Der Hamster hat das Programm erfolgreich beendet – Grundschulkinder lernen Programmieren

Otto Thiele[1], Petra Jückstock[2]

Abstract: Seit 2014 führten die Autoren des Praxisbeitrags jedes Schuljahr eine Arbeitsgemeinschaft Informatik für Kinder der Klassenstufe vier durch. Die Kinder wurden mittels des Hamster-Struktogrammeditors „HaSE“ an das Programmieren herangeführt. Dies erfolgte, indem die Kinder für die Probleme des Hamsters Lösungsalgorithmen entwickelten und als Hamster-Struktogramme implementierten. Da die Struktogramme als Programme interaktiv visuell ablaufen, konnten die Kinder testen, ob der Hamster ausführte, was sie sich vorgestellt und gedacht hatten.
Begleitet wurde das Lernen der Kinder unter Berücksichtigung des lerntheoretischen Ansatzes „Verständnisintensives Lernen“ und des professionstheoretischen Ansatzes „Verstehen zweiter Ordnung“.

Keywords: Arbeitsgemeinschaft Informatik für Viertklässler, Algorithmen entwickeln, Hamster-Struktogramme, Hamster-Strukogrammeditor „HaSE“, Programmieren, „Verständnisintensives Lernen“, „Verstehen zweiter Ordnung“

1 Die Arbeitsgemeinschaft

Anfang des Jahres 2014 wurde das Konzept zur Durchführung einer schulübergreifenden Arbeitsgemeinschaft Informatik für Jenaer Kinder der vierten Klassen erstellt. Danach erfolgte die Ausschreibung.

„Mein Name ist Bommel[3]*. Ich bin ein kleiner Hamster. Kein richtiger Hamster – denn ich lebe nicht in der Natur und auch nicht in einem Käfig mit Laufrad. Am liebsten hamstere ich nach Herzenslust in Fantasiewelten des Computers. Diese Welten kannst Du Dir für mich ausdenken und am Computer gestalten. Mir wird es dort bestimmt gefallen. Wenn Du auch noch meine Sprache lernst, dann werde ich auf Dich hören und mit Freude und Spaß ausführen, was Du mir sagst.“* [wi16]

Für die Eltern wurde in der Ausschreibung ergänzt, dass das Konzept der Arbeitsgemeinschaft darauf ausgerichtet ist, das analytische, logische und algorithmische Denken der Kinder altersgerecht und spielerisch anzuregen und zu fördern. Das heißt, die Kinder sollen befähigt werden, algorithmische Probleme am Computer mit dem an der Universität Oldenburg entwickelten Hamster-Struktogrammeditor[4] „HaSE“ zu lösen [wi16].

[1] Verständnisintensives Lernen e. V., Löbstedter Straße 67, 07749 Jena, otto.thiele@verstehenlernen.de
[2] witelo e. V., Löbstedter Straße 67, 07749 Jena, p.jueckstock@witelo.de
[3] Für die Kinder war es wichtig, dass der Hamster einen Namen hatte.
[4] Der Hamster-Struktogrammeditor „HaSE“ verfügt über eine für die Kinder leicht bedienbare grafische Benutzeroberfläche, die aus der gestaltbaren virtuellen Welt des Hamsters und dem Editor für Hamster-Struktogramme besteht [Bo16].

Die Resonanz auf die Ausschreibung war groß, so dass die Arbeitsgemeinschaft im Schuljahr 2014/15 erstmals durchgeführt werden konnte. Wie die Kinder seitdem lernten und beim Lernen begleitet wurden, wird nachfolgend beschrieben.

2 Mit dem Hamster lernen

In der Arbeitsgemeinschaft waren die Aufgaben in Geschichten verpackt, die die Fantasie der Kinder anregten. Eine solche Aufgabe war beispielsweise die vom „Hamster Bommel als Nachtwächter“.

Jena war im Mittelalter eine kleine Stadt (Abb. 1) mit einer Stadtmauer und vier Stadttoren. Die Stadtmauer war der Schutz vor räuberischen Angriffen. Das Johannistor, der Pulverturm und Teile des Anatomieturms sowie der Stadtmauer sind bis heute erhalten geblieben.

