Interaktives Vorlesungsskript für die Programmier-Grundausbildung

Dr. Markus Brenneis

Wintersemester 23/24

1 Ausgangssituation

Die Veranstaltung *Programmierung* (erstes Fachsemester Informatik) besteht aktuell aus Vorlesung (mit Aufzeichnung) mit wöchentlichen Pflicht-Hausaufgaben, Übungsstunden am Rechner und freiwilligem Zusatzmaterial in Ilias-Lernmodulen (Quizze, interaktive Videos). Die Studierenden haben uns in der Vergangenheit mitgeteilt, dass sie unsere zusätzlichen Materialien und die Aufzeichnungen zur Vor- und Nachbereitung und zur eigenen Wissenskontrolle schätzen. Es wurde aber auch regelmäßig der Wunsch nach einem vollständigen, zusammenhängenden Skript, passgenau zur Vorlesung geäußert.

Den Studierenden stünde mit einem Skript eine bessere Möglichkeit zur Vor- und Nachbereitung der Präsenz-Veranstaltungen, u. a. zur Klausurvorbereitung, zur Verfügung. Zusätzliche Beispiele helfen denjenigen, die nicht sofort alles in der Vorlesung verstanden haben. Weiterführenden Beispiele und Referenzen regen diejenigen, die schon mehr wissen wollen, zum weiteren Erforschen an. Außerdem wird die Barrierefreiheit der Veranstaltung verbessert, da ausformulierte Texte zugänglicher als Screencasts sind, sowohl für Sehbeeinträchtigte, als auch zur einfachen Durchsuchbarkeit.

2 Ziele & Zielgruppen

Das Skript richtet sich primär an die Teilnehmer:innen der Vorlesung Programmierung, was zuletzt über 500 Personen (Tendenz steigend) u. a. aus den Studiengängen Informatik, Computerlinguistik, Naturwissenschaften und Physik waren. Durch die Veröffentlichung als OER soll es aber auch für externen Lernende verfügbar sein.

Eine besondere Herausforderung im ersten Semester sind unterschiedliche Vorkenntnisse der Studierenden, von keinen Programmierkenntnissen bis jahrelange Programmiererfahrung. Das Skript soll hier den verschiedenen Zielgruppen durch zusätzliche Beispiele einerseits und weiterführende Inhalte andererseits zugutekommen.

Wir haben den Anspruch, dass das Skript eine echte Alternative zur Vorlesung wird, sodass Studierende sich ihr Lernmedium frei aussuchen können. Interaktive Quiz-Elemente erlauben Studierenden die eigenständige Kontrolle des Lernfortschritts. Sie sollen zum Nachdenken über den Lernstoff anregen und damit verhindern, dass der Text nur passiv konsumiert wird.

3 Umsetzung

Das Skript wurde als ILIAS-Lernmodul umgesetzt. Ein Textbeispiel und die Inhaltsübersicht sind in Abbildung 1 zu sehen. Für die interaktiven Elemente wurde insbesondere H5P eingesetzt. Ein Beispiel für ein interaktives Quiz-Element zeigt Abbildung 2; das dort verwendete Taschenrechner-Beispiel zieht sich durch die ersten Kapitel und wird immer weiter mithilfe des neu hinzugekommenden Stoffs ausgebaut und verbessert. Ergänzende Informationen, die über die Lernziele der Veranstaltung hinausgehen, sind als ausblendbare Textblöcke umgesetzt (siehe Abbildung 3).

Das Skript wurde im Work-in-Progress-Status den Studierenden etwa ab Mitte der Vorlesungszeit zur Verfügung gestellt. Dadurch wollten wir schon früh Feedback dazu bekommen, ob das Skript dem Bedarf der Studierenden entspricht. Da das Skript in chronologischer Reihenfolge geschrieben wurde, konnte es noch nicht begleitend zur Vorlesung eingesetzt werden, sondern nur als Grundlage zum selbständigen Wiederholen dienen.

Zum Wiederholen und Üben für den zweiten Klausurtermin wurde explizit das Nachlesen im Skript beworben. Etwa 350 Personen haben das Skript mindestens einmal geöffnet.

4 Ergebnisse & Ausblick

Bei der Umsetzung gab es einzelne Probleme durch die Schulung der Hilfskräfte und technische Einschränkungen des Ilias. Das Anlernen der Hilfskräfte, einen didaktisch sinnvoll aufgebauten Text zu schreiben, der für Personen ohne

Vorkenntnisse verständlich ist, war besonders herausfordernd. Dadurch mussten Texte öfter als erwartet überarbeitet und erweitert werden.

Außerdem bietet Ilias leider keine Möglichkeit, Programmcodeschnipsel als Quiztyp zu verwenden, wie es z. B. in Moodle mit dem Coderunner-Plugin möglich wäre. Wir konnten hier mit H5P-Elementen eingeschränkt die Probleme umgehen; die fehlende Syntaxhervorhebung (also die automatische Kennzeichnung z. B. von Programmier-Schlüsselwörtern, wie in Abbildung 1 im Fließtext zu sehen) macht die Darstellung jedoch unattraktiver.

