

Rekursion und Iteration: Voruntersuchung zu einem Test

Michael Fothe

Casio-Stiftungsprofessur
Fakultät für Mathematik und Informatik
Friedrich-Schiller-Universität Jena
Ernst-Abbe-Platz 2
D-07743 Jena
fothe@minet.uni-jena.de

Abstract: Informatiklehrerinnen und -lehrer unterrichten einen Themenbereich und möchten anschließend wissen, ob ihr Unterricht die notwendige inhaltliche Bandbreite hatte und ob er den Prüfungsanforderungen entsprach. Auch möchten sie nähere Informationen zum Lernstand und -verhalten ihrer Schülerinnen und Schüler erhalten. Sie laden dazu Testaufgaben aus dem Internet herunter, lassen ihre Schülerinnen und Schüler den Test bearbeiten, korrigieren deren Antworten und ziehen Schlussfolgerungen für ihren Unterricht und dessen Fortentwicklung. Möglicherweise ist dies ein Szenarium, das die Arbeit von Informatiklehrerinnen und -lehrern in der gymnasialen Oberstufe künftig sinnvoll unterstützen kann. Ein erster Schritt zur Umsetzung dieser Idee wird jedenfalls in diesem Beitrag beschrieben. In dem Test geht es um Rekursion und Iteration.

1 Einführung

Das Friedrichgymnasium Altenburg, das Ernst-Abbe-Gymnasium Eisenach, das Albert-Schweitzer-Gymnasium Erfurt, das Perthes-Gymnasium Friedrichroda und das Staatliche Gymnasium „Karl Theodor Liebe“ Gera arbeiten mit dem Autor in einem Projekt zur Schulentwicklung zusammen. Ziel der Zusammenarbeit ist die Erhöhung der Professionalität des Lehrerhandelns im Informatikunterricht. Dieses Anliegen ist ein relevantes Thema in der fachdidaktischen Forschung, was z. B. der Weltkongress ICME-10 für den Mathematikunterricht deutlich machte (vgl. [Ad04]). In diesem Beitrag wird über das Erarbeiten, Durchführen und Auswerten eines Tests berichtet, den insgesamt sieben Grundkurse (einer davon in Spezialeklassen) und zwei Leistungskurse im September 2004 bearbeiteten und in dem es um das Thema „Rekursion und Iteration“ ging. Rekursion und Iteration besitzen Relevanz in der Abiturprüfung. In den EPA Informatik heißt es auf Seite 7 in der Beschreibung der Kompetenzbereiche: „Die Prüflinge ... können verschiedene Problemlösungsstrategien und Techniken wie Iteration, Rekursion und Klassenbildung einsetzen.“ (vgl. [EPA04]). Die Lehrkräfte korrigierten die Antworten ihrer Schülerinnen und Schüler und zogen Schlussfolgerungen. Dies stellt eine Form von Selbstevaluation dar (vgl. [K103]).

Die eigentlichen Testergebnisse blieben in der Schule. Sie wurden von der Universität Jena nicht erfasst. Die beteiligten Lehrerinnen und Lehrer wurden jedoch zu ihren Erfah-

rungen mit dem Szenarium befragt. Nachfolgend werden Teilkompetenzen, Testaufgaben und Musterlösungen angegeben und kommentiert, die Interviews mit den Lehrerinnen und Lehrern zusammengefasst und die Fortführung der Untersuchung beschrieben. Auf eine vorausgegangene Arbeit soll verwiesen werden. In ihr wurde dargestellt, dass Rekursion in Theorie und Praxis der Informatik eine wichtige Rolle spielt, und es wurden Zugänge zur Rekursion ohne Computereinsatz und Beispiele für rekursive Problemlösungen mit dem Computer angegeben und aus didaktisch-methodischer Sicht kommentiert (vgl. [Fo05]).

2 Kompetenzbeschreibung

Als Grundlage für das Erstellen der Testaufgaben wurde die folgende Zusammenstellung von angestrebten Teilkompetenzen, über die die Schülerinnen und Schüler zum Thema „Rekursion und Iteration“ am Ende ihrer Schulzeit am Gymnasium verfügen sollen, erarbeitet:

- (a) Erläutern der Grundlagen von Rekursion und Iteration¹,
- (b) Definieren von informatischen Begriffen auf rekursive Art,
- (c) sachgemäßes Verwenden der Syntaxdefinition einer Programmiersprache,
- (d) exemplarisches Analysieren und Erläutern von Computerprogrammen, denen rekursive oder iterative Algorithmen zugrunde liegen und
- (e) exemplarisches Entwerfen und Implementieren von solchen Computerprogrammen.

