werden, sind die Darstellungen ggf. leicht abweichend, so dass die Lehrkraft ggf. ergänzende Hinweise geben muss, jedoch ist dies problemlos möglich. Die einzige **Voraussetzung** ist ein installierter Internetbrowser und ein Internetzugang. Es werden nur geringe Datenmengen übertragen, so dass eine kleine Internetbandbreite ausreicht. Zum Handling der Programmierumgebung Scratch auf iPads können aus der Erprobung heraus noch zwei Tipps gegeben werden: 1) Die iPad-Tastatur lässt sich mit zwei Fingern zusammenschieben und „schwebt“ dann verschiebbar über dem iPad-Bildschirminhalt – anderenfalls begrenzt sie häufig den sichtbaren Bereich stark. Durch Auseinanderziehen mit 2 Fingern lässt sich die Tastatur wieder in den Ursprungsmodus versetzen. 2) Um mehr Code im mittleren Bereich von Scratch sehen zu können, lässt sich die Bühne auf der rechten Seite in Scratch mit den darüber angebrachten Tasten (siehe Abbildung rechts) zwischen einer Klein- und Normalversion (und nebenbei bemerkt, ins Vollbild) umschalten. Die Kleinversion bietet dann mehr Platz für den Code-Teil.



Für jeden Teil des Basismoduls existiert eine **Aufgabenkarte**, auf denen die Kinder den Arbeitsfortschritt festhalten und zum Teil auch Notizen anfertigen. Diese Aufgabenkarten können als PDF sowohl auf der Projektseite [promaprim.de](http://promaprim.de) heruntergeladen werden, als auch über die Programmier-Startseite [3yp.de](http://3yp.de) für die Kinder. Dort sind sie immer jeweils in der ersten Aufgabe zu finden. Eine solche Aufgabenkarte sollte ein Partnerteam in **Papierform** bekommen, so dass die Lehrkraft sie vorab kopiert oder beim Ausdrucken hilft. Ebenso gibt es einige wenige **Arbeitsblätter**, die benötigt werden, die in der Regel als weitere Seite den Aufgabenkarten anhängen. Auch hier brauchen die Kinder eine Papierversion und Unterstützung durch die Lehrkraft.

Zum Teil werden **Arbeitsaufträge** direkt in den Aufgabenkarten gegeben, zum Teil erfolgen diese digital über die Plattform [3yp.de](http://3yp.de). Die dort im Browserfenster erscheinenden Arbeitsaufträge können auf [promaprim.de](http://promaprim.de) in Karteiform nebst einer tabellarischen Übersicht aller Basismodul-Aufgaben heruntergeladen werden, so dass Lehrkräfte diese zur erweiterten oder alternativen Unterrichtsplanung verwenden können.

Grundsätzlich wurde sich im Projekt ProMaPrim für ein möglichst **einfaches Dateihandling** entschieden, so dass keine Logins oder Konten notwendig sind. Die Kinder rufen sämtliche Lernangebote über die einfach

einzugebende **URL 3yp.de** (Merkhilfe: 3yp = ‚try your programm‘) auf. Digitale Arbeitsergebnisse werden nicht gespeichert, können aber per Screenshot (am iPad Home+Power) gesichert werden.

Durch die Aufgabenkarten sind die Basismoduleteile gut für **offenes Arbeiten** (z.B. in Wochenplänen) geeignet, so dass sich nicht alle Kinder zur gleichen Zeit damit beschäftigen müssen. Als Lehrkraft können Sie jedoch überlegen, ob und welche Inhalte sie mit den Kindern **gemeinsam besprechen/** durchführen wollen. In der Erprobung hat es sich als sinnvoll erwiesen, die an vielen Stellen vorhandenen Erklärvideos durch einführenden **Frontalunterricht** zu ergänzen oder zu ersetzen, wenn geeignete technische Bedingungen (Beamer, Digitaltafel, ...) vorhanden sind. Die Erklärvideos sind mit Untertiteln belegt, so dass auch ohne Ton gearbeitet werden kann.

