

● Ein Java-Applet zur Eingabe und Überprüfung mathematischer Terme

Rudolf Großmann, Stein

Das hier vorgestellte Java-Applet erlaubt einerseits die Darstellung (Ausgabe) von Termen mit Brüchen, Wurzeln und Funktionen auf HTML-Seiten, andererseits die Eingabe eines Terms und die Überprüfung auf Übereinstimmung mit einem vorgegebenen Lösungsterm. Dabei werden auch äquivalente Terme als Lösung akzeptiert. Als Anwendung können Übungsaufgaben auf Datenträger oder im Internet bereitgestellt werden, bei denen der Schüler sofort eigenständig die Richtigkeit seines Ergebnisses überprüfen kann. Zur Erstellung solcher Übungsaufgaben sind nur grundlegende HTML-Kenntnisse notwendig. Das Applet wird laufend fortentwickelt und kann zu Testzwecken über die Homepage des Autors [1] bezogen werden.

1 Einleitung

Schlägt man ein beliebiges Algebra-Buch auf, stößt man nahezu zwangsläufig auf sogenannte "Aufgaben-Plantagen". Das Durcharbeiten im herkömmlichen Stil (ein Schüler rechnet vor, die anderen schreiben von der Tafel ab — oder noch schlimmer: der Lehrer rechnet vor, um "Zeit zu sparen") führt bei unserer medien-verwöhnten Schülerschaft schnell zu Langeweile. Der Spruch "Wenn alles schläft und einer spricht, das Ganze nennt man Unterricht", ist schon alt. Manche Kollegen lockern den Unterricht mit Gruppenarbeit auf, oder sie verpacken die Algebra-Aufgaben in Spielchen und verkaufen das Ganze als "Freiarbeit". Zu Recht wird eine neue Schwerpunktsetzung in der Aufgabekultur gefordert.

Wie dem auch sei, auf Übungsphasen als Anwendung der Theorie wird man bei keinem Unterrichtsstil verzichten können. Es genügt nicht, den Schülern die Vorzeichen-Rechenregeln bei der Multiplikation nur mitzuteilen (oder herzuleiten) — man muss sie einüben — mit mehr als zwei Faktoren, mit Variablen, mit Abgrenzung von den Rechenregeln für die Addition, mit Wiederholung des Bruchrechnens usw. Bei solchen Übungsphasen ist es wünschenswert, den Großteil der Schüler aus der Passivität des bloßen Abschreibens von der Tafel herauszuholen. Möglichst viele Schüler sollten den Aufgaben-Fundus gleichzeitig unabhängig voneinander bearbeiten können und unmittelbar eine Rückmeldung in der Form "richtig/falsch" erhalten. Diesem Ziel kommen die Freiarbeits-Aufgaben der ALP Dillingen (Lippert et

al., 1999) sehr nahe. Sie bieten Aufgaben auf der Vorderseite und Lösungen zur Selbstkontrolle auf der Rückseite. Eine Umsetzung dieses Konzeptes auf HTML-Basis ist mit den unten beschriebenen Java-Applets möglich.

An unserer Schule müssten die Schüler noch in den (inzwischen gut ausgebuchten) Informatikraum umziehen, und es stünde nur ca. ein Rechner für zwei Schüler zur Verfügung. Wenn sich die Entwicklung der letzten Jahre auf dem Computer-Markt wie bisher fortsetzt, kann man voraussehen, dass in einigen Jahren die Schüler mit dem (eigenen) Notebook im Unterricht ebenso umgehen wie heute mit dem Taschenrechner, noch dazu über Funk miteinander und mit dem Internet vernetzt. Man mag dieser Entwicklung durchaus kritisch gegenüberstehen, wie Clifford Stoll in seinem lesenswerten Buch "LogOut" (Stoll, 2001) — aufhalten wird man sie nicht. Wir müssen uns für das Zeitalter des Computer Aided Learning (CAL) rüsten.

2 Das Trio der Formel-Applets

2.1 Das Formelausgabe-Applet

Es gibt viele Möglichkeiten, mathematische Formeln auf Seiten im WWW darzustellen. Zum Beispiel existiert mit MathML [2] ein HTML-Abkömmling speziell für die Darstellung mathematischer Inhalte im Internet. Leider verstehen nur einige spezielle Browser, z.B. Amaya [3] MathML. Hier wären die Browser-Hersteller — allen voran Microsoft — aufgefordert, endlich MathML-Unterstützung in ihre Browser zu integrieren.

Mit "Hot Equation" [4] der Ruhr-Universität Bochum gibt es ein sehr leistungsfähiges Applet, das TEX-Input in Formeln umsetzen kann.

