

Schriftlicher Unterrichtsentwurf zum Programmierprogramm Scratch

Johanna Mehlkop

Matr.-Nummer: [REDACTED]

Master of Education (G)

Germanistik, Mathematik, Sachunterricht, Bildungswissenschaften

2. Semester

Schriftlicher Unterrichtsentwurf am Arbeitsbereich Didaktik der Informatik der WWU Münster¹

Erstellt von:	<input type="text" value="Johanna Mehlkop"/>
Matrikelnummer:	<input type="text" value="██████"/>
Zeitumfang (Min.):	<input type="text" value="90"/>
Klasse:	<input type="text" value="4"/>
Thema der Stunde:	<input type="text" value="Wir programmieren die Bewegungen einer Figur durch Scratch."/>
Thema der Reihe:	<input type="text" value="Programmieren mit Scratch!"/>

¹ Diese Vorlage basiert auf dem Dokument Schriftliche Arbeit mit Kommentar (Stand 03/2013) des Zentrums für schulpraktische Lehrerbildung Krefeld (ZfsL), Seminar für das Lehramt an Gymnasien und Gesamtschulen.

Inhaltsverzeichnis

Schriftliche Planung des Unterrichts

- | | |
|--------------------------------------|-------|
| 1. Ziele und angestrebte Kompetenzen | S. 1 |
| 2. Didaktische Schwerpunkte | S. 4 |
| 3. Artikulationsschema | S. 10 |

Literaturverzeichnis S. I

Anhang

- | | |
|--------------------------------------|----------|
| I. Tafelanschriften | S. III |
| II. Karten zur Studententransparenz | S. III |
| III. Helfersystem | S. IV |
| IV. Arbeitsheft | S. V |
| V. Tippkarten | S. XVIII |
| VI. Beispiel Programmierung | S. XIX |
| VII. Erwartungshorizont: Arbeitsheft | S. XX |

Versicherung und Verwertungsrechte S. XXI

Schriftliche Planung des Unterrichts

1. Ziele und angestrebte Kompetenzen

Ein Stundenziel/Kernanliegen mit Indikator:

Die Schülerinnen und Schüler sollen

- verschiedene Möglichkeiten zur Programmierung einer bewegenden Figur in Scratch anwenden und erklären können.

Sie zeigen dies, indem sie

- eine Möglichkeit der Programmierung selber entdecken, anwenden und den anderen SuS erläutern.
- eine weitere Möglichkeit selber entdecken oder in den Ausführungen ihrer Mitschüler² nachvollziehen.

Drei bis fünf Teilziele mit Indikator:

Die SuS sollen

- ihr bereits vorhandenes Wissen zu der Programmierung von Bewegungen in Scratch **reflektieren** und **aktivieren** können.

Sie zeigen dies, indem sie

- zu den Aussagen der SuS aus der Nachbarklasse begründet Stellung nehmen.

Die SuS sollen

- ihr vorhandenes theoretisches Wissen über die Programmierung in Scratch aus der vorherigen Stunde **anwenden** können.

Sie zeigen dies, indem sie

- bekannte Blöcke aus der vorherigen Stunde in ihre Programmierung einbauen.
- alleine oder in Partnerarbeit eine erfolgreiche Programmierung vornehmen.

Die SuS sollen

- ihre eigenen Aufgabenlösungen **kritisch reflektieren**.

Sie zeigen dies, indem sie

- gute Aspekte ihrer Programmierung herausstellen.
- zu verbessernde Aspekte anführen.

Die SuS sollen

- ihre eigene Programmierung **erläutern** können.

Sie zeigen dies, indem sie

- ihren Mitschülern ihre Programmierung anhand ihrer Aufzeichnungen und/ oder mithilfe ihres Programms am Computer erklären.

Die SuS sollen

- die zwei unterschiedlichen Wege zur Lösung der Aufgabe **nachvollziehen** und **anwenden** können.

² Aus Gründen der besseren Lesbarkeit wird auf die gleichzeitige Verwendung männlicher und weiblicher Sprachformen verzichtet. Sämtliche Personenbezeichnungen gelten gleichwohl für beide Geschlechter.

Sie zeigen dies, indem sie

- zwei unterschiedliche Möglichkeiten im Unterrichtsgespräch benennen und/ oder den Ausführungen ihrer Mitschüler folgen und diese ggf. ergänzen können.
- einen dieser Wege in der Hausaufgabe anwenden.

Hierdurch sollen folgende Kompetenzen laut Lehrplan/Empfehlungen gefördert werden (je eine inhaltsbezogene und prozessbezogene Kompetenz):

Inhaltsbezogene Kompetenz: Sprache und Automaten [GI17]

Die SuS verwenden Sprache zum Programmieren, indem sie die Programmiersprache von Scratch dazu verwenden, eine Figur am Computer durch ein Labyrinth zu steuern.

Prozessbezogene Kompetenz: Problemlösen/ kreativ sein [MSW08]

Die SuS bearbeiten die vorgegebene Problemstellung (Wie kommt der Affe an seine Bananen?). Dazu stellen sie zunächst eine Planung mithilfe ihres vorhandenen Wissens und der Tippseiten in ihrem Arbeitsheft auf. Anschließend probieren sie diese Planung aus und prüfen und reflektieren ihre Lösung und ihren Lösungsweg, indem sie Fehler finden und korrigieren und gelungene Aspekte hervorheben. Im Rahmen der Hausaufgabe übertragen sie ihren Lösungsweg bzw. den ihrer Mitschüler auf ähnliche Sachverhalte.

Der vorliegende Unterrichtsentwurf wurde für eine vierte Grundschulklasse konzipiert und umfasst eine Doppelstunde. Diese Doppelstunde stellt die zweite Unterrichtsstunde der Unterrichtsreihe *Programmieren mit Scratch!* dar. In dieser Unterrichtsreihe sollen die Schülerinnen und Schüler³ die Fähigkeit erwerben, selbständig Spiele und Projekte mit Scratch zu programmieren, um so einen ersten Einstieg in die Programmierung und in Programmiersprachen zu erhalten. In dieser zweiten Unterrichtsstunde, *Programmieren einer Figur durch ein Labyrinth*, lautet das Stundenziel folgenderweise: Die SuS sollen verschiedene Möglichkeiten zur Programmierung einer bewegenden Figur in Scratch anwenden und erklären können. Sie zeigen dies, indem sie eine Möglichkeit der Programmierung selber entdecken, anwenden und den anderen SuS (in der Gruppenarbeit oder im Klassengespräch) erläutern. Außerdem zeigen sie dies, indem sie eine weitere Möglichkeit der Programmierung entweder selber entdecken oder durch die Ausführungen ihrer Mitschüler nachvollziehen. Ferner sind fünf Teilziele definiert, die die SuS im Laufe des Unterrichts erwerben sollen: Die SuS sollen ihr bereits vorhandenes Wissen zu der Programmierung von Bewegungen in Scratch reflektieren und aktivieren können. Dies zeigen sie, indem sie zu den Aussagen der SuS aus der Nachbarklasse begründet Stellung nehmen (1). Die SuS sollen ihr vorhandenes theoretisches Wissen über die Programmierung in Scratch aus der vorherigen Stunde anwenden können. Die Erreichung dieses Ziels zeigen sie, indem sie bekannte Blöcke aus der vorherigen Stunde in ihre Programmierung einbauen und alleine oder in Partnerarbeit eine erfolgreiche Programmierung vornehmen (2). Die SuS sollen ihre eigenen Aufgabenlösungen kritisch reflektieren. Sie zeigen dies, indem sie gute Aspekte ihrer Programmierung herausstellen und zu verbessernde Aspekte anführen (3). Die SuS sollen ihre eigene Programmierung erläutern können. Dies zeigen sie, indem sie ihren Mitschülern ihre Programmierung anhand ihrer

³ Im Folgenden werden Schülerinnen und Schüler mit SuS abgekürzt.

Aufzeichnung und/ oder mithilfe ihres Programms am Computer erklären (4). Die SuS sollen die zwei unterschiedlichen Wege zur Lösung der Aufgabe nachvollziehen und anwenden können. Sie zeigen dies, indem sie zwei unterschiedliche Möglichkeiten im Unterrichtsgespräch benennen und/ oder den Ausführungen ihrer Mitschüler folgen und ggf. ergänzen können und indem sie eine dieser Möglichkeiten in der Hausaufgabe anwenden (5). Nach den Bildungsstandards Informatik für den Primarbereich [GI17] ergibt sich für die inhaltsbezogene Kompetenz, die im Rahmen dieser Unterrichtsstunde gefördert werden soll, folgende Erwartung am Ende der Klasse vier: Die SuS verwenden Sprache zum Programmieren. Diese Kompetenzerwartung fällt in den Bereich *Sprachen und Automaten* und wird dadurch gefördert, dass die SuS die Programmiersprache von Scratch dazu verwenden eine Figur (Affe) am Computer durch ein Labyrinth zu bewegen. Sie wenden also Programmiersprache an, um die gegebene Problemstellung, Wie kommt der Affe an seine Bananen?, zu lösen. Die Lösung einer Problemstellung fördert nach dem Lehrplan Mathematik für die Grundschulen in NRW [MSW08] die prozessbezogene Kompetenz *Problemlösen/ kreativ sein*. Die SuS stellen dazu zunächst eine Planung mithilfe ihres vorhandenen Wissens und der Tippseiten in ihrem Arbeitsheft auf. Anschließend probieren sie diese Planung aus und prüfen und reflektieren ihre Lösung und ihren Lösungsweg, indem sie Fehler finden und korrigieren und gelungene Aspekte hervorheben. Im Rahmen der Hausaufgabe übertragen sie ihren Lösungsweg bzw. den Lösungsweg ihrer Mitschüler dann abschließend auf einen ähnlichen Sachverhalt.