Nachfolgend ist das Ergebnis einer Analyse des Thüringer Informatiklehrplans (Grund- und Leistungsfach) hinsichtlich dieser Teilkompetenzen zusammengestellt (vgl. [TKM99]). Das zu erwerbende Wissen und Können ist in dem Lehrplan sehr konkret beschrieben. Im Thüringer Zentralabitur werden sowohl im Grund- als auch im Leistungsfach Informatik seit 1994 Aufgaben mit Bezügen zu Rekursion und Iteration gestellt.

Zu (a): Grundlagen von Rekursion und Iteration sind Gegenstand des Grund- und Leistungsfaches Informatik.

Zu (b): Das Definieren der Begriffe „Liste“ und „binärer Baum“ ist im Lehrplan für das Grund- und Leistungsfach Informatik explizit aufgeführt. Im Leistungsfach sind zusätzlich die Begriffe „Suchbaum“, „Inorder“, „Preorder“ und „Postorder“ zu definieren.

Zu (c): Die Schülerinnen und Schüler nutzen im Grundfach Syntaxdiagramme oder die erweiterte Backus-Naur-Form (EBNF), im Leistungsfach beide Möglichkeiten zur Syntaxdefinition von Sprachen. Der Sprachbericht einer Programmiersprache wird von den Schülerinnen und Schülern regelmäßig bei der Arbeit verwendet.

¹ Vergleichen von Rekursion und Iteration, Äquivalenz von Rekursion und Iteration sowie Prinzip der Abarbeitung eines rekursiven Algorithmus auf einem iterativ arbeitenden Computer

Zu (d) und (e):

Die Schülerinnen und Schüler lernen im Informatikunterricht mehrere iterativ bzw. rekursiv arbeitende Algorithmen kennen. Ein Teil von ihnen wird auch auf dem Computer umgesetzt. Die nachfolgende Tabelle bietet eine Übersicht.

	Grundfach Informatik	Leistungsfach Informatik
Analysieren und Erläutern	<ul style="list-style-type: none"> • Programme, in denen Wiederholungsanweisungen vorkommen • binäres Suchen • Suchen in Texten • Quicksort • Türme von Hanoi • Suchen in einem Labyrinth 	<ul style="list-style-type: none"> • Programme, in denen Wiederholungsanweisungen vorkommen • binäres Suchen • Suchen in Texten • Sortieren durch direktes Mischen • Heapsort • Zeichnen einer rekursiv definierten Figur • Problem des Handlungsreisenden (Näherungsverfahren) • Suchen in einem Labyrinth • Baum minimaler Höhe • Tiefen- und Breitensuche in Graphen
Entwerfen und Implementieren	<ul style="list-style-type: none"> • Programme, in denen Wiederholungsanweisungen vorkommen • lineares Suchen • Sortieren durch Auswählen • Wurzelberechnung nach Heron von Alexandria • mono- oder polyalphabetische Substitution • Permutation von Elementen 	<ul style="list-style-type: none"> • Programme, in denen Wiederholungsanweisungen vorkommen • lineares Suchen • Sortieren durch direktes Einfügen • Wurzelberechnung nach Heron von Alexandria • mono- oder polyalphabetische Substitution • Permutation von Elementen • Türme von Hanoi • Quicksort • Damenproblem • Operationen für einfach verkettete Listen und binäre Bäume • Suchbaum

Darüber hinaus wird Rekursion beim logik-orientierten Programmieren mit PROLOG thematisiert (ein Wahl-Themenbereich im Grundfach, verpflichtend im Leistungsfach).

3 Testaufgaben und Musterlösungen

Der Test wurde zu Beginn der Klassenstufe 12 geschrieben². Er befasste sich mit einem Thema, das laut Lehrplan schwerpunktmäßig Gegenstand der Klassenstufe 11 ist. Nachfolgend werden die elf Aufgaben vorgestellt, die an den fünf Gymnasien zu bearbeiten waren (einschließlich *Musterlösungen, die kursiv eingefügt wurden*, und Kommentaren). Während der Bearbeitung des Tests waren keine Hilfsmittel zugelassen.

