

Das bietet moodle

Vielfältige Umsetzungsmöglichkeiten von
Blended Learning Szenarien in einer Plattform

Dr. Guido Rößling

FB Informatik – RBG

TU Darmstadt

guido@informatik.tu-darmstadt.de

Leistungsumfang: Studierende

- ◆ Zugriff auf Arbeitsmaterialien (*Folien, Übungsblätter, Filme, ...*)
- ◆ Foren
- ◆ Online-Abgabe von Aufgaben durch Studierende
 - ◆ Bewertung im Portal mit automatischer Benachrichtigung
- ◆ Übersicht zu allen eigenen Bewertungen
- ◆ Abstimmungen, Wissenstests, Blogs, Wiki und Kalender
- ◆ *Social Presence*: Anzeige gerade online befindlicher Nutzer
- ◆ ... und einiges mehr!

Leistungsumfang: Lehrende

- ◆ Einfache Übertragung von Kursinhalten auf Folgesemester
 - ◆ *Ganz oder teilweise*
- ◆ Festlegung der bewerteten Elemente
 - ◆ *Nur Aufgaben, oder beispielsweise auch Forenbeiträge?*
- ◆ Einstellbare Formeln zur Berechnung der Gesamtpunktzahl
 - ◆ *Unterschiedliche Gewichtungen, Auslassungen, ...*
- ◆ Bereitstellung (fast) beliebiger Inhalte
- ◆ Guter Überblick über studentische Leistungen

Zusätzliche Erweiterungen

- ◆ Online: <http://moodle.org/mod/data/view.php?id=6009>
- ◆ Installation:
 - ◆ Herunterladen der Erweiterung (als „.zip“)
 - ◆ Auspacken im korrekten Verzeichnis *auf dem Server*
 - ◆ Aufrufen der Seite „Mitteilungen“ *als Moodle-Administrator*
- ◆ Derzeit etwa 691 (tw. inaktive) Erweiterungen

Erweiterungen für Studierende

- ◆ Direktes Feedback zum Bearbeitungsstatus von Aufgaben
 - ◆ *noch offen, Abgabeende naht, abgegeben, verpasst*
 - ◆ Erlaubt auch direkten Zugriff auf die Aufgaben
- ◆ Eintragung in Übungsgruppen
 - ◆ Auch als Lerngruppe möglich mit bis zu 7 Studierenden
- ◆ „Inhaltsverzeichnis“ zur Wochen- / Themenübersicht
- ◆ Visualisierung von Java sowie von dynamischen Verfahren
- ◆ ... noch viel mehr

Anmeldung zur Übungsgruppe

The screenshot shows a web browser window with the following elements:

- Browser Title:** Gdl-WS2009: Eintragung in die Übungsgruppen
- Address Bar:** <http://proffs.tk.informatik.tu-darmstadt.de/teaching/mod/gruppenrolment/view.php?id=539&enrol=578,528&enrolgroup=57&addenrol=558>
- Navigation Bar:** Gruppen | XING | LSF | dpunkt.verlag | Bücher | Ortsgerichtsgesetz | WiwA | AoC-TUD | Apple | Google Maps | ACM SIGCSE | News (241) | Popular
- Page Header:** TECHNISCHE UNIVERSITÄT DARMSTADT | Grundlagen der Informatik I, WS 2009/2010
- Breadcrumbs:** Lernportal RBG/Dr. Rößling > Gdl-WS2009 > Gruppen-Auswahl > Eintragung in die Übungsgruppen
- Buttons:** Gruppen-Auswahl bearbeiten, Bearbeite Gruppen
- Main Title:** Eintragung in die Übungsgruppen
- Text:** Bitte nutzen Sie diese Aktivität, um einer Übungsgruppe beizutreten. Sie können auch gleich einige Freunde mit eintragen, sofern diese bereits im Kurs angemeldet sind.
- Section: Anmeldung**
 - 1. Wen möchten Sie eintragen?**

Sie können auch Lernkollegen aus der untenstehenden Liste auswählen, um sich gemeinsam anzumelden. Wir gehen davon aus, dass Sie das vorher mit Ihren Kollegen besprochen haben. Bitte beachten Sie, dass in der Liste nur Studierende aufgeführt werden, die im Kurs eingeschrieben, aber aktuell in keiner Übungsgruppe eingetragen sind. Daher kann es sein, dass Sie nicht alle Freunde sehen können.