2. Didaktische Schwerpunkte

In der vorliegenden Darstellung der didaktischen Schwerpunkte werden die einzelnen Elemente der Unterrichtsstunde für eine vierte Klasse untersucht und didaktisch begründet. Die Elemente des Unterrichts umfassen dabei die während der Planung des Unterrichts getroffenen Ziel-, Inhalts-, und Methodenentscheidungen, die, im Sinne des Didaktischen Rekonstruktionsmodells nach Kattmann [Ka07], auf Grundlage und in Wechselbeziehung zur fachlichen Klärung und zur Lernerperspektiven getroffen worden sind. Deshalb erfolgt im Rahmen dieser Arbeit zunächst sowohl eine Analyse der Lernausgangslage der SuS, als auch eine Auseinandersetzung mit dem Unterrichtsgegenstand *Programmieren mit Scratch*.

Für die vorliegende allgemeine Unterrichtskonzeption, die sich an keine explizite vierte Klasse richtet, werden zunächst grundlegende Fähigkeiten in Schrift und Sprache, die ein Lesen und schriftliches Bearbeiten von Aufgabenstellungen ermöglichen, vorausgesetzt. Außerdem müssen die SuS weitere Voraussetzungen erfüllen, um erfolgreich an diesem Unterricht partizipieren zu können. Die geplante Doppelstunde ist Teil einer Reihe, die das Programmieren mit Scratch behandelt und in einem über den Zeitraum von zehn Wochen wöchentlich stattfindenden Informatik-Projekt umgesetzt wird. Einer der Schwerpunkte dieses Projektes ist die eigene Programmierung eines Spiels mit Scratch, die u.a. in dieser Stunde vorbereitet wird. Für Programmierungen mit Scratch werden sowohl motorische Fähigkeiten (z. B. bei der Benutzung einer Maus), als auch ein erprobter Umgang mit dem Computer erfordert. Es wird deshalb vorausgesetzt, dass die Klasse schon mit dem Arbeiten am Computer vertraut ist und vor allem Kompetenzen im Bereich der Informatiksysteme der Bildungsstandards Informatik [GI17] bereits erworben hat. Vor dieser Unterrichtsstunde haben sich die SuS bereits mit dem Begriff des Programmierens beschäftigt, indem sie diesen u.a. enaktiv erlebt haben (Kinder haben sich gegenseitig mithilfe von vorher festgelegten Befehlen durch den Raum geführt, wobei eines der Kinder die Augen verbunden hatte). Außerdem haben sie bereits in einer Doppelstunde zu dem Programm Scratch gearbeitet. In dieser Stunde haben sie erfahren, wie Scratch gestartet wird, haben einen eigenen Zugang erhalten und erste Erfahrungen mit Scratch gesammelt. Bei diesem ersten Kennenlernen von Scratch sind ihnen zunächst die großen Codeblockbereiche und deren wichtigsten Blöcke vorgestellt worden. Anschließend haben sie diese praktisch erprobt, indem sie die Figur mit jeweils einem Block programmiert und die Auswirkungen dieser Programmierung beobachtet haben. Die Programmierung von ganzen Bewegungsabläufen ist ihnen also noch nicht bekannt. Für den Unterricht ist außerdem wichtig, dass die SuS mit dem Konzept des selbstregulierten Lernens nach Schmitz [Sc01] vertraut sind und im Unterricht fortlaufend geschult werden, dieses anzuwenden. Das heißt, es ist ihnen bewusst, dass ein Lern- und Arbeitsprozess sowohl eine Planungs-, als auch eine Durchführungs- und eine Reflexionsphase umfasst und deshalb eine dreischrittige Bearbeitung von Aufgaben notwendig ist. Des Weiteren wird vorausgesetzt, dass den SuS sowohl die Partner-, als auch die Gruppen- und Einzelarbeit als Methode bekannt und ihnen die notwendigen Verhaltensweisen vertraut sind. Da auch hier mit einer heterogenen Lerngruppe zu rechnen ist, ist ein umfassendes Helfersystem in der Klasse integriert, das auch in dieser Stunde Anwendung findet.

Lehr- und
Lernausgangslage
der SuS

So finden die SuS zum einen Tippkarten vor, auf die sie zurückgreifen können, wenn sie bei der Bearbeitung der Aufgabe nicht weiter kommen. Zum anderen können sie sich Hilfe bei ihren Mitschülern holen, indem sie entweder mit ihnen in Form einer Partnerarbeit zusammenarbeiten oder indem sie eine Person ansprechen, die bereits fertig ist und ihren Namen an den Helferbaum (siehe Anhang III) geklemmt hat. Dieses Helfersystem erfüllt dabei verschiedene Aufgaben: Es soll die SuS sowohl dazu befähigen, die Aufgabe zu lösen, als auch ihre Lernbereitschaft hochhalten. Denn zunächst ist davon auszugehen, dass diese Bereitschaft zum Lernen bei allen SuS vorhanden ist, da das Ziel, ein eigenes Spiel zu programmieren, für SuS sehr reizvoll ist. Außerdem arbeiten SuS sehr gerne mit Computern und digitalen Medien, da dieses Wissen in ihren Alltag schnell Anwendung findet [AZ17].

Zur Unterstützung der Aufrechterhaltung dieser Lernbereitschaft ist Scratch als Programmierprogramm gewählt worden. Scratch ist eine von der Gruppe „Lifelong Kindergarten“ des MIT Media Lab entwickelte kostenlose Programmierumgebung, die sich an Kinder zwischen acht und 16 Jahren richtet. Ziel dieses Programms ist es, dass Nutzer Fähigkeiten zur Programmierung und zur Problemlösung auf einfache Weise lernen und weiterentwickeln [Sw17]. Es ist somit sowohl von der Komplexität, als auch von der Zielstellung für Programmierungen mit SuS einer vierten Klasse gut geeignet. Damit SuS jedoch Spiele in Scratch programmieren können, benötigen sie verschiedene Algorithmen, die für die Kinder als Codeblöcke oder Skripte bezeichnet werden. Diese Skripte bestehen aus verschiedenen algorithmischen Grundbausteinen, vereinfacht für die SuS als Blöcke bezeichnet, die die Bewegung und das Aussehen einer Figur, das Fühlen von Figuren, das Erzeugen verschiedener Klänge und Grafiken, das Einfügen einer zweiten Figur und die Kommunikation verschiedener Figuren untereinander steuern. Durch die Erzeugung solcher Algorithmen, die mithilfe einer Maus zusammengebaut werden, im Zusammenhang mit einer geeigneten Bühne (Hintergrund) ist es möglich, sowohl größere als auch kleinere Spiele zu erstellen [Sw17]. Damit die SuS im Rahmen dieser Unterrichtsreihe am Ende selbst ein Spiel programmieren können, ist es Ziel dieser zweiten Doppelstunde mit den SuS verschiedene Möglichkeiten zu erarbeiten, eine Figur, zunächst durch ein Labyrinth, zu bewegen. Hierbei werden vor allem zwei Möglichkeiten thematisiert, die die SuS entweder selbst erarbeiten oder durch ihre Mitschüler erlernen. Begonnen wird bei beiden Möglichkeiten mit dem Block *Wenn (Fahne) angeklickt*. Die erste Möglichkeit basiert auf dem Projekt *Maus zum Käse*, das die RWTH Aachen zusammen mit anderen Scratch-Projekten in Grundschulen durchführt. Bei dieser Möglichkeit verwenden die SuS nur Blöcke der blauen Gruppe *Bewegung*, indem sie ihre Figuren nur durch das hintereinanderschalten der Anweisungen *gehe (10)er Schritt, drehe dich -> um (15) Grad* und *drehe dich <- um (15) Grad* steuern. Dabei können die Anzahl der Schritte und die Höhe der Gradzahl variiert werden [En16]. Die zweite thematisierte Möglichkeit, die auf Sweigart [Sw17] basiert, ist deutlich komplexer als die erste und wird zunächst nur von wenigen SuS erwartet. Sie umfasst sowohl Blöcke des Bereichs *Bewegung*, als auch *Steuerung* und *Fühlen* und zielt darauf ab, die Pfeiltasten so zu programmieren, dass damit die Figur in verschiedenen Richtungen bewegt werden kann. Dazu muss jede Taste so programmiert werden, dass beim Drücken dieser Taste, sich entweder *x* oder *y* um einen bestimmten Wert verändert. Die Überprüfung ob dies geschieht muss

Begrenzte
Sachanalyse
des U.-
gegenstandes

dabei fortlaufend erfolgen (Beispiel: siehe Anhang VII).