1. Aufgabe:

Fügen Sie die Wörter „Schachtelung“ oder „Aneinanderreihung“ nachfolgend ein:

Rekursion ist Wiederholung durch: <i>Schachtelung</i>

Iteration ist Wiederholung durch: <i>Aneinanderreihung</i>
--

Kommentar: Mit dieser Aufgabe soll Faktenwissen zur Struktur einer rekursiven bzw. iterativen Problemlösung überprüft werden. Die Wahrscheinlichkeit zum zufälligen Treffen der korrekten Zuordnung ist hoch. Daher können die Schülerantworten zu dieser Aufgabe nicht isoliert, sondern nur im Zusammenhang mit anderen Antworten betrachtet werden.

2. Aufgabe:

Mit einer WHILE-Anweisung (WHILE-Schleife) sollen alle Zeichen eines Textes verschlüsselt werden. Geben Sie an, ob der Algorithmus ein Beispiel für Rekursion oder für Iteration ist:

Beispiel für: <i>Iteration</i>

Kommentar: Die Schülerinnen und Schüler sollen einen Algorithmus einordnen, von dem sie nur eine wesentliche Eigenschaft kennen.

3. Aufgabe:

Nennen Sie jeweils ein Sortierverfahren, das iterativ bzw. rekursiv arbeitet:

iterativ: <i>Sortieren durch Auswählen</i>
--

rekursiv: <i>Quicksort</i>

² Das ist in Thüringen die letzte Klassenstufe am allgemein bildenden Gymnasium (G8).

Kommentar: Mit dieser Aufgabe soll ermittelt werden, ob die Schülerinnen und Schüler die grundsätzliche Arbeitsweise von ausgewählten Sortierverfahren kennen.

4. Aufgabe:

Gegeben ist die folgende rekursive Definition für „binärer Baum“: Ein binärer Baum besteht aus einem Element (der Wurzel) und zwei binären Bäumen (dem linken und dem rechten Teilbaum). Die Definition ist unvollständig. Was fehlt?

<i>der Abbruch (leerer Baum)</i>

Kommentar: Die Schülerinnen und Schüler sollen erkennen, dass ein Abbruchkriterium erforderlich ist, da der Baum ansonsten unendlich groß ist.

5. Aufgabe:

Ihre Lehrerin/Ihr Lehrer führt Ihnen eine Matroschka vor. Definieren Sie den Begriff „Matroschka“ auf rekursive Art.

<i>Eine Matroschka besteht aus einer Puppe, die eine Matroschka enthält, oder es ist die kleinste Matroschka.</i>

Kommentar: Anhand eines Begriffes, der nichts mit der Informatik zu tun hat, soll festgestellt werden, ob die Schülerinnen und Schüler in der Lage sind, die ihnen bekannte Methode der rekursiven Definition eines Begriffes auf einen neuen Kontext zu übertragen. Die Schülerinnen und Schüler müssen das Wesentliche des rekursiven Definierens erkannt haben und in der Lage sein, ihre Lösung geeignet zu verbalisieren.

6. Aufgabe:

Gegeben ist das Zeichen **A**. Das Zeichen **A** kann durch das Zeichen **0** oder durch die Zeichenfolge **1A** ersetzt werden. Ergebnis-Zeichenfolgen dürfen kein Zeichen **A** enthalten. Geben Sie vier Ergebnis-Zeichenfolgen an, die nach diesen Regeln gebildet wurden:

<i>0</i>
<i>10</i>
<i>110</i>
<i>1110</i>

Kommentar: Bei dieser Aufgabe ist eine Syntaxdefinition in verbaler Form gegeben, also

nicht durch EBNF oder Syntaxdiagrammen. An dem Beispiel soll ermittelt werden, ob die Schülerinnen und Schüler das Prinzip eines rekursiven Ersetzungsmechanismus verstanden haben.

7. Aufgabe:

Anna, Bert, Clara und Dirk klingeln nacheinander an meiner Wohnungstür:

Es klingelt. Ich lasse Anna herein ▼ und sage Anna guten Tag.
 Es klingelt. Ich lasse Bert herein ▼ und sage Bert guten Tag.
 Es klingelt. Ich lasse Clara herein ▼ und sage Clara guten Tag.
 Es klingelt. Ich lasse Dirk herein und sage Dirk guten Tag.