Schließlich kann man Formeln innerhalb eines Dokuments von MS Word mit dem Formel-Generator erstellen und das Dokument als HTML-Datei speichern. Die Formeln werden dann in Bitmap-Grafiken umgewandelt und an den entsprechenden Stellen im Text eingebunden. Word erzeugt allerdings einen HTML-Code, der nur schwer von Hand nachzubearbeiten ist. Daher empfehle ich die letzte Methode nicht.

Bei so vielen schon existierenden Möglichkeiten bräuchte man das Formelausgabe-Applet gar nicht. Es bietet jedoch den Vorzug eines einheitlichen Designs für die Eingabe und Ausgabe. Durch Anpassung der Hintergrundfarbe an den Hintergrund der umgebenden HTML-Seite wird die rechteckige Applet-Begrenzung gleichsam unsichtbar.

2.2 Das Formeleingabe-Applet

Das Eingabe-Applet erlaubt neben der Verwendung der vier Grundrechenarten die Eingabe von Klammern, Brüchen, Quadratwurzeln und der Funktionen sin, cos, tan, ln, lg und Betrag. So lange die Eingabe nicht abgeschlossen ist, ist der Hintergrund des Applets hellblau. Die Eingabe wird mit der Enter-Taste oder dem Drücken des optionalen OK-Knopfes abgeschlossen. Gleichungen können (noch) nicht eingegeben werden.

Das Applet reagiert vorläufig mit Grünfärbung des Hintergrundes auf richtige Eingaben und mit Orangefärbung auf falsche Eingaben. Alle Farben können mit in der HTML-Seite eingebettetem JavaScript-Code dem persönlichen Geschmack angepasst werden. Die Reaktionsfähigkeit soll in zukünftigen Versionen stark erweitert werden, so dass eine differenzierte Reaktion auf verschiedene Arten von Falscheingaben möglich ist, beispielsweise der Sprung zu anderen HTML-Seiten je nach Fehler-Art.

Abb. 1 zeigt ein Beispiel einer Übungsseite mit Eingabe- und Ausgabeapplet.

Additionstheoreme - Anwendung

Vereinfache:

$$\frac{\cos 2x}{1 - (\tan x)^2} = \frac{(\cos x)^2 - (\sin x)^2}{1 - \left(\frac{\sin x}{\cos x}\right)^2} = \frac{(\cos x)^2 - (\sin x)^2}{\cos^2 x - \sin^2 x}$$

Tipp 3

Forme als erstes die Differenz im Nenner in einen Bruch um. Denke dir dazu den Minuenden 1 als "ein Eintel" geschrieben und bringe Minuend und Subtrahend auf den Hauptnenner. Löse in einem zweiten Schritt den entstehenden Doppelbruch auf und kürze.

[Tipp 4](#)

Wert der Rest-Aufgabe: 5 BE

[zur Index-Seite](#)

Abb. 1: Das Formelausgabe- und Eingabe-Applet

Das Eingabe-Applet kennt folgende Sondertasten:

Taste	Funktion
Strg-W	Wurzel
Strg-B	Bruchstrich
Cursor hoch	Exponent

$5\frac{1}{2}$ wird wie folgt eingegeben: 5, Strg-B, 1, Cursor runter, 2. Das Eingabe-Applet erlaubt auch eine Ansteuerung mit beschrifteten Knöpfen als Eingabehilfe, so dass man sich die Sondertasten nicht merken muss.

2.2 Das Editor-Applet

Das Editor-Applet benötigt nur, wer selbst Übungsseiten mit Ein- und Ausgabeapplets erstellen will, also z.B. der Lehrer als Autor von Übungsaufgaben, nicht der Schüler als Leser von Übungsaufgaben.

```
<APPLET CODEBASE="classes" ARCHIVE="gf02.jar"
CODE="gutformel/gutfa_in.class"
HEIGHT="80" WIDTH="200" NAME="rechts01"
ALIGN="middle">
<PARAM NAME="FORMEL" VALUE="aced00057372001...
...071007e0010">
<PARAM NAME="OKBUTTON" VALUE="false">
</APPLET>
```

Das Eingabe-Applet erscheint im HTML-Quellcode z.B. so:

Abb. 2: HTML-Code des Eingabe-Applets

Der Term, den das Ausgabe-Applet darstellen soll, bzw. der Term, den das Eingabe-Applet als Lösung akzeptieren soll, wird dem Applet in Form eines Textes als Parameter übergeben (in Abbildung 2 kursiv dargestellt). Dieser Parameter stellt, hexadezimal codiert, den Term dar. Das Editor-Applet erzeugt als Output genau diesen Code-Text, der dann

im HTML-Code der Übungsaufgabe eingebettet wird. Das klingt komplizierter als es ist.