Im Rahmen der Unterrichtsstunde wird erwartet, dass die Programmierungen der SuS nicht genau den beschriebenen Möglichkeiten entsprechen, sich aber auf die zwei verschiedenen beschriebenen Grundideen zurückführen lassen. Die Programmierungen also entweder darauf abzielen, die Figur direkt zum Ziel zu programmieren oder die Figur mit den Pfeiltasten selbst zu bewegen. In dieser Stunde lernen die SuS also zwei Möglichkeiten kennen, in einem zukünftig programmierten Spiel ihre eingesetzten Figuren zu bewegen. Dabei wird durch die Vorbereitung der Bühne (Labyrinth) und der notwendigen Figuren (Affe und Bananen) die Komplexität für die SuS so reduziert, dass sie sich auf die Programmierung der Bewegungen des Affen konzentrieren können. Außerdem stellt das verwendete Labyrinth ein Beispiel für ein einfaches, in Scratch programmiertes Spiel dar. Die SuS erfahren deshalb bei der Bearbeitung der Aufgabe und der Hausaufgabe implizit, dass die Programmierung eines Spiels in Scratch nur wenige Komponenten umfasst. Zudem wird ihnen deutlich, dass durch den Einsatz von neuen Hintergründen neue Varianten entstehen, die später als Spiellevel genutzt werden können. Die Bearbeitung dieser Aufgabe kann neben dem Gewinn dieser Erfahrungen auch zu einer erneuten Erhöhung der Lernbereitschaft führen: Denn durch die Bearbeitung eines Spiels wird der Spaßfaktor für SuS deutlich erhöht, da u. a. die Lösung einer Aufgabe direkt sichtbar und die Erreichung des Reihenziels, die eigene Programmierung eines Spiels, für die SuS realistischer wird. Insgesamt dient die Stunde inhaltlich dazu, den SuS die Kompetenz zu vermitteln, Figuren so zu programmieren, dass sie sich dem eigenen Ziel entsprechend bewegen.

Die Stunde fördert außerdem sowohl inhaltliche, als auch prozessbezogene Kompetenzen, die mit den Bildungsstandards Informatik für den Primarbereich [GI17] und dem Lehrplan NRW [MSW08] einhergehen. Die Bildungsstandards Informatik für den Primarbereich fordern für SuS am Ende der Jahrgangsstufe vier u.a., dass sie zum Programmieren Sprache verwenden und Algorithmen entwerfen und realisieren [GI17]. Dazu wird im konzipierten Unterricht durch die Erarbeitung exemplarischer Codeblöcke eine Form der Programmiersprache erarbeitet, die von den SuS als Sprache zum Programmieren in Scratch verwendet werden kann und wird. Mithilfe dieser Sprache formulieren die SuS Algorithmen für die Figur des Affen, die dessen Bewegungen durch ein Labyrinth steuern sollen. Insgesamt bearbeiten sie die Problemstellung also indem sie mithilfe der in Scratch verwendeten Programmiersprache kleine Algorithmen entwerfen.

Ferner wird durch die Bildungsstandards, aber vor allem durch den Lehrplan gefordert, dass die SuS möglichst problemlösend und entdeckend arbeiten. Eine Möglichkeit der richtigen Programmierung des Affen zur Lösung der Problemstellung (Wie kommt der Affe an seine Bananen?) kann dabei selbstständig von den SuS entdeckt werden. Dazu müssen sie – im Sinne der Kompetenz *problemlösen/ kreativ sein* – mithilfe ihres vorhandenen Wissens eine Planung des Problems aufstellen, diese Planung ausprobieren und anschließend ihren Lösungsweg reflektieren. Dieses Vorgehen erfordert von den SuS, neben der Kompetenz zum selbstregulierten Lernen, Kreativität, was sich wiederum positiv auf das Selbstbild der Kinder im Bereich der Informatik auswirkt und somit eine Forderung der Bildungsstandards erfüllt. Zudem werden durch die natürliche Differenzierung der Aufgabe (Kapitel 2.5) Differenzierungsmöglichkeiten gewährleis-

Legitimation
des Vorhabens
durch
curriculare
Vorgaben

tet [GI17] [MSW08]. Des Weiteren wird durch die Unterrichtsstunde eine grundlegende informatische Bildung gefördert, die anschlussfähig für den späteren Informatikunterricht ist. So wird beispielsweise im Lehrplan für die Sekundarstufe eins gefordert, dass SuS zur Darstellung von Problemlösungen Programmiersprache verwenden und Algorithmen in eine Programmiersprache umsetzen. Dafür stellt das Programmieren mit Scratch eine gute Übung dar [MSW15].

Die hohe Zukunftsbedeutung des Themas *Programmieren mit Scratch* ergibt sich für die Kinder aber nicht nur aufgrund des Nutzens für ihre zukünftige Schullaufbahn, sondern auch aufgrund der aktuellen Entwicklungen der heutigen Zeit. So sind digitale Medien mittlerweile in fast allen Haushalten vertreten und nicht mehr aus der Lebenswelt der Kinder wegzudenken. Deshalb ist es besonders wichtig, SuS bei ihren Erfahrungen mit Computern, Handys und Tablets gut zu begleiten und ihnen eine Medienkompetenz zu vermitteln, die sie nicht nur dazu befähigt, Geräte zu bedienen, sondern ihnen auch durch z. B. einfaches Programmieren „einen ersten Eindruck, wie zum Beispiel Maschinen funktionieren“ [AZ17, S. 10] vermittelt. Besonders wichtig ist es dabei, dass SuS die grundlegenden Prinzipien und Funktionsweisen der digitalen Welt „nicht nur erkennen, verstehen und reflektieren, sondern auch nutzen können“ [AZ17, S.10]. Dafür kann die Beschäftigung mit der Programmierumgebung Scratch einen großen Beitrag leisten. Denn es bietet aufgrund seiner einfachen, kindgerechten und motivierenden Gestaltung zahlreiche Möglichkeiten für spielerische und motivieren Unterrichtsthemen, die den SuS erste Erkenntnisse im Bereich der Programmierung, der Programmiersprachen und der Funktionen von Algorithmen erlauben. Des Weiteren bietet die Beschäftigung mit dem Thema eine Möglichkeit zur Herstellung von Chancengleichheit. Denn es ist zwar den meisten SuS möglich, digitale Medien zu nutzen, aber längst nicht alle haben die Chance, den Nutzen von Informatiksystemen als Arbeits- und Lernumgebung zu erfahren. Deshalb dient dieser Unterricht auch dazu, allen SuS die Erfahrungen mit Computern zu ermöglichen und ihnen erste Programmiererfahrungen zu ermöglichen [GI17].

Relevanz für die SuS

Die Aktualität des Themas in der Lebenswelt der SuS, die Zukunftsbedeutung für das Leben und Lernen der SuS und die Forderungen der Bildungsstandards sowie des Lehrplans rechtfertigen das Behandeln von Programmierungen in Scratch in der vierten Klasse der Grundschule.

Da der vorliegende Unterricht an die erste Doppelstunde der Reihe *Wir programmieren mit Scratch* anknüpft, erfolgt in der Einstiegsphase eine kurze Wiederholung der wichtigsten bisher gewonnenen Erkenntnisse. Dazu wird eine kurze Geschichte genutzt, um die SuS zu motivieren und sie zum Diskutieren anzuregen. Bei dieser Geschichte hat die Lehrkraft ein Gespräch zweier SuS der Nachbarklasse mitbekommen und bei ihnen zwei Aussagen aufgeschnappt, die sie den SuS mitbringt. Diese Aussagen werden zusätzlich zur Äußerung an der Tafel visualisiert und unterstützen so, im Sinne einer inhaltlich strukturierende Maßnahme nach Möller [Mö16], alle SuS darin, dem Unterricht zu folgen, an ihm teilzunehmen und so das Lernziel der Stunde zu erreichen. Durch die Nutzung der Aussagen zur Anregung des Austausches unter den Lernenden wird eine kognitive Aktivierung der SuS zu Beginn der Stunde erreicht und eine Anwendung des Vorwissens ermöglicht [Mö16]. Nach diesem Austausch erfolgt