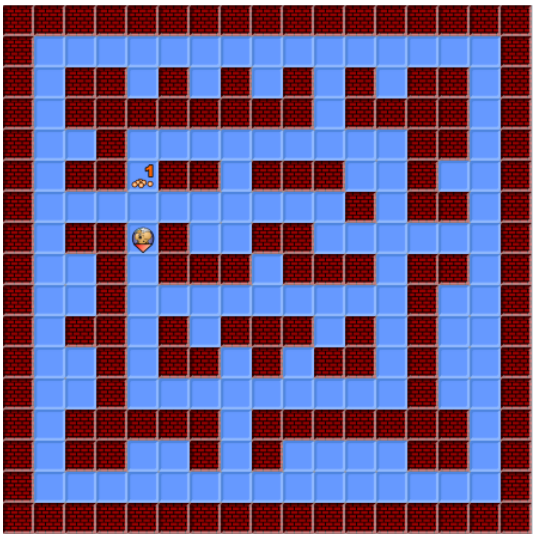

Abb. 1: *Das mittelalterliche Jena mit Hamster Bommel*

Jena hatte damals immer einen Nachtwächter. Der Wächter lief nachts innerhalb von Jena die Stadtmauer ab. Ein Zauberer hat Bommel in das mittelalterliche Jena gebeamt. Dort muss Bommel die Aufgabe des Nachtwächters übernehmen und die Stadtmauer ablaufen. Als Belohnung darf er nach dem Rundgang das leckere Korn verspeisen.

Um diese und andere Aufgaben lösen zu können, lernten die Kinder Algorithmen zu entwickeln. Dazu eigneten sie sich die Hamster-Befehle, Testbefehle, zusammengesetzten Bedingungen, Grundstrukturen (Sequenz, Selektion, Zyklus) und parameterlose Prozeduren an [Bo16].

2.1 Lösen der Einstiegsaufgabe „Hamster Bommel und das leckere Korn“

Bommel schaut aus seinem Bau aufs Feld (Abb. 2). Plötzlich erblickt er am Feldrand ein leckeres Korn. Bommel wäre kein Hamster, würde er das Korn nicht hamstern wollen.

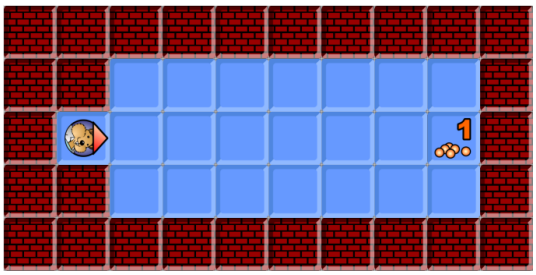

Abb. 2: *Die Welt des Hamsters Bommel*

- *Baue Bommels Welt am Computer.*
- *Hilf Bommel, das Korn zu hamstern.*
- *Prüfe, ob Bommel die Aufgabe löst.*

Als hilfreich erwies sich, dass die Kinder die Aufgabe zunächst in einem Rollenspiel auf den Fliesen zwischen den Wänden des Schulflurs lösten (Abb.3).

Abb. 3: Erfahrungen gehören zum Lernen (Foto: Thomas Corbus)

Im Spiel übernahm jedes Kind die Rolle des Hamsters. Dabei galt es zu beachten, dass der Hamster beim Laufen jeden Schritt nur von einer zur nächsten Fliese ausführen kann. Obwohl alle Kinder den kürzesten Weg zur gegenüberliegenden Wand beschritten, benötigten sie unterschiedlich viele Schritte. Über die Ursachen nachdenkend fanden sie heraus, dass manche von ihnen die Schritte nicht immer „hamsterkonform“ ausführten. Das lag zum einen daran, dass es schwierig war, Fliese für Fliese in kleinen Schritten zu durchlaufen und zum anderen an mangelnder Konzentration. Beim Wiederholen achteten die Kinder untereinander darauf, dass alle mit gleich vielen Schritten zur Wand gelangten. An der Wand angekommen, nahm jedes Kind das Korn in Form eines Papierschnipsels auf. Mit dem korrekten Ausführen der Handlungsfolge, Schritt für Schritt bis zur Wand zu laufen

und das Korn aufzunehmen, hatten die Kinder die Aufgabe im Rollenspiel gelöst.
Zum Rollenspiel sei bemerkt, dass die Kinder insbesondere durch das eigene Handeln und Erleben lernten. Das Rollenspiel hat sich zudem als geeignete Methode erwiesen, die Kinder an die Hamster-Befehle und Grundstrukturen der Algorithmen heranzuführen.