Das Editor-Applet muss diesen Code-Text auf Festplatte speichern oder über die Zwischenablage exportieren können. Damit es das darf, wurde das Applet von mir digital signiert. Wird es geladen, muss der Benutzer, d.h. der Autor von Übungsseiten, ihm erst die Erlaubnis dazu erteilen (siehe Abbildung 3).

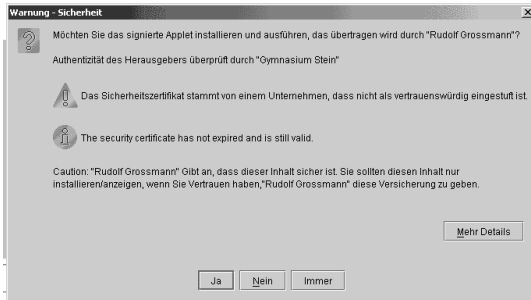


Abb. 3: Das Editor-Applet bekommt Schreibrechte

Das ist eigentlich nichts Besonderes im Vergleich dazu, dass z.B. jedes Installationsprogramm unter Windows in der Regel beliebige Dateien anlegen und ausführen darf. Wer Vertrauen in Shareware unbekanntem Ursprungs hat, kann auch dem Editor-Applet erlauben, seinen Formel-Code-Text in die Zwischenablage zu exportieren.

3 Wieso gerade Java?

3.1 Vorteile von Java

3.1.1 Betriebssystemunabhängigkeit

Java ist eine betriebssystemunabhängige Programmiersprache, d.h. das gleiche Java-Applet läuft unter Windows ebenso wie unter Linux oder auf einem Macintosh, vorausgesetzt, es steht, wie momentan üblich, ein Browser zur Verfügung, der Java unterstützt.

3.1.2 Einbindung in HTML

Durch die Einbindung des Applets in HTML stehen die vielfältigen Möglichkeiten der Seitengestaltung zur Verfügung, die HTML bietet. Die "Auslieferung" der Seiten kann über das Internet erfolgen.

3.1.3 Erweiterbarkeit mit JavaScript

Die Funktionalität (z.B. differenzierte Reaktion auf Falsch-Eingaben) kann durch die Bereitstellung geeigneter Schnittstellen mittels der Sprache JavaScript durch den Benutzer (Autor von Übungsseiten) erweitert werden, ohne dass dieser in Java programmieren

muss. Verwechseln Sie JavaScript [5] nicht mit Java.

3.2 Nachteile von Java

3.2.1 Zeit- und Speicherprobleme

Java hat den Ruf, langsam und speicherhungrig zu sein. Tatsächlich zeigt das Eingabe-Applet auf Rechnern älterer Bauart merkbare Reaktionszeiten auf Eingaben. Es läuft gut auf Rechnern ab ca. 500 MHz Taktfrequenz. Wenn die Entwicklung immer schnellerer Prozessoren und der Preisverfall bei Arbeitsspeicher so weitergeht wie bisher, dürfte diese Thematik kein Problem darstellen.

3.2.2 Die falsche Java-Version?

Bei der Verwendung von Java taucht manchmal folgendes Problem auf: Das Applet ist mit Version Y programmiert und kompiliert worden, der Browser des Benutzers, d.h. des Schülers, verwendet aber die ältere Java-Version X und findet ein paar Java-"Classes" nicht. An der Stelle des Applets bleibt der Bildschirm leer. Um diese Situation zu vermeiden, habe ich bei dem Ausgabe- und Eingabe-Applet das "alte" Java 1.1.8 verwendet. Nur das Editor-Applet setzt eine neuere Java-Version (ab 1.3) voraus.

3.2.3 Microsoft boykottiert Java

Mit Windows XP wird als Standard-Browser ein Internet Explorer ausgeliefert, der kein Java unterstützt — passend zur Firmenpolitik Microsofts — ärgerlich für die Benutzer. Wer als Autor von Übungsaufgaben das Editor-Applet verwenden will, muss seinen Browser mit einer Java-Version 1.3 oder neuer ausstatten. Man kann sich hierzu die neueste Java-Version von SUN herunterladen [6]. Für unsere Zwecke genügt die "Java Runtime Engine" (JRE). Sie ist für Windows ca. 8 Megabyte groß. Nur wer selbst in Java programmieren will, braucht das "Java Development Kit" (JDK). Man kann alternativ auch auf den Begleit-CD's der aktuellen Computer-Fachzeitschriften nach den Browsern "Opera" oder "Mozilla" (Netscape-Nachfolger) suchen. Die Installationsprogramme dieser Browser installieren in der Regel Java mit und versorgen auch den Microsoft Internet Explorer mit dem Java-Plugin von SUN.