Das Zeichen ▼ bedeutet: Bevor ich Anna, Bert und Clara begrüßen kann, klingelt es wieder. Meine Tätigkeit wird also unterbrochen. Die Handlungen werden von einem rekursiv arbeitenden Computerprogramm ausgeführt. In welcher Reihenfolge sage ich den vier Besuchern guten Tag?

1.	2.	3.	4.
<i>Dirk</i>	<i>Clara</i>	<i>Bert</i>	<i>Anna</i>

Kommentar: Die vier Gäste treffen in der Reihenfolge Anna, Bert, Clara und Dirk ein und werden in der umgekehrten Reihenfolge Dirk, Clara, Bert und Anna begrüßt. Die Schülerinnen und Schüler sollen zeigen, dass sie wissen, wie ein rekursives Computerprogramm zur Abarbeitung gebracht wird (vgl. Thüringer Abiturprüfung 2001, Grundfach Informatik (Haupttermin), Aufgabe 1.1).

8. Aufgabe:

Im Unterricht haben Sie das Spiel „Türme von Hanoi“ kennen gelernt. Bei dem Spiel sind Scheiben von einem Feld auf ein anderes in möglichst wenigen Schritten zu versetzen. Auf einem dritten Feld können Scheiben zwischengelagert werden. Beim Versetzen sind zwei Spielregeln zu beachten: Versetze immer nur eine Scheibe von einem Feld auf ein anderes und lege stets eine kleinere Scheibe auf eine größere. Beschreiben Sie einen Algorithmus zur Lösung des Problems „Türme von Hanoi“:

Das Versetzen eines Turmes vom Startfeld zum Zielfeld erfolgt in drei Schritten:

- 1. Schritt: Der Turm wird ohne seine größte Scheibe vom Startfeld zum Hilfsfeld versetzt.*
- 2. Schritt: Die größte Scheibe wird vom Startfeld zum Zielfeld versetzt.*
- 3. Schritt: Der Turm (siehe 1. Schritt) wird nun vom Hilfsfeld zum Zielfeld versetzt.*

Die drei Schritte werden rekursiv auf jeden Turm, der zu versetzen ist, angewandt. Der Abbruch liegt vor, wenn der Turm leer ist.

Kommentar: In dieser Aufgabe sollen die Schülerinnen und Schüler einen Algorithmus wiedergeben, der im Unterricht behandelt wurde. Dabei kann z. B. der rekursive Begriff „Turm“ verwendet werden: Ein Turm besteht aus der größten Scheibe und dem Rest-Turm oder es ist der leere Turm.

9. Aufgabe:

a) Gegeben ist der folgende Suchbaum (siehe Abbildung 1).

Bei der Methode „Inorder“ wird zuerst der linke Teilbaum ausgegeben, dann werden die Daten der Wurzel ausgegeben und anschließend wird der rechte Teilbaum ausgegeben. Diese Vorschrift wird rekursiv auf jeden Teilbaum angewandt. Bei einem leeren Teilbaum wird nichts ausgegeben. Die Methode „Inorder“ wird auf den gegebenen Suchbaum angewandt. In welcher Reihenfolge werden dann die Elemente des Suchbaums ausgegeben?

1.	2.	3.	4.	5.	6.	7.	8.
10	22	31	42	58	62	65	70

b) Gegeben ist ein beliebiger Suchbaum. Die Elemente des Suchbaumes werden mithilfe der Methode „Inorder“ ausgegeben. In welcher Reihenfolge werden die Elemente ausgegeben? Begründen Sie Ihre Antwort.

Die Elemente werden in sortierter Reihenfolge von der kleinsten bis zur größten Zahl ausgegeben.

Begründung: Bei Inorder werden immer zuerst die Zahlen im linken Teilbaum, dann die Zahl in der Wurzel und anschließend die Zahlen im rechten Teilbaum ausgegeben. Bei einem Suchbaum sind immer alle Zahlen im linken Teilbaum kleiner als die Zahl in der Wurzel und alle Zahlen im rechten Teilbaum größer als die Zahl in der Wurzel. Daher ergibt sich die sortierte Reihenfolge.