Begründung der wichtigsten Entscheidungen des geplanten Unterrichts

eine Phase, die der Hinführung zum Thema dient und in der das zu bearbeitende Projekt vorgestellt wird. Auch die Erklärung des Projektes wird mithilfe einer Veranschaulichung, der ersten Seite des Arbeitsheftes, unterstützt. So soll die Motivation aller SuS geweckt werden, das Projekt und die Problemstellung zu bearbeiten. Zusätzlich wird in dieser Phase das Bearbeitungsschema *Planen – Durchführen – Reflektieren*, das an das Selbstregulationsmodell von Schmitz [Sc01] angelehnt ist, wiederholt. Denn durch dieses Schema sollen Kompetenzen des selbstregulierten Lernens, die die Voraussetzung für ein lebenslanges, aktives und eigenverantwortliches Lernen und somit für eine erfolgreichere Bewältigung der Schul- und Arbeitswelt darstellen, gefördert werden [OPS15]. Dieses Bearbeitungsschema findet sich sowohl in dem nach diesen Phasen strukturierten Arbeitsheft, als auch in der ersten Phase der Darstellung der Stundentransparenz wieder. Dieses Raster der Stunde besteht aus Karten für die einzelnen Schritte der Arbeitsphase und enthält zusätzlich Informationen dazu, ob eine Aufgabe von allen SuS (ein Affe) oder nur von fitteren/ schnelleren Kindern (zwei Affen) gelöst werden muss und welche Sozialform in der jeweiligen Phase gefordert wird (angeheftete Klammern). Außerdem werden die SuS auf das bekannte Helfersystem der Klasse hingewiesen, das auf ein gegenseitiges Unterstützen der Kinder und ein gemeinsames Lernen abzielt. Dazu klemmen SuS, die mit der Aufgabe fertig sind, ihren Namen an den Helferbaum. So können sich Kinder, die nicht weiter kommen, einen Helfer aussuchen, dessen Klammer abnehmen und sich bei ihm Hilfe holen. Zum Schluss dieser Phase betont die Lehrkraft, dass es verschiedene Möglichkeiten zur Bearbeitung dieser Aufgabe gibt und dazu wenige oder viele verschiedene Codeblöcke verwendet werden können. So wird sichergestellt, dass die SuS individuelle Lernwege beschreiten und der Austausch in der Präsentationsphase gewinnbringend ist. Mit einem Gong beginnt die Arbeitsphase und es werden 40 Minuten in der Zeitwächteruhr eingestellt.

Anschließend an diese Phase erfolgt die Erarbeitungsphase, die von SuS im Sinne einer sozialen Differenzierung in Einzel- oder Partnerarbeit ausgeführt werden kann. Dies bietet den Vorteil, dass Kinder, die noch kaum eigene Ideen und Vorstellungen zum Programmieren haben, sich mit einem Partner austauschen und SuS, die sehr fit sind, ihr eigenes Tempo bei der Bearbeitung wählen können. Das konzipierte Projekt ist dabei von sich aus ansatzweise natürlich differenzierend, da es verschiedene Lösungen auf unterschiedlichen Niveaus zulässt [Kr20]. So können SuS eine einfache Programmierung wählen, indem sie lediglich Bewegungsblöcke hintereinander reihen oder verschiedene Blöcke aus verschiedenen Codeblockbereichen kombinieren und so eine schwierigere Lösung wählen. Außerdem ist es den SuS möglich, mehrere Möglichkeiten für die Lösung der Problemstellung zu finden. Durch die Knobelaufgabe, die schnelleren SuS angeboten wird, wird ein individuelles Lern- und Arbeitstempo gewährleistet und sichergestellt, dass leistungstärkere SuS nicht nur die einfache, sondern sich auch mit der schwereren Möglichkeit auseinandersetzen. Dieses individuelle Lerntempo wird zusätzlich durch Tippkarten und -seiten unterstützt und sorgt dafür, dass jedes Kind das Lernziel erreichen kann. Zur Ergebniskontrolle überprüfen die SuS ihre Planungen am Computer und werden so durch eine schnelle Rückmeldung motiviert. Zudem ermöglicht diese Form der Ergebniskontrolle der Lehrkraft Zeit, um individuell auf einzelne SuS, denen nicht durch das Helfersystem der Klasse geholfen werden kann, einzugehen und sie in ihrem Arbeitsprozess zu unterstützen. Diese Erar-

beitungsphase dient also im Hinblick auf das übergeordnete Lernziel zunächst vor allem dazu, dass jedes Kind **eine** Möglichkeit zur Programmierung einer bewegenden Figur in Scratch anwenden und erklären kann. Der durch die Zeitwächteruhr ausgelöste Gong beendete auch diese Unterrichtsphase und ist für die SuS ein bekanntes Zeichen dafür, dass jetzt die nächste Phase des Unterrichts beginnt.

In der Auswertungs-/ Präsentationsphase bilden die SuS anhand ihrer Gruppentische Gruppe, da diese Zusammenstellung bereits im Voraus durch die Lehrkraft erfolgt ist. Dabei war entscheidend, dass die SuS zwar eine hohe Leistungsheterogenität aufweisen, die Gruppen aber zueinander ungefähr gleich leistungsstark sind. Diese Leistungsheterogenität ist erwünscht, da vermutlich auch so die Lösungen der SuS sehr heterogen sind. So lernen die SuS im Austausch mit ihrer Gruppe neue Möglichkeiten der Lösung des Problems kennen und werden darin unterstützt, **verschiedene** Möglichkeiten der Programmierung einer sich bewegenden Figur in Scratch anzuwenden und erklären zu können. In dieser Phase wird die Gruppenarbeit als Sozialform gewählt, da sie zum einen den Vorteil bietet, dass alle Kinder ihre Lösung präsentieren können und zum anderen, dass es vielen SuS einfacher fällt die eigene Lösung vor einer kleineren Gruppe zu präsentieren. Als Abschluss der Gruppenarbeit wählen die SuS zwei verschiedene Lösungen aus, die sie in der Sicherungsphase der Klasse präsentieren wollen. Zur Entscheidung ziehen die SuS eigene Kriterien heranziehen, die sie dann in einer etwaigen Präsentation erklären.

Die letzte Phase des Unterrichts stellt die Sicherung dar. In dieser Phase präsentiert eine geloste Gruppe ihre verschiedenen Lösungen der Klasse, erklärt worin genau sich die Lösungen unterscheiden und wird von den anderen Gruppen ergänzt. Ist an dieser Stelle der Zeitpuffer noch vorhanden, kann er dazu verwendet werden eine weitere Gruppe präsentieren zu lassen. Im anschließenden Klassengespräch wird mit den SuS erarbeitet, dass alle verschiedenen Lösungsmöglichkeiten auf zwei verschiedene Varianten zurückzuführen sind. Diese zwei Möglichkeiten werden zur Sicherung an der Tafel und durch die Kinder im Arbeitsheft festgehalten. Außerdem wird sich bei ausreichender Zeit mit der weiterführenden Frage beschäftigt, ob diese Arten der Programmierung von Bewegungen in Scratch immer funktionieren. Dabei können die Kinder zum Beispiel entdecken, dass nur eine Möglichkeit (Programmierung der Pfeiltasten) identisch übernommen werden kann. Diese Phase dient somit sowohl der Kontrolle des Verständnisses aller Kinder, als auch der Sicherung der Erreichung des Stundenziels durch alle SuS. In der Hausaufgabe soll das Erlernte auf ein neues, komplexeres Labyrinth übertragen, gefestigt und so erkannt werden, dass eine Programmierung der Pfeiltaste der weniger aufwendige Weg ist. Auch diese Hausaufgabe ist im Arbeitsheft einzutragen, so dass vor allem schwächere SuS die Möglichkeit haben, zur Lösung der Aufgabe die Aufzeichnungen der Stunde zur Hilfe zu nehmen.

Auffallend ist, dass diese Stunde vor allem einen häufigen Wechsel der Sozialformen aufweist. Diese werden aus verschiedenen Gründen vorgenommen: Zur Unterstützung der Erreichung des Stundeziels, zur Steigerung der Konzentrations- und Aufnahmefähigkeit durch mehr Abwechslung, zur Entwicklung des Sozialverhaltens und zur Aktivierung und Motivation der SuS [Be07].

Zusammenfassend ist zu sagen, dass die Umsetzung des Themas *Wir programmieren die Bewegungen einer Figur durch Scratch!* in Scratch eine geeignete Methode darstellt, die in Kapitel 1 angestrebten Ziele und Kompetenzen zu erreichen. Denn durch

die Bearbeitung des Arbeitsheftes entdecken und wenden die SuS eine Lösungsmöglichkeit der Problemstellung an. Durch die Partizipation an der Gruppenarbeit präsentieren die SuS ihre Ergebnisse und lernen neue Lösungsmöglichkeiten kennen. Anschließend wird durch die Sicherungsphase im Plenum ihr Wissen gefestigt, so dass die SuS am Ende der Doppelstunde verschiedene Möglichkeiten zur Programmierung einer bewegenden Figur in Scratch anwenden und erklären können, wodurch das Stundenziel erreicht wurde.