Ebenso kann es sich anbieten, einzelne **Arbeitsergebnisse** der Kinder gemeinsam zu besprechen (bei vorhandener technischer Ausstattung der Schule beispielsweise AirDrop des Programmierergebnisses vom Kinder-Tablet auf die digitale Tafel). Zudem kann die Lehrkraft überlegen, einen gemeinsamen Einstieg oder eine Abschlussreflexion durchzuführen. Die Aufgabenkarten haben also eine Art „Laufkarten“-Charakter, ersetzen aber nicht eine gründliche Unterrichtsplanung mit ggf. überlegten handlungsbedingten Alternativen.

Zudem hat sich in der Erprobung gezeigt, dass die verwendete **Sprache** im Sinne eines sprachbildenden Mathematikunterrichtes eine große Rolle spielt. Daher sollten Sie als Lehrkraft ebenso überlegen, **Begriffe und Formulierungen** vorab mit den Kindern gemeinsam zu erarbeiten. In manchen Aufgabentypen sollen die Kinder selbst Programmschritte in einem genannten **Programmablaufplan** (kurz: **PAP**) aufschreiben, dabei können sie die angebotenen **Sprachspeicher** verwenden, in anderen sollen sie fertige Programmablaufpläne interpretieren. Dazu lassen sich einige wichtige Kernpunkte festhalten:

- Verwendung des **Imperativs**: Aus der Erprobung wissen wir, dass die Kinder bei eigenen Beschreibungen gerne Formulierungen in der zweiten Person verwenden (Beispiel: „Du musst zuerst ... falten.“). Es ist jedoch propädeutisch sinnvoll, sie direkt zur Nutzung der Imperativ-Form (Beispiel: „Falte zuerst...“) anzuhalten, denn später sind auch alle Programm-Blöcke in der Programmierumgebung Scratch (wie in allen anderen üblichen Programmiersprachen auch) in der imperativen Form geschrieben. Gegebenenfalls sollten Sie also diesen Sprachduktus thematisieren.
- Die angebotenen Begriffe und Formulierungen in den **Sprachspeichern** sind nicht selbsterschließend. Vielfach muss **erst besprochen werden**, was zum Beispiel „Wiederhole ... bis ...“ bedeutet. Zudem stellen die Sprachspeicher Angebote dar, zum Teil mit Dopplungen/ Alternativen. Aus der Erprobung wissen wir, dass manche Kinder sich verpflichtet fühlen, alle Begriffe zu verwenden und abzuheben. Weisen Sie also darauf hin, dass dies nicht erforderlich ist. Und selbstverständlich sind auch eigene Formulierungen erlaubt. Dies gilt später dann analog auch in den Programmierumgebungen, in denen bereits Programmblöcke als Angebot bereit liegen, aber nicht immer vollständig genutzt werden müssen, und durch andere Blöcke ersetzt werden können.
- Ein Grundproblem bei der **Formulierung von Teilschritten** eines Algorithmus besteht darin, wie ausführlich oder wie kompakt die Beschreibungen ausfallen, und wie ausführlich oder wie kompakt der Algorithmus in Teilschritte aufgeschlüsselt wird. Zum Beispiel ließe sich die Formulierung „Falte die untere T- Shirt-Kante nach oben“ in die vielleicht zu ausführliche Form „Nimm die untere T-Shirt-Kante mit beiden Händen auf. Bewege die Kante Richtung Kragen. Lege sie so ab, dass...“ ausdifferenzieren, während ein einfaches „Falte hoch“ dann vielleicht doch zu unpräzise erscheint. Ein **passendes Maß an Abstraktion** zu finden, ist dabei eines der Lernziele bei der Entwicklung algorithmischer Denkfähigkeit, hier gibt es keine eindeutig richtigen Lösungen. Weiterhin ist die **Eindeutigkeit** eines Algorithmusschrittes ein wichtiges Kriterium. Eine Formulierung, wie „Falte nach links oder rechts“, ist zum einen nicht eindeutig, und müsste als Verzweigung dargestellt werden – zum anderen ist „links“ zwar aus der Betrachtendenperspektive eindeutig, hier ist aber unklar, was genau nach links gefaltet werden soll (die rechte T-Shirt-Kante?), bis wohin genau (auf die linke T-Shirt-Kante?). Zum einen haben

wir dazu Aufgaben implementiert, in denen die Kinder die Wirksamkeit ihrer Beschreibungen untereinander testen. Zum anderen wird es sinnvoll sein, dass Sie Beschreibungsversuche der Kinder **in gemeinsamen Besprechungsphasen** thematisieren, damit die Kinder voneinander lernen können, und Sie diese im Sinne einer fortschreitenden Mathematisierung kommentieren, um Anregungen für passend abstrahierte Beschreibungen zu geben. Auch dieser Grundgedanke lässt sich später auf die selbst entwickelten Programme übertragen.