Mit dem X hinter dem Namen von Freunden können Sie diese wieder von der gemeinsamen Anmeldung entfernen.

 - Rößling, Guido
 - Bauer, Carsten X
 - Rothmann, Fabian X
 - Utting, Enrico X

Hinweis: Sie haben die maximale Größe einer Lerngruppe erreicht.
 - 2. Welcher Gruppe möchten Sie beitreten?**

In der folgenden Liste sind nur diejenigen Übungsgruppen verzeichnet, in denen derzeit noch Platz für Sie und ggf. Ihre Freunde ist. Je mehr Nutzer Sie gleichzeitig eintragen wollen, desto weniger Gruppen stehen daher potenziell zur Verfügung.
- Footer:** Di. 08:00-09:40 in S103/109, Tutor: Roman Jöris » Die Übung wird geleitet von Roman Jöris; erster Übungstermin ist der 20. 10. 2009 (4/30)
- Buttons:** » Eintrag vormerken «

Erweiterungen für Lehrende

- ◆ Anzeige noch zu korrigierender Aufgaben (mit Direktzugriff)
- ◆ Optionale Plagiarismusprüfung
- ◆ Übertragung von Kursen auf Folgesemester (ganz oder teilweise)
- ◆ Vereinbarung von Sprechstunden
- ◆ Rubrik-basierte Bewertung
- ◆ Einstellbare Formeln zur Berechnung der Gesamtpunktzahl
- ◆ Feedback-Fragebögen mit automatischer Auswertung
- ◆ Mehrsprachigkeit

Ausgewählte Lernszenarien

- ◆ Einzelbearbeitung von Aufgaben mit Abgabefrist
- ◆ Zusammenarbeit in einer Gruppe
- ◆ Peer Review
- ◆ Führen eines „Lerntagebuchs“
- ◆ Automatische Bewertung von Programmieraufgaben
- ◆ Simulationen von Verfahren, Algorithmen, Datenstrukturen
- ◆ Lernspiele
- ◆ Interaktives Lehr- / Lernbuch
- ◆ Verwendung einer Lernlogik

Szenario I: Abgabe mit Frist

- ◆ Nutzung des am besten passenden Aufgabentyps
- ◆ Einstellen der Abgabefrist
 - ◆ *Spätere Abgaben* können automatisch verhindert werden
- ◆ Bewertung erfolgt durch Tutor oder Lehrenden
- ◆ Studierende werden automatisch über Ergebnis informiert
- ◆ Ergebnis wird direkt bei „Bewertungen“ angezeigt