Kommentar: Die Schülerinnen und Schüler sollen darstellen, welche Wirkung die rekursive Prozedur „Inorder“, die in verbaler Form gegeben ist, auf Suchbäume hat. In Teilaufgabe a) ist ein konkretes Beispiel zu bearbeiten. In Teilaufgabe b) ist nach einer Verallgemeinerung gefragt, wobei nicht explizit angegeben ist, was ein Suchbaum ist. Die Schülerinnen und Schüler sollen erkennen, dass die Elemente des Suchbaums stets in sortierter Reihenfolge ausgegeben werden. Die Begründung dafür ist eine anspruchsvolle Aufgabe.

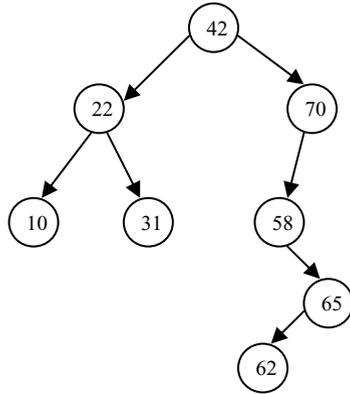


Abbildung 1: Gegebener Suchbaum in der 9. Aufgabe

10. Aufgabe:

Gegeben ist das folgende Quadrat:

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Das Quadrat wird in den 1., 2., 3. und 4. Teil zerlegt:

2. Teil	1. Teil
3. Teil	4. Teil

Die vier Teile werden wiederum in den 1., 2., 3. und 4. Teil zerlegt. Ausgaben erfolgen stets in der Reihenfolge 1. Teil – 2. Teil – 3. Teil – 4. Teil. Geben Sie an, in welcher Reihenfolge die Zahlen von 1 bis 16 des gegebenen Quadrates nach diesen Erläuterungen ausgegeben werden:

4	3	7	8	2	1	5	6	10	9	13	14	12	11	15	16
---	---	---	---	---	---	---	---	----	---	----	----	----	----	----	----

Kommentar: Die Schülerinnen und Schüler sollen zeigen, dass sie ein vorgegebenes rekursives Zerlegungsschema in einer konkreten Situation anwenden können. Die beschriebene Art der Zerlegung ist eine Möglichkeit zum Speichern von Bildern in Grafikprogrammen (vgl. [Fo05]).

11. Aufgabe:

Die 11. Aufgabe bezieht sich auf die 10. Aufgabe. Sie wird erst bearbeitet, nachdem die Schülerantworten zu den ersten zehn Aufgaben eingesammelt wurden. Die Schülerinnen und Schüler notieren sich ihre Antwort zur 10. Aufgabe zusätzlich auf einem Extrablatt. Ihnen wird die Lösung der 10. Aufgabe mitgeteilt und dann heißt es: Bitte schildern Sie, was Sie sich bei der Lösung der 10. Aufgabe überlegt haben (unabhängig davon, ob Sie die 10. Aufgabe richtig oder falsch gelöst haben).

Kommentar: Die Schülerinnen und Schüler sollen sich mit ihrer Antwort zur 10. Aufgabe anhand der korrekten Lösung auseinandersetzen. Damit soll ein Reflexionsprozess initiiert werden.

In der folgenden Tabelle ist zusammengestellt, welche Teilkompetenzen mit den einzelnen Aufgaben überprüft werden sollen:

Aufgabe	1	2	3	4	5	6	7	8	9	10/11
Kompetenzen	(a)	(a) (d)	(d) (e)	(b)	(b)	(c)	(a) (d) (e)	(d) (e)	(b) (d)	(d)

Alle im Abschnitt 2 aufgeführten Teilkompetenzen spielen also in dem Test eine Rolle. Quelltexte von Programmen sind jedoch weder zu analysieren noch zu entwickeln.