Nach einer kurzen Einführung war es den Kindern problemlos möglich, die virtuelle Welt des Hamsters am Computer zu bauen. Danach ging es damit weiter, einen Algorithmus zu entwickeln. Dazu überlegten sich die Kinder, dass der Hamster in der virtuellen Welt, analog zum Rollenspiel, eine Folge von Schritten ausführen muss, um von seinem Bau zur gegenüberliegenden Wand zu gelangen. Es ergab sich die Frage, wie der Hamster am Computer zum Laufen gebracht werden kann. Als Antwort auf die Frage lernten die Kinder die Notation `vor()` des Hamster-Befehls, sich einen Schritt vor zu bewegen, und die Darstellung im Hamster-Struktogramm kennen. Im Rollenspiel entsprach `vor()`, genau einen Schritt vor zur nächsten Fliese zu laufen. Letztlich mussten die Kinder noch wissen, wie der Hamster am Computer ein Korn „hamstern“ kann. Dazu lernten sie die Notation `nimm()` des Hamster-Befehls, ein Korn aufzunehmen, und die Darstellung im Hamster-Struktogramm kennen. Mit den Erfahrungen aus dem Rollenspiel, der Kenntnis der Hamster-Befehle und deren Notationen entwickelten die Kinder Handlungsfolgen zum Lösen der Aufgabe am Computer (Abb. 4).

Abb. 4: Ein Hamster-Struktogramm erstellen (Foto: Thomas Corbus)

Die Handlungsfolgen stellten sie als Struktogramme (Abb. 5) mithilfe des Hamster-Struktogrammmeditors dar. Da Hamster-Struktogramme auf grafische Weise Hamster-Programme repräsentieren, die interaktiv ausführbar sind, konnten die Kinder überprüfen, ob der Hamster tatsächlich von seinem Bau zur gegenüberliegenden Wand läuft und das Korn „hamstert“.
Es kam vor, dass Hamster-Struktogramme mancher Kinder nicht zu den von ihnen erwarteten Programmabläufen führten. Beispielsweise, wenn der Hamster die Wand nicht erreichte, weil er zu wenige Schritte ausführte und die Fehlermeldung „Hamster versucht ein Korn aufzunehmen, das nicht da ist. :-(“ ausgegeben wurde oder er zu viele Schritte ausführen sollte und die Fehlermeldung „Hamster läuft gegen eine Mauer! :-(“ ausgegeben wurde. Solche Erfahrungen stimmten die Kinder nachdenklich. Sie suchten die Ursachen, weshalb der Hamster nicht tat, was sie sich vorgestellt hatten.

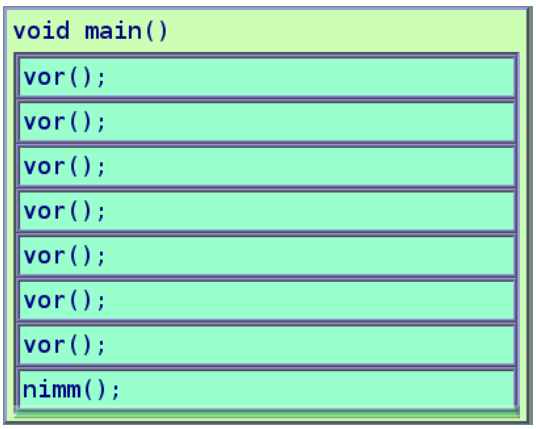

Abb. 5: Algorithmus aus einer Sequenz von Hamster-Befehlen

Beginnend mit dem Rollenspiel, dem Bauen der Hamster-Welt, dem Erstellen der Hamster-Struktogramme am Computer sowie dem Testen des Programmablaufs und den sich eventuell daraus ergebenden Nachbesserungen gelang es allen Kindern, die Aufgabe zu lösen. Das Lösen belohnte das Programm mit der Ausgabe „Hamsterprogramm erfolgreich beendet! Herzlichen Glückwunsch, der Hamster hat das Hamsterprogramm erfolgreich beendet. :-)“.
Das Hamster-Struktogramm ließ sich durch das Einführen der while-Anweisung und des Hamster-Testbefehls `vornFrei()` modifizieren (Abb. 6).