Beispiel: Aufgabenbewertung

The screenshot shows a web browser window with the URL `https://moodle.informatik.tu-darmstadt.de/mod/assignment/view.php?id=1798`. The page title is "AlgoAnim2010: Aufgabe: Abgabe zum Übungsblatt 5". The browser's address bar shows the URL and a search engine (Google). The page content includes a navigation menu at the top, a date stamp "Dienstag, 25. Mai 2010, 17:52", and a text box with instructions: "Bitte ladet hier den Quelltext des überarbeiteten Generators sowie eine Beispielanimation zum Übungsblatt 5 hoch." Below this, there are dates for availability and submission: "Verfügbar ab: Donnerstag, 1. April 2010, 08:35" and "Abgabetermin: Sonntag, 14. November 2010, 23:55". The main section is titled "Feedback zu den eingereichten Aufgaben". It features a comment from "Guido Röbling" dated "Donnerstag, 20. Mai 2010, 10:05" with a rating of "Bewertung: 14,00 / 15,00". The feedback text reads: "Hallo [redacted] vielen Dank für die Überarbeitung! Ich bin fast schon wunschlos glücklich. Folgende Anmerkungen habe ich aber noch: 1. Ist die bestimmte Position besetzt, wird die Kollisionsfunktion genutzt, um eine Folgeposition zu bestimmen, was ja auch richtig ist. Allerdings wäre es gut, hier unter den Text "...ergibt h(x) = 5" (etwa in Schritt 6 der Beispielanimation) einen Text "Position ist bereits belegt - verwende die Kollisionsfunktion" (oder so ähnlich) anzuzeigen, damit der Nutzer sieht "das geht ja gar nicht". Eventuell würde hier ein Marker auf dem Zielarray und der berechneten Zielposition helfen. 2. Die "ContentLocale" sollte "Locale.GERMANY" sein (der Ort, nicht die Sprache). 3. Die Methode "generate(..., ...)" umgeht ihr derzeit noch elegant - das ändert sich dann in der folgenden Übung 😊". Below the main feedback, there is a "PS" section: "PS. Hier noch die bisherigen Kommentare...: vielen Dank für die Animationsdateien! Ich habe noch ein paar kleinere Anmerkungen...: 1. Am Anfang (Schritt 2 und 3) legt ihr ein paar "leere" Texte an ("ikollision" und "text"). Da diese keinen Wert haben und damit "unsichtbar" sind, solltet ihr auf das "lang.nextStep" hier verzichten: der Betrachter fragt sich, was diese offenbar". The browser's status bar at the bottom shows "Online: 3", "Mittellungen: 1", and a "Logout" button.

Szenario II: Gruppenarbeit

- ◆ Variante I: normale Nutzung von Moodle
 - ◆ Ein Student reicht ein, gibt Namen der Kollegen an
 - ◆ Bewertung wird dann für alle Studierenden eingetragen
- ◆ Variante II: Nutzung des Plugins „Team Assignment“
 - ◆ Studierende tragen sich selbst in Gruppen ein
 - ◆ Nur eine Abgabe und Bewertung pro Gruppe
 - ◆ Punktzahlen werden automatisch bei allen eingetragen
- ◆ Kollaboration etwa über gruppeneigene *Foren* oder *Wikis*

Szenario III: *Peer Review*

- ◆ *Peer Review*: Studierende begutachten gegenseitig Abgaben
- ◆ Ablauf (mit Erweiterung *Peer Review*):
 1. Studierende reichen die eigene Lösung ein
 2. System weist zwei zu begutachtende Einreichungen zu
 3. Studierende bewerten anhand von Ja/Nein Rubriken
 4. Dozent bewertet optional zusätzlich
 5. Endpunktzahl berechnet für Reviews und Einreichung
- ◆ *Sinnvolle Definition der „Ja/Nein-Rubriken“ ist entscheidend*

Szenario IV: „Lerntagebuch“

- ◆ Standardelement *Journal*
- ◆ Ein einfacher Texteditor, erlaubt aber auch Formatierung
- ◆ Dozent stellt Bearbeitungszeit ein (etwa „7 Tage“)
- ◆ Studierende können an ihrem eigenen Text arbeiten
- ◆ Dozent kann am Ende die Abgabe bewerten
- ◆ Nutzbar für *Lerntagebücher, Seminartexte, Aufsätze, ...*

Beispiel: Lerntagebuch

The screenshot shows a web browser window with the URL <https://moodle.informatik.tu-darmstadt.de/mod/journal/view.php?id=3298>. The page title is "Spielwiese: Mein persönliches Lernjournal". The browser's address bar shows the URL and a search bar with "Google". The page content includes a header with navigation links like "AppleDL", "EJunior", "Login ITiCSE'11", "ITiCSE'11", "Portal", "MoodleAddOns", "AoC-TUD", "NiftyAss", "MoodleMonthly", ">Mendeley", "News (324)", "ACM SIGCSE", "Popular", and "livecollab". Below the header, there is a section titled "Zeige 1 Journaleinträge". A text box contains the instruction: "Bitte tragen Sie hier die wesentlichen Lernfortschritte und -vorgehensweisen in dieser Arbeitswoche ein, möglichst mit einer Angabe der verwendeten Zeit. Das hilft uns dabei, den Schwierigkeitsgrad der Aufgaben besser an die Studierenden anzupassen." Below this, there is a button labeled "Journaleintrag beginnen oder bearbeiten". The main content area shows two entries: "Montag" and "Dienstag". The "Montag" entry describes reading chapters 2 and 3, taking notes, and studying for a math exam. The "Dienstag" entry describes discussing open questions with a learning group and publishing them in a forum. At the bottom, there is a footer with the text "Zuletzt bearbeitet: Mittwoch, 13. Oktober 2010, 09:24 (159 Wörter)" and "Bearbeitungszeitraum endet: Donnerstag, 14. Oktober 2010, 00:00". The browser's status bar shows "Online: 3", "Mittellungen: 0", and "Du bist angemeldet als Guido Rößling (Logout)".