3.2.4 Speichern verboten?

Das Sicherheits-Konzept von Java verhindert zunächst, dass Applets lesend oder schreibend auf die lokale Festplatte oder die Zwischenablage zugreifen. So wird verhindert, dass Java-Applets dazu missbraucht werden, bösartigen Code (Viren, Würmer,...) beim Surfen im Internet einzuschleusen. Wie in 2.2 beschrieben, muss der Benutzer des Editor-Applets erst zustimmen, dass das von mir digital signierte Applet diese Rechte erhält. Damit die Rechte-Verwaltung klappt, ist, wie oben erwähnt, eine neuere Java-Version (ab 1.3) nötig. Meine digitale Signatur könnte ich wiederum von einer Firma wie z.B. VeriSign signieren lassen, die dann dafür garantiert, dass die Signatur wirklich von mir stammt. Das kostet allerdings nicht unerhebliche Gebühren, die nicht einmalig, sondern jährlich erhoben werden. Sie haben sicherlich Verständnis, dass ich eine Signatur verwende, die nicht von einer „Certification Authority“ signiert wurde.

3.3 Pro und contra Java

Resümee: Die Nachteile von Java lassen sich mit vertretbarem Aufwand alle überwinden. Die Vorteile überwiegen. Daher fiel meine Entscheidung zugunsten von Java als zu verwendender Programmiersprache für meine Zwecke.

4 Zielgruppen für die Formel-

4.1 Der engagierte Mathelehrer

Jeder interessierte Kollege kann sich bereits jetzt (September 2003) von meiner Homepage [1] eine Vorab-Version der drei Formel-Applets mit Beispielen herunterladen. Das Editor-Applet (nicht das Eingabe-Applet) hat einen eingeschränkten Funktionsumfang: Es erlaubt nicht die Eingabe von Quadratwurzeln und Funktionen. Man kann damit Übungsseiten erstellen, die einen Großteil der gymnasialen Applets Algebra bis zur 8. Jahrgangsstufe abdecken.

4.2 Der Schulbuch-Verlag

So manches Übungsprogramm aus dem Edutainment-Sektor entspricht dem Schema "viel Verpackung, wenig Inhalt". Die Lernsoftware ist oft multimedial mit Animationen und Sprachausgabe aufgepeppt. Die eigent-

lichen Übungen sind dann bessere Multiple-Choice-Tests, decken nur Arithmetik ab oder akzeptieren nur sture Lösungen in Texteingabe-Fenstern, sagen z.B. bei $a+b$ "richtig", aber bei $b+a$ "falsch".

Bisher kenne ich nur zwei Titel mathematischer Lernsoftware, die einigermaßen flexibel auf die Eingaben der Schüler reagieren.

Erstens: "Ali, der Mathemaster" (Heureka-Klett, 1998). "Ali" ist zwar schon alt; ihn gab es schon für den legendären Commodore C 64. Das Programm erlaubt aber sogar die Eingabe eigener Aufgaben, die es dann auf Wunsch vorrechnet!

Zweitens: Das "Telekolleg Algebra" aus dem Lernpaket "Mathe und Physik" des Franzis-Verlags teilt Ergebnisse nicht nur in "richtig" oder "falsch" ein, sondern kommentiert eine Eingabe auch mal mit "Das ist zwar richtig. Du kannst aber weiter vereinfachen". Eine Weiterentwicklung des Eingabe-Applets könnte in Verbindung mit JavaScript eine ähnliche Funktionalität bieten.

Genau auf ein Schulbuch abgestimmte Begleitsoftware gibt es bereits im Fremdsprachenbereich (z.B. "Green Line" von Klett). Mit den Formel-Applets lassen sich Übungsseiten erstellen, die dementsprechend auf ein Algebra-Buch abgestimmt sind.

Ähnlich wie bei einem Vokabel-Trainer könnte der Lernerfolg in einer Datenbank mitnotiert und grafisch aufbereitet dargestellt werden. Aufgabentypen mit vielen Fehlern werden häufiger, gut beherrschte Aufgaben seltener abgefragt.

4.3 Die Fachdidaktik Mathematik

An der Universität Bayreuth wurde ein Java-Applet namens "GeoNeXt" entwickelt [7], mit dem man dynamische Geometrie betreiben kann, ähnlich wie mit "Euklid", "Cinderella", "Cabri Geometre" usw.