Abb. 6: Modifiziertes Hamster-Struktogramm

Bevor die Kinder allerdings das Struktogramm mit dem Hamster-Struktogrammeditor modifizierten, führten sie auf den Fliesen des Schulflurs im Rollenspiel die Handlungsfolge „solange vorn frei, laufe Schritt für Schritt vor und hamstere danach das Korn“ aus.
Die Aufgabenstellung wurde darüber hinaus noch dadurch erweitert, dass der Hamster mit dem Korn in seinen Bau zurückkehren, das Korn dort ablegen und wieder aus dem Bau schauen sollte. Das Lösen dieser und weiterer Aufgaben sowie das damit verbundene Lernen führte die Kinder zu einem „intuitiven Algorithmusbegriff“, der sich auf Handlungsabläufe bezieht, die als Programme auf dem Computer ausführbar sind.

Nach und nach zeigte sich, dass die Kinder beim Entwickeln von Hamster-Struktogrammen nicht mehr auf Rollenspiele angewiesen waren, weil sie sich die vom Hamster auszuführenden Handlungen immer besser vorstellen konnten.

2.2 Lösen fortgeschrittener Aufgaben

Die Aufgabe „Hamster Bommel als Nachtwächter“ (Abb. 1 in Abschn. 2) lösten die Kinder aufgrund der Erfahrungen, die sie bereits in den Rollenspielen und beim Entwickeln von Algorithmen gemacht hatten. Sie überlegten sich, dass der Hamster entlang der Stadtmauer zum Korn gelangt, indem im Hamster-Struktogramm viermal nacheinander die while-Anweisung und der Befehl `linksUm()` (Abb. 7) angeordnet und abschließend noch die Befehle `vor()` und `nimm()` anfügt werden. Am Computer verfolgten die Kinder mit, wie das Programm ablief und der Hamster entlang der Stadtmauer zum Korn gelangte und es aufnahm. Damit war die Aufgabe gelöst.

Abb. 7: Schleife gefolgt von einem Hamster-Befehl

Leistungsstarke Kinder wurden angeregt, sich weiter mit der Aufgabe zu beschäftigen. Mit Hilfestellungen fanden sie heraus, dass das Wiederholen der while-Anweisung und des Befehls `linksUm()` im Hamster-Struktogramm durch eine äußere Schleife realisiert werden kann. Allerdings ergab sich dabei beispielsweise das Problem, dass der Hamster Runde für Runde lief, nicht zum Stehen kam und dadurch auch nicht seine Belohnung bekam. Jetzt wurde es schwierig, denn die Frage ergab sich, wie der Hamster zum Stehenbleiben gebracht werden kann. In diesem Zusammenhang lernten die Kinder die zusammengesetzte Bedingung `!(kornDa())` kennen (Abb. 8).

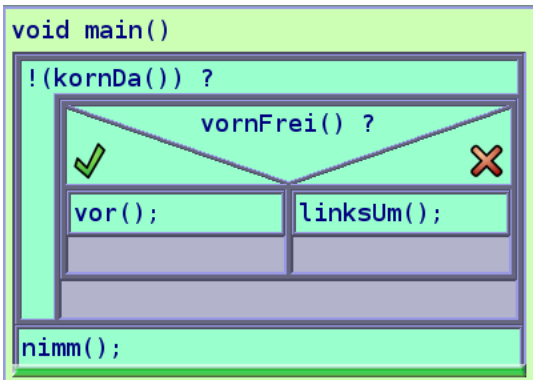

Abb. 8: Lösungsalgorithmus mit hohem Anspruch

Im Laufe der Zeit lernten die Kinder immer besser, analytisch und algorithmisch zu denken. Sie waren in der Lage, in den umrahmenden Geschichten der Aufgaben die Probleme zu erfassen und Lösungsalgorithmen zu entwickeln. Ebenso waren einige Kinder in der Lage, gegebene Aufgaben zu erweitern oder zu modifizieren.