Spielwiese: Mein persönliches Lernjournal

<https://moodle.informatik.tu-darmstadt.de/mod/journal/view.php?id=3298> Reader Google

AppleDL EJunior Login ITiCSE'11 ITiCSE'11 Portal MoodleAddOns AoC-TUD NiftyAss MoodleMonthly >Mendeley News (324) ACM SIGCSE Popular livecollab

Konfigurationspr... Übersicht zu mei... VMware KB: Incre... Mein persönliche... Download Free Pa... DFSee disktool, f... Bearbeite Journal

[Zeige 1 Journaleinträge](#)

Bitte tragen Sie hier die wesentlichen Lernfortschritte und -vorgehensweisen in dieser Arbeitswoche ein, möglichst mit einer Angabe der verwendeten Zeit. Das hilft uns dabei, den Schwierigkeitsgrad der Aufgaben besser an die Studierenden anzupassen.

[Journaleintrag beginnen oder bearbeiten](#)

Montag

Ich habe morgens gegen 10 Uhr mit dem Lesen der Kapitel 2 und 3 begonnen. Das hat etwa zwei Stunden gedauert. Während des Lesens habe ich mir direkt Notizen am Rand der Folien gemacht. Später habe ich diese Notizen anhand des weiteren gelernten Stoffes nochmals überarbeitet und auf einem separaten Blatt die noch offenen Fragen notiert. Hierfür habe ich eine weitere Stunde aufgewendet.

Den Rest des Tages habe ich für die Mathematik-Prüfung gelernt.

Dienstag

Heute bin ich die offenen Fragen durchgegangen und habe sie mit meiner Lerngruppe besprochen. Bis auf zwei Fragen konnten dabei alle meine Fragen geklärt werden. Für die Diskussion meiner Fragen und die der anderen haben wir insgesamt etwa drei Stunden verwendet.

Meine beiden noch offenen Fragen sowie einige andere haben wir anschließend im Lernportal im Forum veröffentlicht. Innerhalb von wenigen Minuten hatten wir auf meine beiden Fragen sowie einige andere bereits Antworten.

Zuletzt bearbeitet: Mittwoch, 13. Oktober 2010, 09:24 (159 Wörter)
Bearbeitungszeitraum endet: Donnerstag, 14. Oktober 2010, 00:00

Online: 3 [Mittellungen: 0](#) Du bist angemeldet als [Guido Rößling](#) (Logout)

Szenario V: Automatische Bewertung von Programmieraufgaben

- ◆ Dozent stellt eine Aufgabe und automatisierte Tests bereit
- ◆ Studierende reichen ihren Lösungsvorschlag ein
- ◆ Varianten:
 1. Direkte Übersetzung und Test: sofortiges Feedback
 2. Übersetzung auf Anfrage durch Lehrkraft
- ◆ Automatisches Feedback: Testergebnisse (Fehlertext / „OK“)
- ◆ *Keine* automatische Punktevergabe
- ◆ Auch als Quizfrage in Wissenstests einsetzbar

Java-Programmieraufgaben

Feedback:Alpha Tester:asd

Diese Abgabe kompilieren und testen!

Guido Rilling
Dienstag, 3. August 2010, 12:57

Bewertung 100 / 100
Endbewertung: 100,00

...
Time: 0.042

OK (3 tests)

Moodle-Text-Format ?