3. Artikulationsschema⁴

Dauer (Min.)	Unterrichtsphase	Unterrichtsinhalt	Sozial-/ Aktionsform	Materialien/ Medien/Werkzeuge	didaktisch-methodischer Kommentar
5 Minuten	Einstieg	<p>Wiederholung der wichtigsten bisher gewonnenen Kompetenzen aus der Reihe „Wir programmieren mit Scratch!“ anhand von falschen Aussagen</p> <ul style="list-style-type: none"> - Geschichte: LK hat Gespräch zweier Kinder aus der Nachbarsklasse über das Programmieren mit Scratch mitbekommen, dabei ist ihr aufgefallen, dass die Kinder über zwei Aussagen diskutiert haben (Aussagen stehen an der Tafel) - „Eine Figur kann bei Scratch immer nur geradeaus laufen!“ - „Eine Figur kann bei Scratch immer nur kleine Schritte gehen!“ - Lehrkraft bittet die SuS dazu Stellung zu nehmen 	Klassengespräch im Theaterkreis	Tafel, Kreide Alternativ: Smartboard; OHP	<ul style="list-style-type: none"> - aktiviert das vorhandene Vorwissen - regt zum Austausch über die bekannten Funktionen an - festigt die bereits erlernten Funktionen
5 Minuten	Hinführung	<p>Vorstellung des heutigen Projektes mithilfe der Seite 1 des Arbeitsheftes</p> <ul style="list-style-type: none"> - Affe Charly hat seine Bananen im Labyrinth verloren - SuS sollen mit Scratch den Affen so programmieren, dass er seine Bananen wiederfindet <p>Kurze Wiederholung des Bearbeitungsschemas: Planung – Durchführung - Reflexion (bekannt aus dem Sachunterricht und der Arbeit an Projekten)</p>	Klassengespräch im Theaterkreis	Seite 1 des Arbeitsheftes; Bearbeitungsschema an der Wand; Tippkarten; Studententransparenzkarten	<ul style="list-style-type: none"> - Motivation wecken durch Kennenlernen des Projektes - Studententransparenz durch Vorstellung der einzelnen Unterrichtsphasen - Organisation der Arbeitsphase - Anwendung des bekannten Konzepts des selbstregulierten Lernens

⁴ Unter der Artikulation wird im didaktischen Kontext die (zeitliche) Abfolge der Unterrichtsphasen verstanden.

Dauer (Min.)	Unter- richtsphase	Unterrichtsinhalt	Sozial-/ Aktionsform	Materialien/ Medien/Werkzeuge	didaktisch-methodischer Kommentar
		<p>Organisation der Arbeitsphase:</p> <ul style="list-style-type: none"> - Freie Wahl der Sozialform (Einzel- oder Partnerarbeit) - Vorstellung des Pflichtteils: S. 1-4 - Vorstellung der Tippkarten - Erläuterung der Aufgaben für die SuS, die früher fertig sind (Name an den Helferbaum; Knobelaufgabe; S. 5) - Lehrkraft betont, dass mehrere Möglichkeiten für die Lösung der Problemstellung und die Verwendung ganz unterschiedlicher Codeblöcke möglich sind. 			
40 Minuten	Erarbeitung	<p>Bearbeitung des Arbeitsheftes</p> <ul style="list-style-type: none"> - SuS arbeiten an ihrem Arbeitsheft, um das Problem vom Affen Charly zu lösen. - Sie wählen frei, ob sie in Einzel- oder in Partnerarbeit die Aufgaben lösen wollen. - Sie haben die Möglichkeit das Projekt/ die Aufgabe auf verschiedene Weisen zu lösen - Schnelle SuS können als Helfer den anderen SuS helfen oder die Knobelaufgabe lösen 	Einzelarbeit/ Partnerarbeit	Arbeitsheft, mindestens für jedes zweite Kind einen Computer mit Internetzugang oder dem installiertem Programm Scratch; Tippkarten	<p>SuS führen das Projekt anhand der Arbeitsblätter eigenständig durch</p> <ul style="list-style-type: none"> - Anwenden des bereits erlernten Wissens und Ausbau der eigenen Kompetenzen - Verschiedene Lösungen auf unterschiedlichen Niveaus ermöglichen differenziertes Arbeiten - SuS helfen sich durch den Helferbaum gegenseitig → kooperatives Lernen <p>Lehrkraft gib bei Bedarf (falls die SuS sich nicht gegenseitig helfen können) individuelle Hilfestellungen und unterstützt</p>

Dauer (Min.)	Unterrichtsphase	Unterrichtsinhalt	Sozial-/Aktionsform	Materialien/Medien/Werkzeuge	didaktisch-methodischer Kommentar
					sie in ihrem individuellen Lernprozess
20 Minuten	Auswertung/Präsentation	<p>Vorstellung der Arbeitsergebnisse Die SuS bilden Gruppen, in denen sie sich ihre verschiedenen Lösungen mithilfe ihrer Planungen vorstellen (ggf. können sie sich einzelne Projekte auch nochmal am PC vorspielen). SuS wählen anhand eigener Kriterien mindestens zwei verschiedene Lösungen aus, die sie später der Klasse präsentieren wollen.</p>	Gruppenarbeit	Arbeitsheft	Austausch über die Lösungswege für viele SuS in der Gruppe leichter (einfacher frei zu reden) und es kommen mehr Kinder zu Wort
10 Minuten	Sicherung	<p>Vorstellung der Lösungen in der Klasse: Eine Gruppe stellt ihre Lösung am Smartboard oder PC mit Beamer vor (wird gelöst), die anderen Gruppen ergänzen diese oder stellen eine andere Möglichkeit zum Lösen des Projektes vor. Festhalten der zwei Lösungsmöglichkeiten durch Ergänzung der Sätze (im Arbeitsheft): Der Affe Charly kann auf zwei Weisen mit unserer Hilfe an seine Bananen gelangen:</p> <ol style="list-style-type: none"> 1. Durch genaue Programmierung des Weges 2. Durch Programmierung einer Steuerungsfunktion <p>Weiterführende Fragen: Funktionieren diese Möglichkeiten der Program-</p>	Klassengespräch im Plenum	Loskugeln, Arbeitshefte, Smartboard (ggf. PC mit Beamer), Computer	<ul style="list-style-type: none"> - Gemeinsames Zusammenfassen der Ergebnisse dient einer Kontrolle des Verständnisses → sichert das Erreichen des Stundenziels durch alle Kinder - Beseitigung von letzten Fragen - Motivation der SuS für die folgenden Stunden zu diesem Thema sichern

Dauer (Min.)	Unter- richtsphase	Unterrichtsinhalt	Sozial-/ Aktionsform	Materialien/ Medien/Werkzeuge	didaktisch-methodischer Kommentar
		mierung von Bewegungen in Scratch auch bei einem anderen Labyrinth?			
5 Minuten	Hausaufgabe	Anwendung des heutigen Erlernten auf ein anderes Labyrinth - SuS können sich einen der beiden Wege aussuchen	Einzelarbeit	Computer (falls vorhanden, ansonsten Lösung auf dem Blatt) Arbeitsheft S.7	- Festigung des Erlernten - Verständniskontrolle
5 Minuten	Zeitpuffer				Zeitpuffer, falls Probleme oder Unerwartetes auftreten oder um mehrere Gruppen vorstellen zu lassen.

Literaturverzeichnis

- [AZ17] Aachener Zeitung: Das Tablet ersetzt Stift und Heft in der Schule nicht, <http://www.aachener-zeitung.de/lokales/region/das-tablet-ersetzt-stift-und-heft-in-der-schule-nicht-1.1704876>, 01.09.2017
- [Be07] Becker, Georg: Unterricht planen. Handlungsorientierte Didaktik. Teil III. Beltz Verlag, Weinheim und Basel, 2007.
- [En16] Enactus Aachen e. V. (Hrsg.): Informatik und Programmieren für Kinder. Modulhandbuch für die Verwendung in Informatikkursen der Klassen drei bis sechs, <https://www.it-for-kids.org/sites/default/files/Lehrkonzept.pdf>, 1. Vorabversion, Stand: 20.6.2016
- [GI17] Gesellschaft für Informatik (GI) e. V. (Hrsg.): Bildungsstandards Informatik für den Primarbereich – zur Diskussion (Empfehlungen), <https://metager.to/gibspdf>, Version: 689, Stand: 21.08.2017
- [KS10] Krauthausen, Günter; Scherer, Petra: Umgang mit Heterogenität. Natürliche Differenzierung im Mathematikunterricht der Grundschule, https://www.sinus-an-grundschulen.de/fileadmin/uploadsMaterial_aus_SGS/Handreichun_Krauthausen-Scherer.pdf, Stand: März 2013
- [Ka07] Kattmann, U.: Didaktische Rekonstruktion. In: Krüger, D./ Vogt, H. (Hrsg.): Theorien der biologiedidaktischen Forschung. Ein Handbuch für Lehramtsstudenten und Doktoraten. Berlin: Springer, S.93-98, 2007
- [MSW08] Ministerium für Schule und Weiterbildung des Landes NRW (Hrsg.): Richtlinien und Lehrpläne für die Grundschule in NRW, Ritterbach Verlag, Frechen, 2008, http://www.schulentwicklung.nrw.de/lehrplaene/upload/lehrplaene_download/grundschule/grs_faecher.pdf, Stand: 07.09.2017.
- [Mö16] Möller, Kornelia: Bedingungen und Effekte qualitätvollen Unterrichts – ein Beitrag aus fachdidaktischer Perspektive. In: McElvany, N.; Bos, W.; Holtappels, H.; Gebauer, M.; Schwabe, F. (Hrsg.): Bedingungen und Effekte guten Unterrichts, Waxmann, Münster, S.43-64, 2016.
- [MSW15] Ministerium für Schule und Weiterbildung des Landes NRW (Hrsg.): Richtlinien und Lehrpläne für die Grundschule in NRW, Ritterbach Verlag, Frechen, 2015, <https://www.schulentwicklung.nrw.de/lehrplaene/lehrplannavigator-s-i/gesamtschule/wahlpflichtunterricht/informatik/klp/index.html>, Stand: 12.09.2017
- [OPS15] Otto, Babara; Perels, Franziska & Schmitz, Bernhard: Selbstreguliertes Lernen. In Ditton, Hartmut; Gräsel, Cornelia; Gniewosz, Burkhard & Reinders, Heinz (Hrsg.): Empirische Bildungsforschung. VS Verlag für Sozialwissenschaften, Wiesbaden, S. 41–53, 2015.
- [Sc01] Schmitz, Bernhard: Self-Monitoring zur Unterstützung des Transfers einer Schulung in Selbstregulation für Studierende. Eine prozessanalytische Untersuchung. Zeitschrift für Pädagogische Psychologie, 15, 179–195, 2001