Spaß und Freude bereitete den Kindern, sich zu gegebenen Hamster-Welten (Abb. 9) Geschichten und Aufgabenstellungen auszudenken, wie z. B. die folgende Geschichte.

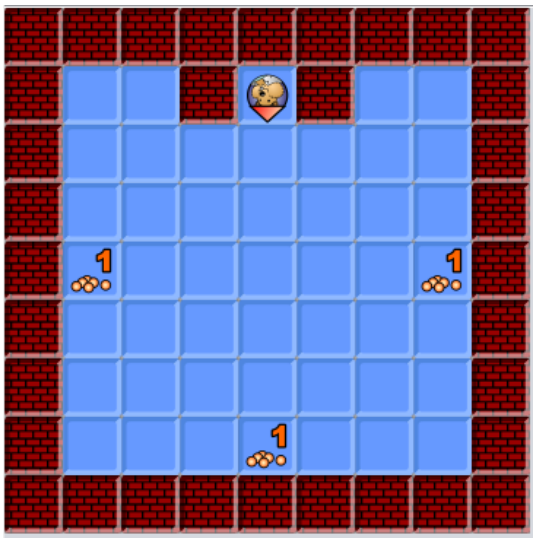

Abb. 9: Vorgegebene Hamster-Welt

„Heute hat der kleine Hamster Bommel Lust auf einen Spaziergang. Er steckt seinen Kopf aus dem Bau. Da erblickt er die drei Körner, die verteilt auf dem Feld rumliegen. Hamster Bommel folgt den Körnern. Nehmen tut er sie aber nicht. Bommel lässt sie lieber für die Vögel übrig. Am Ende sitzt der kleine Hamster Bommel wieder in seinem Bau, als ob nichts gewesen wäre. :-)“

Die Kinder dachten sich auch eigene fantasievolle Hamster-Welten und dazugehörige Aufgaben aus, die sie in Geschichten einbanden.

Außerdem lernten die Kinder parameterlose Prozeduren kennen und wendeten sie an. Beispielsweise erwies es sich als zweckdienlich, mithilfe einer Prozedur aus drei Hamster-Befehlen `linksUm()` den Befehl `rechtsUm()` zu erstellen.

```
void rechtsUm()
    linksUm();
    linksUm();
    linksUm();
```

Abb. 10: Eine parameterlose Prozedur

Nach einem halben Jahr konnten die Kinder bereits anspruchsvolle Aufgaben mit dem Hamster-Struktogrammeditor lösen. Beispielsweise entwickelten sie Algorithmen, mit denen der Hamster aus gegebenen Bildnegativen (Abb. 11) die Bildpositive machte. Natürlich bot es sich an, dass die Kinder ihre Vornamen in Bildnegativen versteckten. Dann sollte der Hamster die Vornamen schreiben.

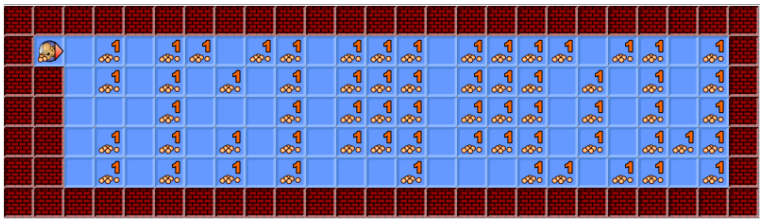

Abb. 11: Aus dem Bildnegativ das -positiv erstellen

3 Das Lernen begleiten

3.1 Verständnisintensives Lernen

Bei der pädagogischen Arbeit mit den Kindern in der Arbeitsgemeinschaft wurde der lerntheoretische Ansatz „Verständnisintensives Lernen“ [Fa03, Fa08] berücksichtigt.