- Die auf einigen Arbeitsblättern angebotenen **Templates für Programmablaufpläne** stellen eine Darstellungshilfe dar. Aus der Erprobung wissen wir, dass einige Kinder sich verpflichtet fühlen, alle Kästchen auszufüllen – dies ist so nicht gedacht. Die Programmablaufpläne können beliebig verkürzt oder verlängert werden, und auch in ihrer Struktur geändert werden. Dazu gibt es in den Modul-Materialien und auch als einzelnes Dokument auf der Materialseite der Webseite ein „PAP-Template“, das Sie ausgedruckt bereithalten können (für grobmotorisch Schreibende ggf. auch in A3), damit die Kinder bei Bedarf die Templates der Aufgabenblätter abändern können. Zudem können wir aus der Erprobung berichten, dass die Kinder die einzelnen Kästchen der Templates unterschiedlich voll beschriften. Es ist sinnvoll, pro Kästchen einen Programmschritt aufzuschreiben – im vorangehenden Absatz wurde aber bereits deutlich, dass dies eben nicht eindeutig bestimmt ist, was genau ein Programmschritt ist. Auch hier sind gemeinsame Besprechungen sinnvoll, damit die Kinder ein Gefühl für eine passende Schritt-Formulierung bekommen.

Zum Abschluss sei erwähnt, dass es hilfreich ist, wenn die Lehrkraft das Basismodul vor Beginn der Unterrichtsaktivitäten gründlich durcharbeitet und die unterrichtliche Rahmung überlegt, um selbst ‚fit‘ in der Bedienung der analogen wie digitalen Medien und der Programmierumgebung zu sein und ein adaptiv auf die Lerngruppe passendes Angebot bereiten zu können. ‚Mitlernen‘ während der Aufgabenaktivitäten hat sich in der Erprobung als erhöht stressauslösend erwiesen, wird sich aber auch bei gründlicher Vorbereitung in Details nicht immer vermeiden lassen.

## ÜBERSICHT ÜBER DIE BASISMODUL-TEILE

Im **Basismodul Teil 1** (BP – Was ist ein Programm?) lernen die Kinder zunächst **einfache Programmstrukturen** kennen, die anhand von **Programmablaufplänen (PAPs)** dargestellt werden. Das Kennenlernen dieser PAPs ist wichtig, denn sie spielen in den Mathe-Modulen später eine wichtige Vermittlerrolle im Transfer zwischen arithmetischem und digitalem Algorithmus.

Das **Basismodul Teil 2** (BS – Was ist Scratch?) führt dann in die **Programmierungsumgebung Scratch** und ihre Basishandhabung ein.

Im **Basismodul Teil 3** (BU – Unterscheidungen und Schleifen) werden die einfachen PAPs aus Basismodul Teil 1 wieder aufgegriffen und erweitert: Während in Modul BP die informatische Grundidee *Sequenz* – also aufeinanderfolgende Schritte – aufgegriffen wird, so werden in Modul BU dann die informatischen Grundideen *Bedingung* und *Wiederholung* und in beider Kombination (*Wiederholung mit Bedingung*) thematisiert. Die Behandlung dieser **drei informatischen Grundideen** wird von der Gesellschaft der Didaktik der Informatik für die Grundschule empfohlen, ebenso werden diese Grundideen in den Mathe-Modulen benötigt und dort wieder aufgegriffen.

Das **Basismodul Teil 4** (Zahlen und Zählen) stellt dann die **vertiefende Einführung in Scratch** in einem **mathematischen Kontext** dar (Einführung von Variablen, Rechenoperationen, Vergleiche, Ausgabevarianten). Damit sind alle Grundlagen für die sich anschließenden Mathe-Module gelegt.