4 Durchführen und Korrigieren des Tests

Die Bearbeitung der Testaufgaben erfolgte innerhalb einer Unterrichtsstunde (45 min). Die Schülerinnen und Schüler kannten das Thema des Tests, waren auf ihn jedoch nicht speziell vorbereitet. An den beteiligten Schulen wurde Informatik in der Klassenstufe 11 regelmäßig erteilt. Vor dem Schreiben des Tests stellten die Lehrkräfte für jede Aufgabe die Vorleistungen aus dem Unterricht zusammen. Sie korrigierten die Antworten ihrer Schülerinnen und Schüler und ermittelten für jede Aufgabe die Anzahl an Schülern, die die Aufgabe gelöst, teilweise gelöst bzw. nicht gelöst haben. Sie setzten für jede Aufgabe das Korrekturergebnis in Beziehung zu den Vorleistungen aus dem Unterricht und gaben Auffälligkeiten wie z. B. typische Fehler an. Unmittelbar nach dem schriftlichen Test wurden einzelne Schülerinnen und Schüler von ihrer Lehrerin oder ihrem Lehrer zu ihren Antworten befragt.

5 Reflexionen

Nach Auswertung des Tests hat der Autor die beteiligten sechs Lehrkräfte im Oktober/November 2004 in leitfadenorientierten Interviews befragt. Im Mittelpunkt der Interviews standen Nutzen, Aufwand, Probleme und Varianten des Vorgehens. Nachfolgend sind wesentliche Ergebnisse der Interviews zusammengefasst.

5.1 Vorgehen

Die Lehrerinnen und Lehrer sahen das Vorgehen, interessierten Schulen Testaufgaben zu übergeben, als sinnvoll an. Dafür wurden unterschiedliche Gründe genannt. So werden externe Aufgabenstellungen als objektiver betrachtet als selbst zusammengestellte (im Sinne der Vorbereitung auf die Anforderungen im Zentralabitur). Die Aufgaben und Musterlösungen geben Hinweise zum im Unterricht anzustrebenden Anforderungsniveau. Ein Vergleich mit anderen Schulen wird möglich. Mehrere Lehrer erklärten, dass durch die Aufgaben die eigene Schwerpunktsetzung zum Teil bestätigt wurde. Die Aufgaben lieferten aber auch Ideen für die künftige Unterrichtsgestaltung. Betont wurde insbesondere die Bandbreite der Anforderungen. Ein Lehrer wies darauf hin, dass er an seiner Schule der einzige Informatiklehrer ist, so dass er die Aufgaben als hilfreiche Unterstützung ansah.

5.2 Konsequenzen für den eigenen Unterricht

Die beteiligten Lehrerinnen und Lehrer erklärten, dass sie aus den Testergebnissen Rückschlüsse auf den Lernstand ihrer Schülerinnen und Schüler ziehen konnten. Teilweise waren sie von den Schülerantworten positiv oder negativ überrascht. Den Lehrkräften war es möglich, Schlussfolgerungen sowohl für die weitere Arbeit in dem Kurs als auch allgemein für ihren künftigen Unterricht abzuleiten. Das Verhältnis von Aufwand und Nutzen (Schreiben und Auswerten des Tests auf der einen Seite – gewonnene Erkenntnisse auf der anderen Seite) wurde als angemessen angesehen. Die Lehrkräfte regten an, für weitere Gebiete des Informatikunterrichts in der gymnasialen Oberstufe Testaufgaben zu entwerfen und interessierten Schulen bereitzustellen (konkret genannt wurden: Kommunikation in Netzen, Algorithmenstrukturen, einfache und strukturierte Datentypen, Sortieren und Suchen sowie Möglichkeiten und Grenzen des Einsatzes von Informatiksystemen).

5.3 Design der Aufgaben

Die einbezogenen Lehrkräfte schätzten die Testaufgaben nach ihrer Eignung zum Überprüfen der Teilkompetenzen, nach deren Schwierigkeitsgrad und der Qualität der Formulierung ein. Sie unterbreiteten die folgenden Änderungsvorschläge: Bei der 1. Aufgabe sollten künftig keine Begriffe mehr vorgegeben werden, um das Raten einzuschränken. Wichtige Voraussetzung für das Lösen der 3. Aufgabe war die Kenntnis der Namen von Sortierverfahren. Wichtiger als die Namen sind jedoch die Ideen, die den Verfahren zugrunde liegen. Daher sollten die Ideen vorgegeben werden und die Schülerinnen und