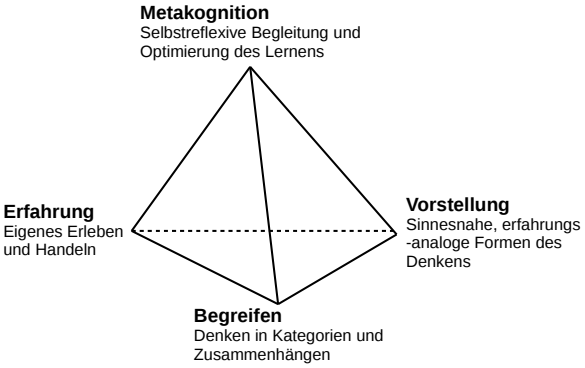

Abb. 12: Tetraedermodell des „Verständnisintensiven Lernens“ [Vi17]

Der Ansatz geht davon aus, dass die Qualität des Lernens zum einen durch das Verstehen bestimmt wird. Verstehen ist grundlegend für das Lernen und lässt sich als kognitiver Modellierungsprozess beschreiben, in dem Erfahrung, Vorstellung, Begreifen[5] und Metakognition dynamisch zusammenwirken. In der Abb. 12 sind die vier Dimensionen des Verstehens grafisch veranschaulicht[6]. Zum anderen bestimmen die Autonomie, soziale Eingebundenheit und Kompetenzen der Lernenden die Lernqualität [Fa08, DR93].

Der lerntheoretische Ansatz erwies sich als hilfreich, die Verstehensprozesse der Kinder mitverfolgen, die individuellen Unterschiede beim Lernen genauer erfassen und besser berücksichtigen sowie die Ursachen für Verstehensprobleme anlysieren zu können. Dazu einige erläuternde Beispiele.

[5] Begreifen wird im lerntheoretischen Ansatz als eine Dimension des Verstehens aufgefasst.
[6] Erfahrung, Vorstellung und Begreifen sind dem Denken und Metakognition ist dem Nachdenken zuordbar.

Wie wichtig eigenes Erleben und Handeln beim Lernen waren, zeigte sich u. a. in den Rollenspielen, die mit den Kindern im Rahmen des Lösens von Aufgaben durchgeführt wurden (Abschn. 2.1).

Interessant war zu beobachten, wie die Kinder erlebten, was der Hamster bei den Programmabläufen ausführte oder auch nicht (Abschn. 2.1). Führte der Hamster nicht das Erwartete aus, wurden die Kinder angeregt, über ihre Algorithmen nachzudenken, nach den Ursachen, weshalb die Algorithmen nicht das Erwartete leisteten, zu suchen und die Algorithmen nachzubessern.

Verstehensprobleme ergaben sich beispielsweise, wenn in Wortlauten von Aufgaben Begriffe verwendet wurden (wie etwa Feldrain), deren Bedeutung die Kinder nicht kannten oder mit denen sie andere Vorstellungen verbanden. Deshalb mussten Formulierungen von Aufgaben geändert werden.

Das logische Denken der Kinder war u. a. bei den zusammengesetzten Bedingungen gefordert. Formale Notationen, wie z. B. `!vornFrei()` oder `vornFrei() && !kornDa()`, bereiteten Probleme und mussten insbesondere geübt werden.

Die Kinder lernten selbstbstimmt („aus sich selbst heraus“) aus Interesse an den Geschichten und den vom Hamster zu bewältigenden Problemen. Wenn Kinder einmal nicht lernten, am Computer in der Welt des Hamsters versanken und keine Probleme lösen wollten, lag es oft am anstrengenden Schultag, den sie bereits hinter sich hatten. Dann ließ man diese Kinder auch einmal in der Welt des Hamsters spielen.

Die Kinder fühlten sich stets in die Arbeitsgemeinschaft eingebunden. Zu keiner Zeit fehlte es einem Kind an Akzeptanz bzw. Anerkennung innerhalb der Lerngemeinschaft.