## BP - BASISMODUL TEIL 1 - WAS IST EIN PROGRAMM?

Vorzubereitendes Material aus *BP1.pdf* (verfügbar unter *promaprim.de* - Projektmaterialien oder unter *3yp.de*):

- Blatt *Aufgabenkarte* (mit Anleitung): 1x pro Partnerteam.
- Arbeitsblatt *BP6* (ein T-Shirt falten): 1x pro Partnerteam.
- Arbeitsblatt *PAP-Vorlage*: ‚mehrfach‘ - diese Vorlage wird mindestens zweimal pro Partnerteam benötigt und könnte als ‚Verbrauchsmaterial‘ bereitliegen.
- Dazu ein digitales Arbeitsmittel pro Partnerteam.
- Dazu Pro Partnerteam ein kleines Handtuch („Gästehandtuch“) für die Aufgaben BP4/5 und ggf. auch als Hilfe für BP3. Das Handtuch können die Kinder mitbringen. Ggf. kann auch ein T-Shirt für BP6 mitgebracht werden.

**BP1** Die Kinder lesen die Anleitung, die sich mit auf dem Blatt *Aufgabenkarte* befindet.

**BP2** Im Video lernen die Kinder einen Programmablaufplan (PAP) mit der Basisstruktur *Sequenz* am Beispiel des Tischdeckens kennen.

**BP3** Die Kinder sehen im Video das Falten eines Handtuchs. Sie sollen in der *PAP-Vorlage* die Faltabfolge selbst aufschreiben. Der Arbeitsauftrag „Er soll möglichst kurz sein, aber er soll alles Wichtige enthalten.“ böte einen Anlass zur Reflexion – mit einem Partnerteam, wie auf der *Aufgabenkarte* vorgeschlagen, oder gemeinsam in der Klasse. Hintergrund der Aufgabenstellung ist, dass ein Aspekt von algorithmischem Denken die Fähigkeit der Reduktion komplexerer Vorgänge auf ihre Kernidee ist, die aber gleichsam noch präzise genug sein muss, damit sie eindeutig wiederholbar bleibt. Hier sind verschiedene Lösungsvarianten denkbar, wie bereits weiter oben ausführlich beschrieben wurde.

**BP4/5** Vor der Bearbeitung der Aufgaben BP4/ BP5 ist es sinnvoll, die Klasse in 2 Gruppen zu teilen, so dass eine Hälfte der Teams die Aufgabe BP4 bearbeiten kann, die andere Hälfte die Aufgabe BP5. In den beiden Videos werden zwei verschiedene, weitere Faltenarten für Handtücher dargestellt. Die Idee hinter diesem Aufgabenformat ist, dass die erstellten PAPs zwischen Teams beider Gruppen ausgetauscht werden, und zuerst versucht wird, nach dem PAP der anderen Gruppe ein Handtuch zu falten. Mit dem Video der Gegengruppe (das erst jetzt angesehen werden darf) kann dann überprüft werden, ob der PAP die zuvor in BP3 erarbeiteten Bedingungen erfüllt, also ob er zum Ziel führt. Wenngleich nicht auf der *Aufgabenkarte* erwähnt, kann eine Überarbeitung der PAPs sinnvoll sein. Ziel dieser Aufgaben ist es folglich, funktionierende Algorithmen (zum Zusammenlegen eines Handtuchs) in Form von PAPs zu formulieren und die aufgestellten Algorithmen zu validieren. Ein abschließendes gemeinsames Reflexionsgespräch wäre hier möglich.

**BP6** Im Video sind drei verschiedene Faltenarten eines T-Shirts zu sehen. Auf dem Arbeitsblatt BP6 sind dazu passend drei PAPs notiert. Die PAPs müssen also rekodiert werden, um eine Zuordnung zur Video-Faltung zu ermöglichen. Ggf. müssen dazu vorab Begrifflichkeiten wie „Kragen“ in den PAPs geklärt werden. Zur Begründung der Entscheidung können die Kinder in den PAPs entscheidende Stellen farblich markieren, etwa solche, die Indikatoren für eine bestimmte Version sein könnten. Auch hier ist neben der Besprechung zweier Teams auch eine gemeinsame Reflexion denkbar. Ggf. kann ein ‚echtes‘ T-Shirt bei der Rekodierung der PAPs helfen.

## BS - BASISMODUL TEIL 2 - WAS IST SCRATCH?

Vorzubereitendes Material aus *BS1.pdf* (verfügbar unter *promaprim.de* - Projektmaterialien oder unter *3yp.de*):

- Blatt *Aufgabenkarte* (mit Anleitung): 1x pro Partnerteam.
- Dazu ein digitales Arbeitsmittel pro Partnerteam.