Was "GeoNeXt" für die Geometrie leistet, könnte das Formel-Applet bei entsprechender Weiterentwicklung für die Algebra leisten. Denkbar wäre ein im Team entwickeltes mathematisches Lehrwerk, das auf CD oder über das Internet bereitgestellt wird.

4.4 Last but not least — die Schüler

Die Schüler benutzen nur Ein- und Ausgabe-Applet, basierend auf Java 1.1.8, so dass die

Übungsseiten auch auf älteren Browsern dargestellt werden können.

5 Die Zukunft der Formel-Applets

5.1 Die endlose Geschichte

Wann ist ein Programm fertig? Eigentlich nie. Wie lange gibt es "Word"? Und immer noch erscheint beinahe jährlich eine neue Version. Und immer noch stolpert es manchmal bei den Fußnoten.

Auch Hobby-Programmier-Projekte sind in der Gefahr, nie fertig zu werden. Zu oft haben Fremde Wünsche nach neuen "Features", die man dann aus Ehrgeiz versucht, zu verwirklichen. Dabei vergisst man allzu oft, dass der Zeitaufwand für das Programmieren zu ca. 90 Prozent aus Fehlersuche besteht.

5.2 Was noch verwirklicht wird

Im Formeleingabe-Applet lassen sich mit der Backspace-Taste einzelne Variablen und Ziffern von Zahlen löschen, jedoch z.B. keine Rechenzeichen. Hat man sich so vertippt, dass die Struktur des Terms nicht stimmt, sollte man die ganze Formel mit der Entfernen-Taste löschen und neu mit der Eingabe beginnen. Das Eingabe-Applet ist in der gegenwärtigen Version noch zu wenig fehlertolerant gegen Fehleingaben, als dass es bereits für den Einsatz im harten Unterrichtstag taugen würde. (Stand: September 2003)

Für die 10. Jahrgangsstufe fehlen noch Logarithmen zur allgemeinen Basis b und höhere Wurzeln.

Der Definitionsbereich von Termen darf bisher nicht eingeschränkt sein.

5.3 Was das Formel-Applet nie werden wird

Das Formel-Applet wird nicht zu einem Computer-Algebra-System (CAS) wie "Mathematica", "Derive", "MuPAD" usw. ausgebaut werden. Es gibt schon genug CAS. Die Verwendung eines CAS in der Unter- und Mittelstufe ist auch didaktisch wenig sinnvoll, da die Einstiegshürden durch die spezielle Umgebung, Syntax usw. viel zu hoch sind. Außerdem wird zwar der Output in herkömmlicher Schreibweise (Bruchstriche, Wurzeln

usw.) dargestellt, aber nicht der Input. Der geschieht meist in einer Textzeile und ist daher fremdartig im Vergleich zur Tafel-Schreibweise.

5.4 Was noch verwirklicht werden könnte

In loser Reihenfolge: Periodenstrich, Griechische Buchstaben, Konstante Pi, Grad, physikalische Einheiten, Gleichungen und Ungleichungen, Integralzeichen, Summenzeichen, Taste "EE" für Zehnerpotenzen, Indices.

Sicher haben Sie noch ein paar Anregungen für mich.

E-Mail: rudolf.grossmann@odn.de

Literatur

Lippert, Gerhard und Wolfram Thom (Hrsg.) (1999): Freies Arbeiten am Gymnasium, Band 2, Akademiebericht Nr. 330. Dillingen: Akademie für Lehrerfortbildung und Personalführung

Neidhardt, Wolfgang und Thomas Oetterer (2000): Geonet... und die Geometrie lebt! Bamberg, C.C. Buchners Verlag

Ostermann, Peter et al. (1998): Ali, der Mathe-master. Stuttgart, Heureka-Klett

Stoll, Clifford (2001): LogOut. Frankfurt: Fischer

Versch. Autoren (2001): Lernpaket Mathe & Physik. Poing, Franzis

Internet

[1] Testversion des Formel-Applets:

www.users.odn.de/~odn22355/formelapplet

www.fuemo.de/formelapplet

[2] MathML Dokumentation (engl.):

www.w3.org/Math/

Informationen zu MathML (deutsch)

www.formelbaska.com/deutsch/foba_mathml.html

[3] Amaya Homepage (Browser mit MathML-Unterstützung):

www.w3.org/Amaya/

[4] Hot Equation Java-Applet

www.esr.ruhr-uni-bochum.de/VCLab/software/HotEqn/HotEqn.html

[5] Vergleich von JavaScript mit Java

www.bluedom.de/bluedom/wissen_javascript.php

[6] Java-Plugin von SUN:

java.sun.com

[7] GeoneXt Java-Applet und -Applikation:

www.geonext.de