Benachrichtigungsmails versenden

Änderungen speichern Abbrechen Speichern und nächste anzeigen Nächste

Alpha Tester
Freitag, 18. Juni 2010, 16:21 (6 Tage 23 Stunden fr¼h)

Factorial.java

```
was:<0.0>
ava:43)
t was:<0.0>
ava:43)
```

Online: 3 Mittelungen: 0

Szenario VI: Simulation

- ◆ Sinnvoll für das Verständnis *dynamischer Verfahren*
- ◆ In der Informatik: Algorithmen und Datenstrukturen
- ◆ Dynamisches Verhalten nachvollziehbar machen
 - ◆ Möglichst *feingranulare* Kontrolle (“Schritt vor/ zurück”)
- ◆ Nutzt in der Regel Java-Programme auf dem Client
 - ◆ Direkt aus Moodle heraus startbar

Beispiele: Simulation

The screenshot displays the JELIOT simulation environment. On the left, a code editor shows the following Java code:

```
1 import jeliot.io.*;
2
3 class Example {
4     static void main() {
5
6         int input = Input.readInt();
7         double real = 20;
8         int i = 0;
9         while (real > input) {
10            real = real * 0.4;
11            i = i + 1;
12            Output.println(i);
13        }
14    }
15 }
16 }
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
```

The main simulation window, titled "Animal Animation: LZW Compression", features a progress bar (0-1000) and a "Zoom" slider (0-500). It displays the LZW algorithm's dictionary and a sequence of characters:

LZW

↓ i

L Z W L Z 7 8 L Z 7 7 L Z C L Z M W L Z A P

LZ	256
ZW	257
WL	258
LZ7	259
78	260
8L	261
LZ77	262
7L	263

The simulation also shows the following code snippet:

```
public void lzw(String[] array) {
    String w, k, result;
    int cnt = 256;
    Hashtable dict = internSetupMethod();
    for (int i=0; i < text.length; i++) {
        k = text[i];
        if (dict.containsKey(w + k)) w = w + k;
        else {
            result += dict.get(w);
            dict.put(w + k, cnt);
            cnt++;
            w = k;
        }
    }
}
```

Current state: w: LZ, k: Z

Ausgabe: 76 90 87 256 55 56 259 55

The bottom control panel includes "Edit", "Compile", "Step", and "Play" buttons, along with "Animation Speed" and "Manuelle Schrittkontrolle" (Manual Step Control) sliders. The "Manuelle Schrittkontrolle" slider is set to 60/100.

Szenario VII: Lernspiele

- ◆ Lernen mit Spaß: zur Auflockerung und Motivation
- ◆ Erweiterung *Game* unterstützt derzeit 8 Lernspiele
 - ◆ Kreuzworträtsel, Galgenmännchen, *Wer wird Millionär*, ...
- ◆ Basis sind in der Regel Wissenstests oder Glossareinträge
- ◆ Intensivere Nutzung als bei „normalen Quizzes“
 - ◆ Motivation „ich will die Millionenfrage knacken“

Lernspiele

The screenshot shows a Moodle game interface. At the top, the browser address bar displays the URL: <http://proffs.tk.informatik.tu-darmstadt.de/teaching/mod/game/attempt.php>. The page title is "Gdl-WS2009: Ha". The navigation bar includes "Lemportal RBG/Dr. Rößling", "Gdl-WS2009", "Games", and "Ha". A "Game bearbeiten" button is visible on the right.

The game interface features a timer showing ~~50:50~~, a crossed-out flag icon, and three person icons. The question text is: "Von wann (und von wem) stammt der erste funktionstüchtige programmgesteuerte Rechenautomat?".

On the right side, there is a score table:

15	1000000
14	500000
13	125000
12	64000
11	32000
10	16000
9	8000
8	4000
7	2000
6	1000
5	500
4	300
3	200
2	100
1	50

Below the question, there are two answer options:

- B John Vincent Atanasoff und Clifford Berry (Atanasoff-Berry-Computer, 1939)
- E Konrad Zuse (Z3, 1941)

At the bottom, there is a link for "Moodle-Dokumentation für diese Seite" and a "Gdl-WS2009" button.