- BS1** Die Kinder lesen die Anleitung, die sich mit auf dem Blatt *Aufgabenkarte* befindet. Darauf werden sie wiederum aufgefordert, sich das Video BS1 anzusehen, indem das Handling der digitalen Lernumgebung dargestellt wird. Eine zentrale Idee des Konzeptes ist hier, dass die Kinder über die Browser-Tabs zwischen Aufgabe, Erklärvideo und Scratch-Programmierungsumgebung (nicht immer sind alle drei Elemente vorhanden) hin- und herschalten können und das Tab-Handling beherrschen. Hier würde sich eine frontale Anleitung ergänzend zum Video gut anbieten. Im Video und zusammengefasst in der Anleitung erfolgen auch zwei Hinweise zum Scratch-Handling: Über ‚Reload‘ kann jederzeit die Scratch-Programmierungsumgebung auf den Ursprungszustand, so wie sie für die entsprechende Aufgabe vorbereitet wurde, neu geladen werden. Über die ‚Weltkugel‘ muss ggf. die Sprache wieder auf Deutsch gestellt werden, falls diese verstellt wurde. Natürlich dürfen die Kinder hier auch andere Sprachen ausprobieren, wenn sie beispielsweise eine andere Herkunftssprache haben.
- BS2** Im Video werden die zentralen Elemente der Scratch-Programmierungsumgebung vorgestellt. Hier wird auf jene Elemente fokussiert, welche die Kinder mindestens benötigen. Weitere Elemente (wie zum Beispiel das Anfertigen neuer Hintergrundbilder o.ä.) entdecken die Kinder in der Regel selbst.
- BS3** Während die erste Arbeitsanweisung reproduktiv ist (erstes Erstellen eines einfachen Scratch-Programms, das Tab-Umschalten zu Aufgabe und Erklärvideo hilft dabei), ist in der zweiten Arbeitsanweisung dann Kreativität gefragt, die sich möglicherweise auch über die gestellten Arbeitsanweisungen hinaus zeigen wird. Neben den Arbeitsanweisungen ist ein PAP dargestellt, auf den in der Arbeitsanweisung nicht näher eingegangen wird. Lehrkräfte könnten hier ein Unterrichtsgespräch einfügen, auch zum hier neuen PAP-Element *Ausgabe*.
- BS4** In dieser Aufgabe wird der Scratch-Block *frage...* eingeführt, zudem auch die Systemvariable *Antwort* gehört. In Reflexionsgesprächen könnte sowohl über die auf die Frage hin getätigte Eingabe, die in der Systemvariable *Antwort* gespeichert wird, gesprochen werden, als auch über das neue PAP-Element *Eingabe*.
- BS5** Diese Aufgabe ist freiwillig und dient zur Differenzierung, sie fängt unterschiedliche schnelle Bearbeitungen auf. Gegenstand sind Blöcke zur Animation, die hier kreativ und offen verwendet werden können. Inhaltlich sind animierend-gestalterische Programmelemente für die weiteren Programmieraktivitäten im Projekt ProMaPrim eher von untergeordneter Bedeutung, erfahrungsgemäß beschäftigen sich die Kinder damit aber auch ohne konkreten Auftrag, so dass hier der Impuls für kreative Gestaltungen *am Ende* von eigenen inhaltsbezogenen Programmieraktivitäten gegeben wird.

## BU - BASISMODUL TEIL 3 - UNTERSCHIEDUNGEN UND SCHLEIFEN

Vorzubereitendes Material aus *BU1.pdf* (verfügbar unter *promaprim.de* - Projektmaterialien oder unter *3yp.de*):