Das auf Verstehen ausgerichtete Lernen trug mit dazu bei, dass die Kinder folgende fachliche Kompetenzen erwarben:

- Analysieren von Aufgaben, um zu lösende Probleme zu erfassen
- Entwickeln von Lösungsalgorithmen unterschiedlicher Schwierigkeitsgrade
- Darstellen der Lösungsalgorithmen in Hamster-Struktogrammen
- Suchen von Fehlern und Korrigieren fehlerbehafteter Algorithmen
- Lesen und Verstehen gegebener Hamster-Struktogramme
- Erweitern und Modifizieren gegebener Aufgaben
- Erstellen und Verwenden von parameterlosen Prozeduren
- Konstruieren von Aufgaben, Schreiben umrahmender Geschichten für diese Aufgaben und Erstellen von Musterlösungen

3.2 Verstehen zweiter Ordnung

Neben dem lerntheoretischen Ansatz fand der professionstheoretischen Ansatz „Verstehen zweiter Ordnung“ [Fa08] Berücksichtigung. Das heißt, es wurde ein professionell kontrollierter Perspektivwechsel zum Lernen der Kinder vollzogen, der ein ko-konstruierendes

Begleiten des Lernens ermöglichte. Durch Ko-Konstruktion fand ein Lernen statt, dass durch Interaktion und Zusammenarbeit mit den Kindern bestimmt war. Beispielsweise wurden beim Ko-Konstruieren immer wieder Differenzerfahrungen bzgl. des eigenen Verstehens und dem der Kinder gemacht, wenn Kinder Aufgaben und Probleme auf unerwartete und teils bemerkenswerte Art und Weise lösten. Diese Erfahrungen waren ausschlaggebend dafür, den Kindern die Denkwege offen zu halten. Führten die Gedankengänge bei Kindern zu inkorrekten oder keinen Lösungen, erwies sich Ko-Konstruktion ebenso als helfend. Dabei wurde in der Interaktion mit diesen Kindern anerkannt, dass sie sich bei ihren Lösungsversuchen etwas dachten. Was sie sich gedacht hatten, ließ man sich von ihnen erklären. Keinesfalls wurde gegenüber den Kindern geäußert, dass das, was sie gemacht hatten, falsch sei. Stattdessen wurden die Kinder angeregt, über ihre Gedankengänge nachzudenken und Korrekturen vorzunehmen oder andere Wege hin zu den Lösungen einzuschlagen.

4 Fazit

Ein Journalist der Ostthüringer Zeitung, der über die Arbeitsgemeinschaft den Artikel „Informatik kommt bei Viertklässlern gut an“ verfasste, stellte die Frage, ob Grundschulkinder überhaupt die Voraussetzungen mitbringen, sich mit informatischen Problemen zu befassen [Zi15]. Die Frage kann aufgrund der guten Erfahrungen aus der Arbeitsgemeinschaft, der erzielten Lernergebnisse und -qualität sowie dem Spaß und der Freude der Kinder beim Lernen bejaht werden.

Literaturverzeichnis

[Bo16] Boles, D.: HaSE-Hamster-Struktogrammeditor. `http://www.boles.de/hamster/hase.html`, 2016. Stand: 09.03.2017.

[DR93] Deci, E.; Ryan, R.: Die Selbstbestimmungstheorie der Motivation und ihre Bedeutung für die Pädagogik. Zeitschrift für Pädagogik 39, S. 223–237, 1993.

[Fa03] Fauser, P.: Lernen als innere Wirklichkeit. Über Imagination, Lernen und Verstehen. In (Rentsch, I.; Madelung, E.; Fauser, P., Hrsg.): Bilder im Kopf. Texte zum Imaginativen Lernen, S. 201–213. Kallmeyer, Seelze-Velber, 2003.

[Fa08] Fauser, P.: Theorie des Verständnisintensiven Lernens. In (Fauser, P.; Prenzel, M.; Schratz, M., Hrsg.): Was für Schulen! Profile, Konzepte und Dynamik guter Schulen in Deutschland, S. 5–26. Kallmeyer, Seelze-Velber, 2008.

[Vi17] ViL e.V.: Das Tetraeder im ViL. `http://www.verstehenlernen.de/?page_id=2788`, 2017. Stand: 09.03.2017.

[wi16] witelo e.V.: Hamster Bommel und der Computer. `http://witelo.de/lernorte/projekt/hamster-bommel-und-der-computer`, 2016. Stand: 10.03.2017.

[Zi15] Zippel, T.: Informatik kommt bei Viertklässlern gut an. Ostthüringer Zeitung, 14. Februar 2015.