Schüler entscheiden dann, ob die Algorithmen iterativ oder rekursiv arbeiten. Die 7. Aufgabe greift eine Alltagsszene zu sehr gekünstelt auf. Eine Variante im Teil a) der 9. Aufgabe ist die Vorgabe eines beliebigen binären Baums (also keines Suchbaums), damit die Schülerinnen und Schüler nach dem Bearbeiten des einfachen linken Teilbaums (10, 22, 31) für den schwierigen rechten Teilbaum die Reihenfolge (58, 62, 65, 70) nicht raten können. Der Aufgabenteil b) wäre dann anders zu gestalten. Eine andere Variante ist die Vorgabe eines „vollständigen“ binären Baums, der bei einer Höhe von 4 aus 15 Knoten besteht. Die Lehrerinnen und Lehrer unterbreiteten weitere Aufgabenvorschläge zum Thema „Rekursion und Iteration“. Diese beziehen sich auf rekursiv definierte Grafiken, auf konkrete Algorithmen (z. B. die iterative und rekursive Berechnung von Potenzen x^n) und auf weitere Eigenschaften rekursiver Problemlösungen (z. B. Selbstaufruf von Unterprogrammen).

5.4 Musterlösungen

Grundlage der Korrekturen waren die Musterlösungen aus Abschnitt 3. Die Lehrerinnen und Lehrer sehen Musterlösungen als sinnvoll an, weil durch sie eine einheitliche Bewertung gefördert wird und weil sich der Zeitaufwand bei der Korrektur verringert. Mehrere Lehrer schlugen vor, Schülerantworten nicht nur in einer dreistufigen Skala (Aufgabe gelöst, teilweise gelöst bzw. nicht gelöst) zu bewerten, sondern wie üblich mit Punkten. Weitergehende Auswertungshinweise – über Musterlösungen hinaus – wünschte kein Lehrer. Die Lehrkräfte interviewten jeweils ein bis drei Schüler zu ihren Antworten und schätzten die ergänzenden Interviews als sinnvoll und wichtig ein. Wenn mehrere Schüler interviewt werden konnten (das war aus schulorganisatorischen Gründen nicht immer möglich), wurde Wert darauf gelegt, dass diese Schüler unterschiedliches Leistungsvermögen besitzen.

6 Ausblick

Die Voruntersuchung erfolgte unter idealen Bedingungen. So arbeitet der Autor mit einigen der beteiligten Lehrerinnen und Lehrern bereits seit Jahren in der Lehrerfortbildung zusammen, das Thüringer Institut für Lehrerfortbildung, Lehrplanentwicklung und Medien (ThILLM) Bad Berka ermöglicht Arbeitstreffen der Beteiligten und an allen einbezogenen Schulen gilt der gleiche Informatiklehrplan. Die Ergebnisse der Voruntersuchung ermutigen zur Fortführung der Arbeiten. Im Jahr 2005 wird der Test überarbeitet und allen interessierten Schulen im deutschsprachigen Raum bereitgestellt. In der zweiten Runde sollen also mehr Schulen als bisher einbezogen werden, um die Ergebnisse auf eine breitere Datenbasis zu stellen. Es soll auch ermittelt werden, in welchem Maße das dargestellte Szenarium greift, wenn die gleichen Aufgaben an Schulen mit einheitlichen Prüfungsanforderungen, jedoch unterschiedlichen Lehrplänen bearbeitet werden. Nähere Informationen sind unter <http://www.informatiktest.de> erhältlich.

Literaturverzeichnis

- [Ad04] Adler, J.: Research on mathematics teacher education: Mirror images of an emerging field. In: ICME-10 2004, Abstracts, S. 13-15.
- [EPA04] Einheitliche Prüfungsanforderungen in der Abiturprüfung Informatik. Beschluss vom 1.12.1989 i. d. F. vom 5.2.2004. Luchterhand München, Neuwied 2004.
- [Fo05] Fothe, M.: Rekursion. Ein Thema für den Informatikunterricht. In: LOG IN Heft Nr. 133 (2005), S. 46-54.
- [KI03] Klieme, E. u. a.: Zur Entwicklung nationaler Bildungsstandards. Eine Expertise. Bildungsreform Band 1. BMBF, Berlin 2003.
- [TKM99] Lehrplan Informatik für das Gymnasium. Thüringer Kultusministerium, Erfurt 1999.

Annemarie List, Martina Zidek, Klaus Hohmann, Falk Möckel, Wolfgang Moldenhauer, Thomas Stoll und Günter Straßburg danke ich für die angenehme und konstruktive Zusammenarbeit innerhalb des Projektes zur Schulentwicklung.