- Hälfte des Blattes *Aufgabenkarte*: 1x pro Partnerteam.
- Arbeitsblatt *BU4* (Sport-Programme): 1x pro Partnerteam, wird zerschnitten. Gerne in Farbe gedruckt, da in Scratch Schleifen auch in gelb dargestellt werden. Ein Graustufendruck ist aber auch einsetzbar.
- Arbeitsblatt *PAP-Vorlage*: ‚mehrfach‘ - diese Vorlage wird pro Partnerteam mindestens fünfmal benötigt und könnte als ‚Verbrauchsmaterial‘ bereitliegen, ggf. sind noch Reste aus BP vorhanden. Ggf. müssen auch zwei PAP-Vorlagen aneinandergeklebt werden, falls die Programme länger als fünf Kästchen werden.
- Arbeitsblatt *PAP-Vorlage*: ‚eine Hand voll‘ - diese Vorlage *kann* benutzt werden, wenn BP2 oder BP5f. bearbeitet werden, aber die Kinder können die Raute auch einzeichnen.
- Dazu ein digitales Arbeitsmittel pro Partnerteam.
- Ggf. können ‚Briefe‘ (Papier Din-A4-blanko) und Umschläge in DIN-C6 und DIN-lang (so heißt der breitere der beiden Umschläge im Video) zum Ausprobieren für BU2 hilfreich sein.

**BU1** PDF mit den Materialien, eine Anleitung ist nicht mehr erforderlich.

**BU2** Hier wird an den für die Umschläge in DIN-C6 und DIN-lang unterschiedlich notwendigen Faltungen eine Bedingung/ Entscheidung dargestellt. In die PAP-Vorlage sollen die Kinder also eine Verzweigung einbauen und beide Faltwege parallel darauf darstellen, wie im Video angedeutet. Neben der Doppel-Team-Reflexion kann auch eine gemeinsame Besprechung sinnvoll sein. Für die Unterrichtsvorbereitung ist es wichtig, die weiter oben gegebenen allgemeinen Hinweise zur Sprache in PAPs zu beachten.

**BU3/4** In Video BU3 wird zunächst die Schleife mit einer fest vorgegebenen Anzahl an Wiederholungen am Beispiel des Tischdeckens eingeführt. Eine denkbare weiterführende Differenzierungsaufgabe wäre hier, eine andere Deckvariante (erst 4 Teller, dann 4 Gläser, ...) als PAP darzustellen. Im Video BU4 werden zunächst 3 Sportübungen dargestellt, welche dann jeweils einzeln mehrfach wiederholt werden, und als PAP dargestellt werden sollen. Dazu können die Kinder die Elemente des Arbeitsblattes BU4 ausschneiden und auf leeren PAP-Vorlagen arrangieren. Freiwillige Arbeitsanweisung 4: Im letzten Teil des Videos wird eine geschachtelte Schleife dargestellt – die 3 jeweils aus Wiederholungen bestehenden Sportprogramme werden verkettet und insgesamt wiederholt. Dazu können die Kinder die PAPs der drei Einzelprogramme auch aneinanderkleben und umarrangieren. Auf Grund der Komplexität werden nicht alle Kinder zu dieser Aufgabe kommen.

**BU5-9** Die Schleife mit (Abbruch-)Bedingung stellt eine Kombination aus den zuvor erarbeiteten Elementen Bedingung/ Entscheidung und Schleife dar. Dazu wird in Video BU5 die Idee des „wiederhole... bis...“ am Beispiel des Glaseinschüttens eingeführt. Die vier Videos BU6 bis BU9 enthalten alle diese Idee und sind zur Auswahl gedacht, von denen die Kinder (mindestens) eines als PAP darstellen sollen. Es bietet sich an, in einem Reflexionsgespräch die Grundidee der Schleife mit (Abbruch-)Bedingung („wiederhole... bis...“ – so heißt später dann auch der Scratch-Block) zu thematisieren, und zu präzisieren, dass immer *vor* einem weiteren (ggf. auch *vor* dem ersten) Schleifendurchlauf die Bedingung überprüft wird.



## BZ - BASISMODUL TEIL 4 - ZAHLEN UND ZÄHLEN

Vorzubereitendes Material aus *BZ1.pdf* (verfügbar unter *promaprim.de* - Projektmaterialien oder unter *3yp.de*):

- Hälfte des Blattes *Aufgabenkarte*: 1x pro Partnerteam.
- Dazu ein digitales Arbeitsmittel pro Partnerteam.

**BZ1** PDF mit der Aufgabenkarte, eine Anleitung nicht mehr erforderlich.

**BZ2** Im Video lernen die Kinder eine Variable kennen, wie man ihr einen Wert zuweist, und wie man diesen Wert verwendet. Neben dem Video kann der abgebildete PAP helfen, ein entsprechendes Scratch-Programm zu erstellen. Die Variable *Robos Alter* ist in der Scratch-Programmierungsumgebung (wie auch im Folgende, wenn Variablen benötigt werden) schon angelegt, so dass die Kinder keine eigene Variablendeklaration vornehmen müssen. Trotzdem kann es im Laufe des Lernprozesses irgendwann sinnvoll sein, dass die Lehrkraft das Erstellen eigener Variablen erklärt, damit die Kinder diese für eigene Projekte nutzen können.