Szenario VIII: Interaktives Lehr- / Lernbuch

- ◆ Inhalte werden segmentiert in Kapitel und Absätze
- ◆ Ein Inhaltsverzeichnis wird auch bereitgestellt
- ◆ Jeder kann absatzweise Fragen oder Notizen einfügen
- ◆ Ebenso *Textmarker*, *Markierung „unverständlich“*, *Lesezeichen*
- ◆ Ausdruck seiten- oder buchweise, mit / ohne Kommentare
- ◆ Integration von Simulationen möglich


Beispiel: Lehr- / Lernbuch

Sortieren durch Einfügen (Insertion Sort)

Das Sortierverfahren "Sortierung durch Einfügen", oft auch als Insertion Sort bezeichnet, ist ein sehr einfaches und naheliegendes Sortierverfahren. Bei dem Verfahren wird, beginnend mit der zweiten Position der Datenstruktur, in jedem Schritt das aktuelle Element in das bereits vorsortierte Feld einsortiert. Das Vorgehen ähnelt dem Sortieren von Spielkarten beim Aufnehmen.

Die folgende Abbildung illustriert das Verfahren und gibt gleichzeitig den Code einer Java-Implementierung an. In der Abbildung sind die grauen Felder noch zu sortieren, während die gelb gefüllten Elemente 1, 2, 3, 6, 7 bereits sortiert vorliegen. Der Zeiger i markiert die Stelle, an der der aktuelle Wert v (hier $v=5$) stand. j hingegen läuft durch das vorherige sortierte Teilfeld und sucht nach der korrekten Einfügeposition für v , hier die Position 3 des Feldes (zwischen den Werten 3 und 6). Die aktuell hervorgehobene Codezeile ist $v = \text{array}[i]$, d.h. der aktuelle Wert von v wurde soeben gesetzt und die Suche nach der Einfügeposition beginnt im nächsten Schritt. Entsprechend stammt die aktuelle Position des Zeigers j noch der vorherigen Iteration (beim Einfügen von $v = 3$).

Insertion Sort

Zuweisungen 
Vergleiche 
 $v = 5$

```
public void insertionSort(int[] array)
{
    int i, j, v;
    for (i=1; i<array.length; i++)
    {
        v = array[i];
        for (j=i; j>0 && v<array[j-1]; j--)
            array[j] = array[j-1];
    }
}
```

Comments:

Bug?
In steps 12, 18, 28, 36, and 46, $j--$ is executed and I expected the pointer j to move one to the left but it didn't. The next step is the insertion which is correctly done but at that time j is pointing one to the right of the insertion point. Also in step 50 $j = i$ is executed but j doesn't move in the animation.
Is that correct?
Bearbeitet von O. William McClung am Fr, 31. Jul 2009, 16:17

JAVE Also Has Bug?
Interestingly JHAVE also appears to have a bug for insertion sort. It uses pseudocode rather than code and is harder for me to follow in part because it uses indenting rather than braces to indicate the body of loops and conditionals. The pseudocode appears to have an error: `vacant--` should be in the body of the if statement. It inserts into index vacant but vacant is one too small at the time of insertion. The animation is correct

Szenario IX: Lernlogik

- ◆ Mehrere miteinander verknüpfte Seiten
- ◆ Jede Seite endet mit Fragen
- ◆ Je nach Antwort wird zu „passender“ Seite gewechselt
 - ◆ „Weiter“ bei korrekter Antwort
 - ◆ „Zurück“ bei komplett falscher Antwort
 - ◆ „Zur Hilfsseite mit mehr Informationen“
 - ◆ ...
- ◆ Gesamte Logik ist vom Dozent festlegbar

Fazit

- ◆ Moodle bietet sehr viele Unterstützungsmöglichkeiten
 - ◆ Aktive Einbeziehung des Studierenden
 - ◆ Beschleunigung des Bewertungsprozesses
 - ◆ Flexibilisierung der Bearbeitung bzgl. Ort und Zeit
- ◆ Wiederverwendung von Elementen in Folgejahren
 - ◆ Kurse können einfach „organisch wachsen“
 - ◆ Jedes Jahr kommen neue Elemente hinzu
- ◆ Probieren Sie es einfach einmal aus!