[Sw17] Sweigert, A.: Coole Spiele mit Scratch. Lerne programmieren und baue deine eigenen Spiele, dpunkt.verlag, Heidelberg, 2017

Abbildungsverzeichnis

Abbildungen von Scratch: Alle Abbildungen wurden aus dem Programm Scratch genommen:
<https://scratch.mit.edu/>, Stand: 10.09.2017

Helferbaum: <http://insertmedia.office.microsoft.com>

Anhang

I. Tafelanschriften

Einführung:

„Eine Figur kann bei Scratch immer nur geradeaus laufen!“

„Eine Figur kann bei Scratch immer nur kleine Schritte gehen!“

Sicherung (Beispiel):

Der Affe Charly kann auf zwei Weisen mit unserer Hilfe an seine Bananen gelangen:

- 1 Durch die genaue Programmierung des Weges.
- 2 Durch Programmierung einer Steuerungsfunktion.

II. Karten zur Studententransparenz:

Projektarbeit (S. 1-4)



Planung – Durchführung – Reflexion

Knobelaufgabe + Helfer (S.5)



1. Gruppenarbeit
Vergleich der Lösungen



2. Präsentation der Ergebnisse (S. 6)



Klammern für die Sozialformen:

Zu 1.:

Einzelarbeit

Partnerarbeit

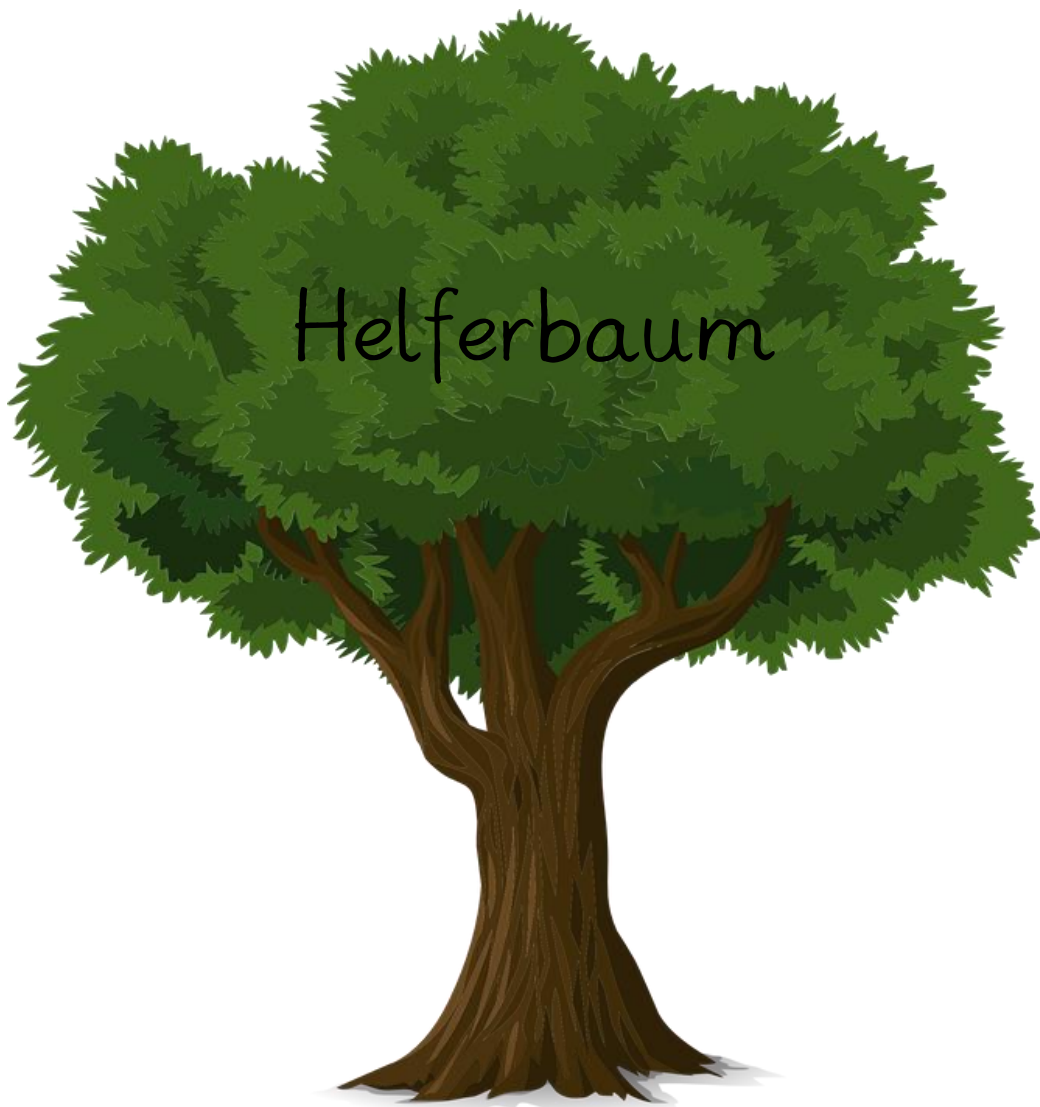
Zu 2.:

Einzelarbeit

Zu 3.:

Gruppenarbeit

III. Helfersystem: Helferbaum: (Namen der SuS werden daran geklemmt)



IV. Arbeitsheft

Mein



SCRATCH

Arbeitsheft

Thema: Bananen im Labyrinth.

Mein Name: _____

Falls du die einzelnen Blöcke nicht mehr kennst, schau dir doch die Tippseiten an!



- Überlege dir nun, wie du den Affen Charly am Computer so programmieren kannst, dass er an sein Ziel kommt.
Welche der bekannten Blöcke benötigst du dazu?
Nutze dazu auch deinen aufgezeichneten Weg aus Aufgabe 1.

So solltest du beginnen:

Hilfe zur Ideenfindung findest du auf Tippkarte 2!



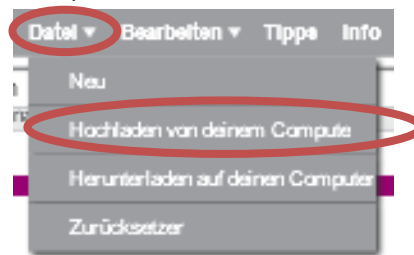
Falls du nicht mehr weißt, wie und wo du Blöcke hinzufügst, schaue dir nochmal den Aufbau von Scratch an (Tippkarte 1).



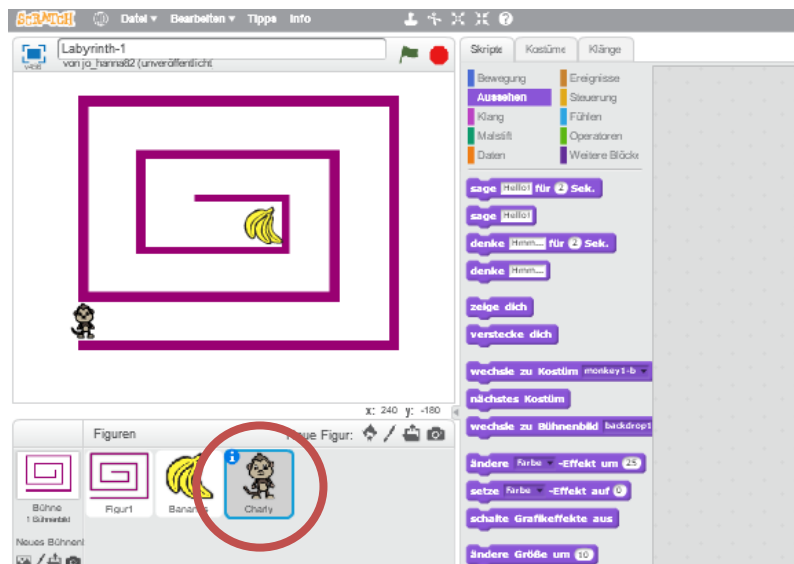
Durchführung:

3. Nun kannst du deinen Plan am Computer umsetzen.

Lade dazu die Datei Labyrinth-1.sb2 in den Scratch-Editor



und bearbeite den Scratch Bereich des Affen Charly :



Tipp:

Funktioniert deine Programmierung nicht wie gedacht, dann schau dir noch einmal deine Planungen genau an. Vielleicht passen deine Blöcke ja nicht alle zusammen.

Wenn du keinen Fehler finden kannst, frage einen deiner Mitschüler, dessen Name schon am Helferbaum klemmt.