**BZ3** Eine der Grundideen des Projektes ProMaPrim ist es, die Kinder immer zuerst ‚unplugged‘ – also ohne PAPs und Scratch – die zugrundeliegende Arithmetik oder Algorithmik nachvollziehen zu lassen, hier, indem sie im Kopfrechnen, später auch, indem sie mit Papier und Bleistift Zahlenwerte/ Rechnungen notieren. Nachdem sie sich dann mit PAPs und Scratch mit dem mathematischen Inhalt beschäftigt haben, bekommen sie in der Regel einen Auftrag, sich wieder mathematisch mit den Ergebnissen zu beschäftigen. Dies wird als projekttypischer ProMaPrim-Dreischritt bezeichnet. Hier begegnen die Kinder diesem Dreischritt zum ersten Mal in ganz einfacher Form: 1) Im Kopf rechnen ‚361 Jahre minus Alter des Kindes‘, 2) das Programm erstellen und die Rechnung digital durchführen lassen, 3) die Rechnung(en) wieder im Kopf überprüfen. In einer Reflexion könnte daher das Einhalten dieser drei Schritte thematisiert werden. Neu ist in diesem Programmierauftrag die Verwendung eines Operators, hier Minusrechnen. Der Arbeitsauftrag 4 stellt ein differenzierendes Zusatzangebot dar, hier wird eine dritte Variable benutzt und die Rechnung vor der Ausgabe durchgeführt.

**BZ4** Im Video zu BZ4 wird zunächst der ‚eigene Block‘ (so heißen in Scratch selbstdefinierte Funktionen, das entsprechende Unterprogramm versteckt sich unterhalb der bereitliegenden Blöcke) ‚rede‘ vorgestellt. Dieser wird in ProMaPrim immer dann verwendet, wenn sukzessive Ausgaben (also das schrittweise Füllen der Sprechblase) notwendig sind, da dies in Scratch sonst nur auf Umwegen möglich wäre. Die Aufgabe BZ4 stellt eine erste offenere Programmierherausforderung für die Kinder dar. Wichtig ist auch hier wiederum der ProMaPrim-Dreischritt: Zuerst mit Papier und Bleistift auf der Aufgabenkarte ‚zählen‘, dann Programmieren, dann überprüfen (stimmen Start-, Ausgabe- und Endwerte genau mit der Zählweise auf der Karte überein?). Der abgebildete PAP, auf den verwiesen wird, verkörpert zwar die arithmetische Struktur des Zählens, enthält aber bewusst keinen konkreten Wert für den Startwert der Zählzahl, dieser muss experimentell beim Erstellen des Programms ermittelt werden. Die Programmierungsumgebung in Scratch ist schon weit vorbereitet, so dass die Kinder auf die zentralen Stellen des Zählalgorithmus fokussieren und die bereitliegenden Elemente in eine sinnvolle Reihenfolge bringen. Dies geschieht durch ‚konstruierendes Programmieren‘: Die Kinder arrangieren und manipulieren den Code so lange, bis die Ausgabe dem gewünschten Ergebnis entspricht. Dabei kann es durchaus verschiedene Lösungen geben – je nach Reihenfolge der Blöcke ‚setze Zählzahl auf Zählzahl + 1‘ und ‚rede Zählzahl‘ müssen der Startwert und die Abbruchbedingung anders moduliert werden. In der Erprobung des Moduls kam es zudem dazu, dass die Kinder auch außerhalb der Schleife Elemente hinzufügten, um das gewünschte Ergebnis zu erhalten (Beispiel: Robo zählt nur bis 9 – hinter die Schleife kommt ‚rede (10)‘). Diese Lösungen funktionieren aber häufig nur für die konkrete Zielzahl, nicht für eine beliebige. Darum ist in Arbeitsauftrag 4 auch das Überprüfen einer weiteren Zielzahl vorgegeben.

Sowohl unterschiedliche als auch (nicht) allgemeingültige Lösungen könnten hier Gegenstand einer gemeinsamen Reflexion sein, ebenso der Lösungsprozess als solcher.