4. Wenn du mit deiner Programmierung zufrieden bist, speichere dieses ab. Hefte nun deinen Namen mit einer Klammer an den Helferbaum und bearbeite danach die Reflexionsaufgabe.



Reflexion:

5. Was gefällt dir an deiner Programmierung gut?

6. Was würdest du bei der nächsten Programmierung besser machen?



Knobelaufgabe:

1. Es gibt mehrere Wege diese Aufgabe zu lösen.

Fällt dir noch eine Möglichkeit ein, die sich von deiner unterscheidet?

Beschreibe sie und überlege dir gegebenenfalls schon eine Programmierung mit den bekannten Blöcken.



Klassenergebnis

Der Affe Charly kann auf zwei Weisen mit unserer Hilfe an seine Bananen gelangen:

1. _____

2. _____

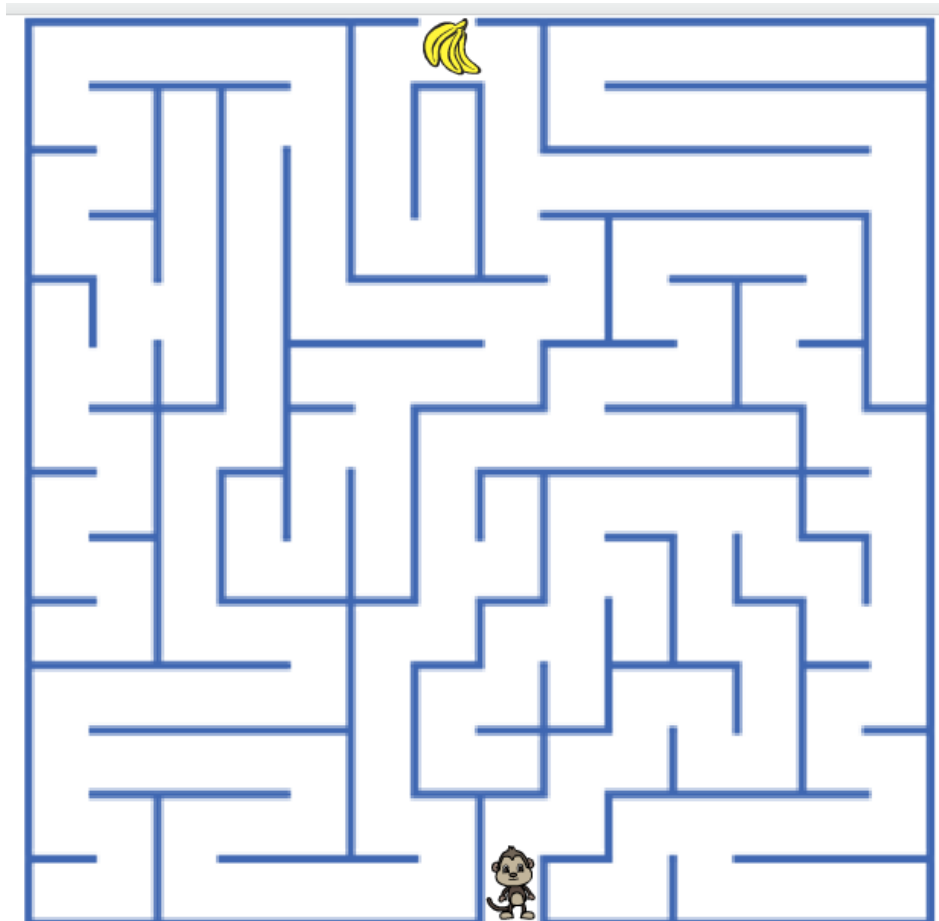
Tipp: Nutze das Arbeitsheft als Hilfe!



Hausaufgabe

Du hast heute Charly schon einmal geholfen an seine Bananen zu kommen. Auf dem Weg mit den Bananen nach Hause sind sie ihm aber leider von älteren Affen geklaut worden. Diese haben die Bananen absichtlich ganz schwer versteckt und Charly kann sie nicht finden. Kannst du ihm beim Suchen helfen, indem du ihn richtig programmierst? (Öffne: Labyrinth-2.sb2).

Denke dabei an die zwei unterschiedlichen Wege, die wir heute kennengelernt haben und überlege dir welcher besser passen könnte.



Platz für Ideen:

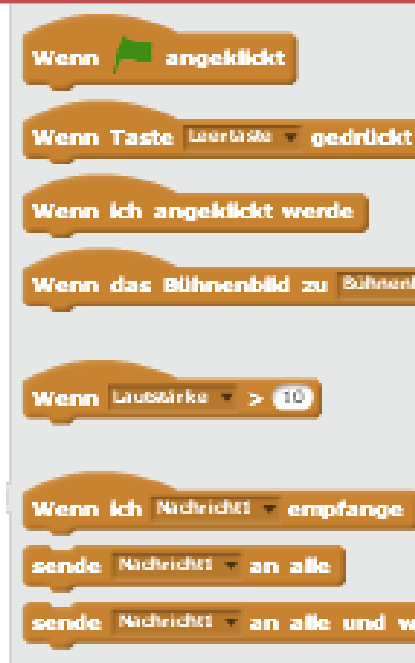
Tippseiten: Blöcke



Bewegung

Kategorie	Blöcke	Erklärung
Bewegung		<ul style="list-style-type: none"> - Figur geht 10 Schritte nach rechts - Figur dreht sich: 90 Grad entsprechen dabei einer Viertel Drehung
		<ul style="list-style-type: none"> - So läuft die Figur in eine andere Richtung - Hier dreht sich die Figur zum Mauszeiger - Hier geht die Figur zu einer bestimmten Position
		<ul style="list-style-type: none"> - Hier verschiebst du die Position horizontal (X) oder vertikal (Y) um eine bestimmte Länge oder auf eine bestimmte Position
		<ul style="list-style-type: none"> - Hier prallt die Figur von der Begrenzung ab

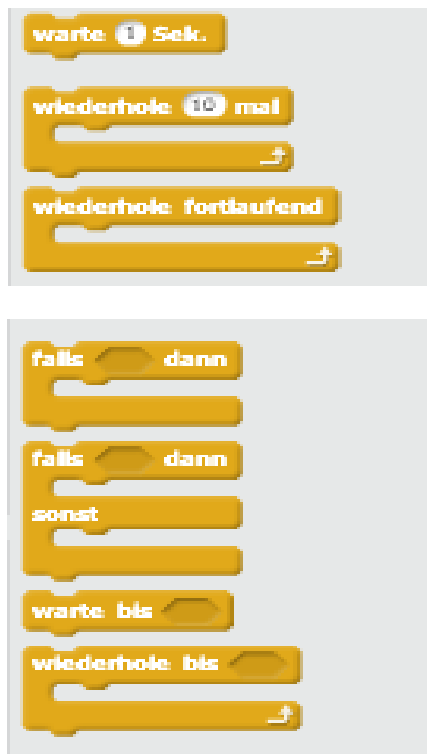
Ereignisse



- Stellt sicher, dass das Programm mit einem Klick auf grüne Fahne startet.

Die übrigen Blöcke sind Wennblöcke, das heißt, alle Blöcke die dahinter gereiht sind, werden ausgeführt, wenn das jeweilige Ereignis (bestimmte Taste, Nachricht etc.) erfüllt wird.

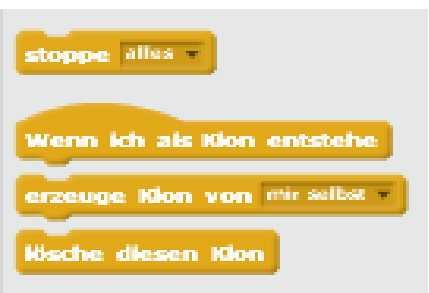

Steuerung



Diese Blöcke werden verwendet, um Skripte zu steuern.