- BZ5** Diese Aufgabe stellt eine Transferaufgabe von BZ4 dar. Sollte dort vorwärts gezählt werden, soll nun rückwärts („Countdown“) gezählt werden, unter Rückgriff auf die zuvor erstellte Programmstruktur und ggf. notwendigen Adaptionen. Darum wird auch am Ende von BZ4 angeregt, das fertige Programm nicht zu schließen. Es kann aber auch schnell aus der Erinnerung wiederhergestellt werden, oder direkt mit der Startumgebung BZ5 gearbeitet werden, die identisch mit der von BZ4 ist. Erneut ist der ProMaPrim-Dreischritt durchzuführen, und wieder gibt der bereitgestellte PAP nur Anregungen, aber keine konkrete Lösung, die wieder im konstruierenden Programmieren gefunden werden soll. Die Aufgaben BZ4 und BZ5 können als arithmetisch-algorithmischer Kern des Basismoduls und der Einführung des konstruierenden Programmierens gesehen werden. Dieses Vorgehen wird auch in den späteren Mathe-Modulen immer wieder aufgegriffen und ermöglicht entdeckendes Lernen im Bereich des algorithmischen Denkens. Daher ist es wichtig, die Kinder in BZ4/ BZ5 nicht zu früh anzuleiten, sondern probieren zu lassen, auch Umwege zuzulassen, und nur mit gezielten, sparsamen Impulsen Anregungen zu geben.
- BZ6** Zum Abschluss des Basismoduls programmieren die Kinder einfaches Spiel. Auf der Aufgabenkarte werden vorab die Symbole kleiner, gleich und größer wiederholt. Die Programmierumgebung ist erneut weit vorbereitet, thematisiert wird hier noch einmal die Bedingung/ Entscheidung. Zudem wird der Operator *Zufallszahl* eingeführt, der in späteren Mathe-Modulen eine Rolle spielen wird. Erfahrungsgemäß schmücken die Kinder das Rumpfprogramm noch aus, so dass hier ‚Ende offen‘ zum Abschluss des Basismoduls gelten kann.



## HINWEISE ZUM COPYRIGHT

Einige Materialien enthalten Screenshot von Scratch/CC-BY-SA-4.0 Creative Commons. Scratch ist ein Projekt der Scratch Foundation und der Lifelong Kindergarten Group am MIT Media Lab. Es steht kostenlos unter <https://scratch.mit.edu> zur Verfügung.

**Urheber dieses Kurses ist das Projekt ProMaPrim, vertreten durch Dr. Ulrich Schwätzer, Universität Duisburg-Essen, Didaktik der Mathematik, Thea-Leymann-Strasse 9, D-45127 Essen. Die Urheberrechte an diesem Programmierkurs stehen unter der Creative Commons Lizenz CC BY-NC-SA 4.0 (Namensnennung - Nicht-kommerziell - Weitergabe unter gleichen Bedingungen 4.0 international). Sie dürfen:**

- Teilen — das Material in jedwedem Format oder Medium vervielfältigen und weiterverbreiten
- Bearbeiten — das Material remixen, verändern und darauf aufbauen

Unter folgenden Bedingungen:

- Namensnennung — Sie müssen angemessene Urheber- und Rechteangaben machen, einen Link zur Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden. Diese Angaben dürfen in jeder angemessenen Art und Weise gemacht werden, allerdings nicht so, dass der Eindruck entsteht, der Lizenzgeber unterstütze gerade Sie oder Ihre Nutzung besonders.
- Nicht kommerziell — Sie dürfen das Material nicht für kommerzielle Zwecke nutzen.
- Weitergabe unter gleichen Bedingungen — Wenn Sie das Material remixen, verändern oder anderweitig direkt darauf aufbauen, dürfen Sie Ihre Beiträge nur unter derselben Lizenz wie das Original verbreiten.
- Keine weiteren Einschränkungen — Sie dürfen keine zusätzlichen Klauseln oder technische Verfahren einsetzen, die anderen rechtlich irgendetwas untersagen, was die Lizenz erlaubt.

Das ProMaPrim Icon wurde erstellt unter Verwendung des Icons pc.png von Freepik auf [www.flaticon.com](http://www.flaticon.com) (freie Lizenz bei Namensnennung).