- Hier wartet die Figur eine bestimmte Anzahl von Sekunden
 - Die in den Klammern stattfindenden Blöcke werden mehrmals/ oder fortlaufend (also die ganze Zeit über) wiederholt

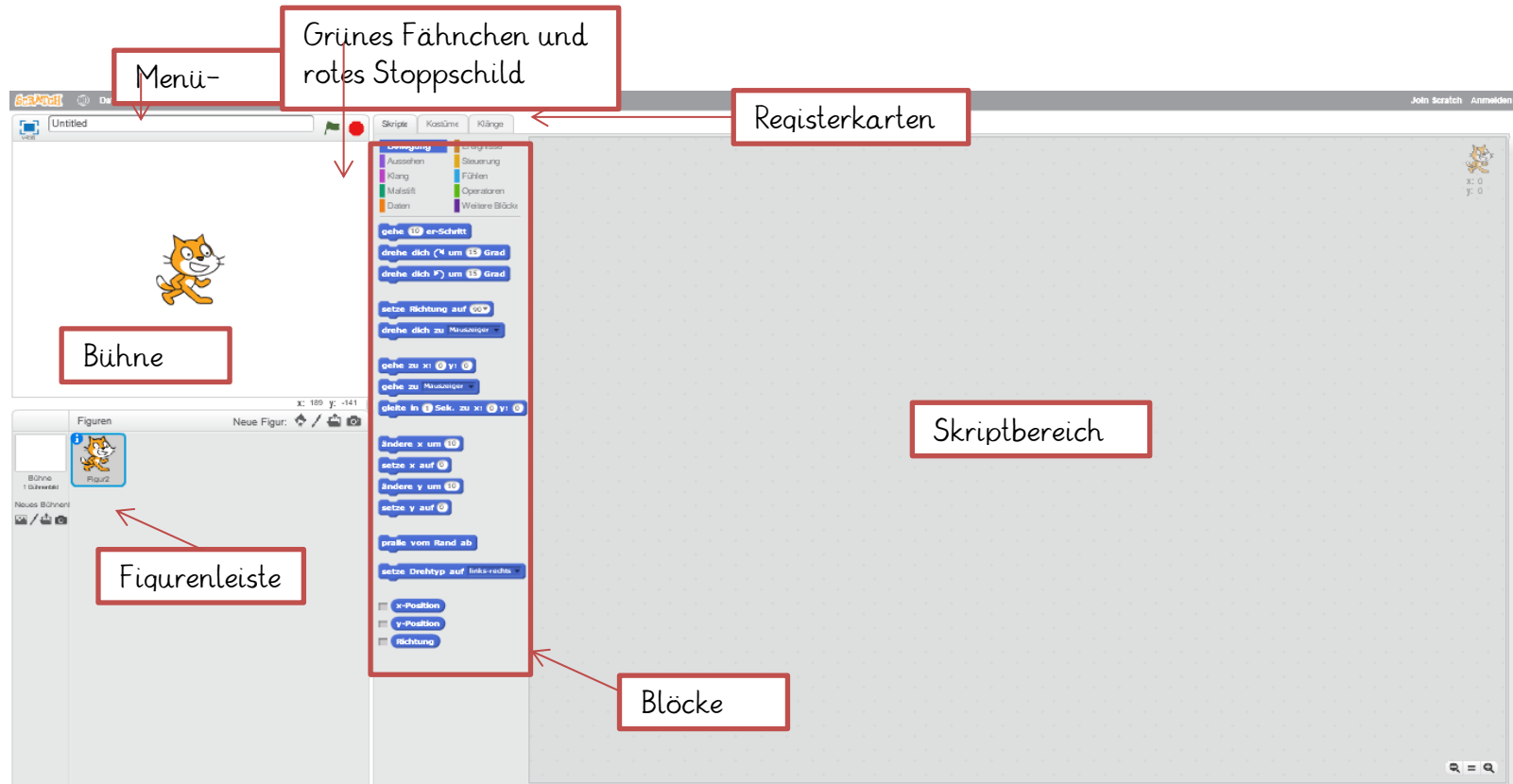
- Auch diese hier sind Klammerblöcke, falls ein bestimmtes Ereignis passiert, soll auch etwas anderes passieren oder falls etwas nicht passiert, soll auch etwas Bestimmtes passieren.

		<ul style="list-style-type: none"> - Endblock. Alles wird gestoppt. - Diese Blöcke betreffen das Klonen.
<p><u>Fühlen</u></p>	<p>Diese Blöcke stellen bestimmte Zustände fest (ob etwas gedrückt ist, oder nicht) oder messen diese.</p> 	<p>Gibt an, ob die Figur ein anderes oder den Mauszeiger berührt.</p> <p>Gibt an, ob die ausgewählte Taste im Augenblick gedrückt wird.</p> <p>Gibt an, ob die Maustaste gedrückt ist.</p>

V Tippkarte

Tippkarte 1: Aufbau des Scratch-Editors

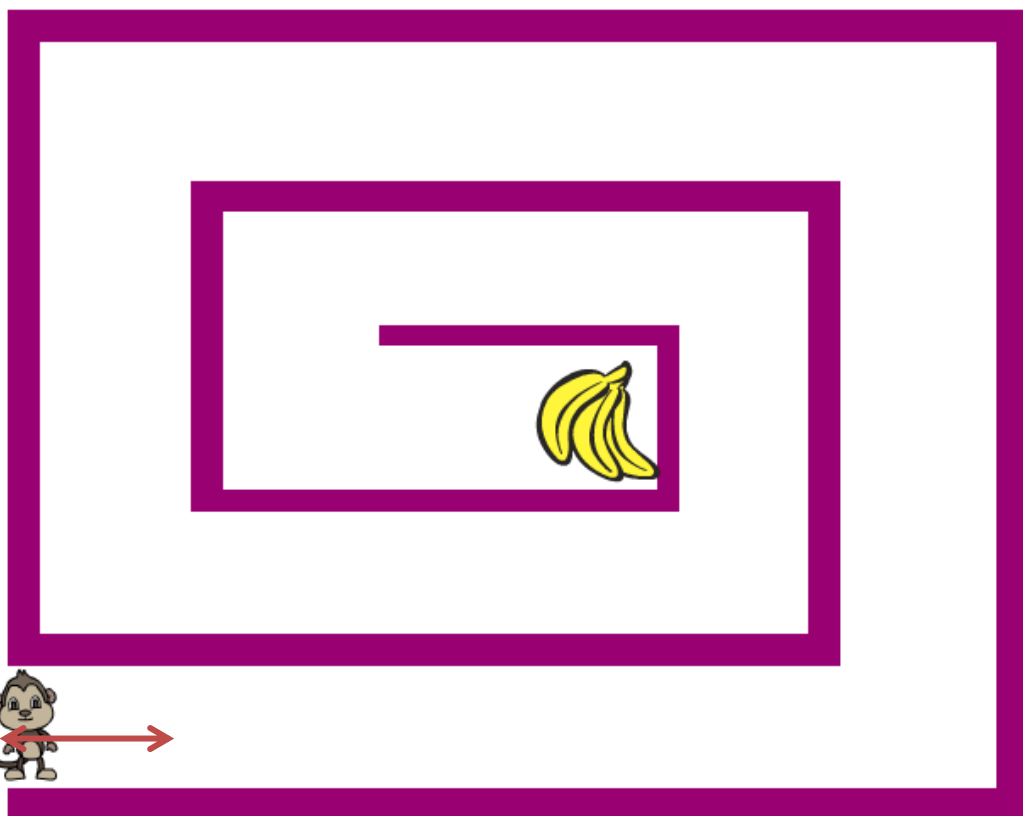
Der Scratch-Editor ist die Software, in der du Codeblöcke zusammenbaust, um deine Spiele, Animationen und Kunstwerke zu gestalten.



Für jede Figur kannst du ein eigenes Skript anlegen. Dazu musst du eine Figur in der Figurenliste auswählen und dann Blöcke aus der Blockpalette in den Skriptbereich ziehen.

Tippkarte 2: Ideenhilfe

Dieser Pfeil entspricht 50 Schritten.



VI: Beispiel: Programmierung der Pfeiltaste



VII: Erwartungshorizont: Arbeit mit dem Arbeitsheft

a) Inhaltliche Leistung

	Die Schülerin/ der Schüler	Max.	Err.
1	plant seine Lösung zur Problemstellung, indem er/ sie <ul style="list-style-type: none"> - einen zielführenden Weg in das Labyrinth zeichnet. - bei seinen Überlegungen zur Programmierung Blöcke, die Einfluss auf die Bewegungen des Affen haben, verwendet. 	4	
2	setzt seine Planungen am Computer um, indem er/sie <ul style="list-style-type: none"> - zunächst sein geplantes Skript verwendet. - ggf. sein Skript überarbeitet und dieses erneut ausprobiert. 	4	
3	reflektiert seinen Lösungsweg, indem er/ sie <ul style="list-style-type: none"> - Aspekte benennt, die an der eigenen Planung gelungen sind. - Aspekte benennt, die er/sie bei der nächsten Planung besser machen möchte. 	4	
4	Bearbeitet die Hausaufgabe, indem er/ sie <ul style="list-style-type: none"> - eine Lösung plant und dazu <ul style="list-style-type: none"> o einen zielführenden Weg durch das Labyrinth zeichnet. o bei seinen Überlegungen zur Programmierung Blöcke, die Einfluss auf die Bewegungen des Affen haben, verwendet. o bei seinen Überlegungen zur Programmierung einen möglichst effektiven Weg wählt. - seine Planungen am Computer umsetzt und dazu <ul style="list-style-type: none"> o zunächst sein geplantes Skript verwendet. o -ggf. sein Skript überarbeitet und dieses erneut ausprobiert. 	6	
	Summe	18	

Die Knobelaufgabe wird nur in die Darstellungsleistung einbezogen, da sie nicht von allen SuS bearbeitet wird.

b) Darstellungsleistung

	Die Schülerin/ der Schüler	Max.	Err.
1	strukturiert seine Antworten schlüssig.	2	
2	schreibt sprachlich richtig	2	
3	gestaltet seine Planungsskizze verständlich, indem er/ sie <ul style="list-style-type: none"> - die Blöcke verständlich beschriftet. - Veränderungen angemessen kennzeichnet. 	2	
4	nutzt fachsprachliche Formulierungen, etwa <ul style="list-style-type: none"> - Blöcke/ Codeblöcke - Skript 	2	
	Summe	8	