Zum Projektende wurden die mehr als 300 Studierenden, die das Skript angeklickt haben, zu einer Evaluation eingeladen. Dazu gab es leider nur 14 Rückläufer, sodass die Aussagekraft der Ergebnisse eingeschränkt ist. Sehr gut bewertet wurden die Unterstützung beim Lernen, die Einfachheit und die Passgenauigkeit zur Vorlesung. Im Freitext wurden u. a. die flexiblen Lernmöglichkeiten (Vorlesung vs. Nachlesen), die Nachschlagemöglichkeit und die Quizze gelobt.

Mäßig erfolgreich sind wir – gemäß der kleinen Umfragedaten – zurzeit noch bei der Steigerung der Lernmotivation, der Abwechslung und der Steigerung des subjektiven Lernerfolgs. Im Freitext-Feedback wurde u. a. vorgeschlagen, einen Glossar einzuführen und die Schwierigkeit der Aufgaben über eine größere Spannbreite zu variieren.

Das Skript soll in den nächsten Semestern weiter vervollständigt, verbessert und in der Lehrveranstaltung eingesetzt werden. Die Verbesserungsvorschläge aus dem studentischen Feedback werden dabei berücksichtigt. Außerdem planen wir, Querverweise zu dem (noch in Entwicklung befindlichem) Online-Skript der zweiten Erstsemester-Pflichtveranstaltung *Rechnerarchitektur* einzubauen, um fächerübergreifenden Wissenstransfer anzuregen.

5 Ergänzendes Anschauungsmaterial

- vollständigen Evaluationsergebnisse in der Datei `Evaluation.pdf`
- vollständige Kopie des interaktiven Skripts in der Datei `Skript.pdf`; online unter `https://link.cs.hhu.de/prograskript`

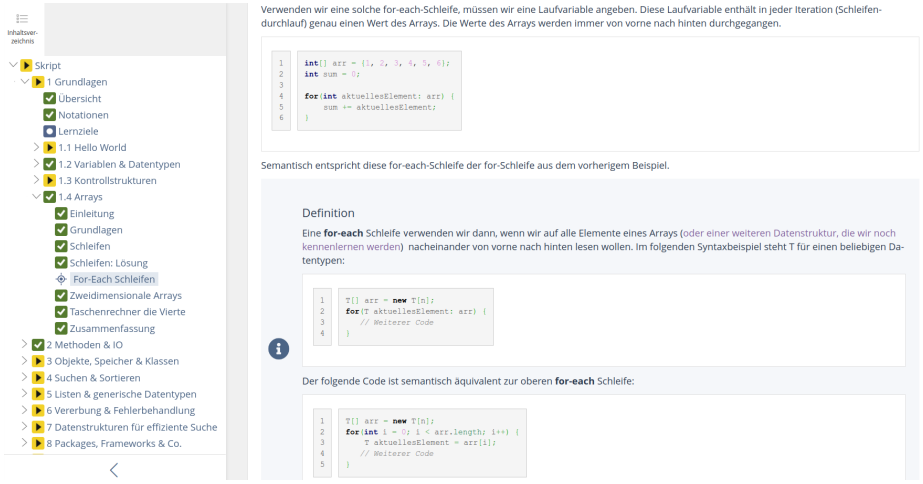

Abbildung 1: Beispieltext zum Thema Kontrollstrukturen und Arrays. Links ist außerdem die Kapitelübersicht zu sehen. Da chronologisch gearbeitet wurde, ist insbesondere die zweite Hälfte des Skripts noch lückenhaft.

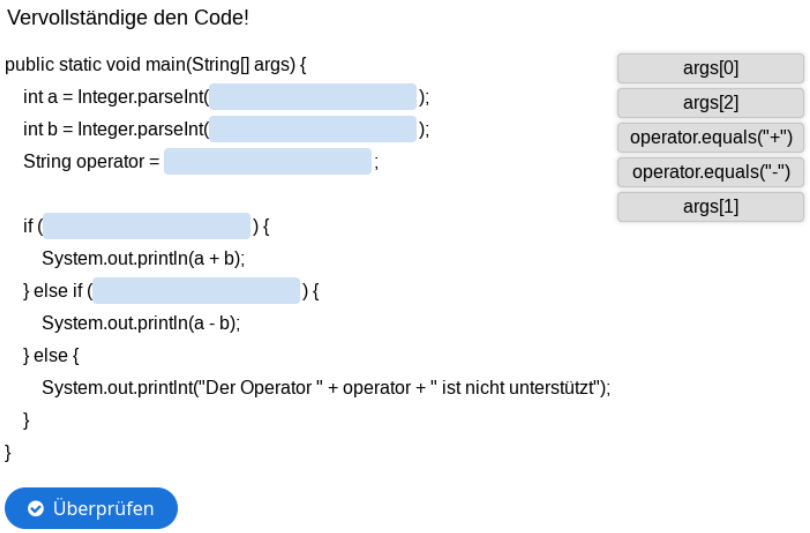

Abbildung 2: H5P-Drag'n'Drop-Aufgabe zum Thema Kontrollstrukturen. Durch direktes Feedback können die Studierenden überprüfen, ob sie die Inhalte bis zu dieser Stelle richtig verstanden haben. Außerdem wird ein passives Konsumieren von Text durch aktive Interaktion unterbrochen.

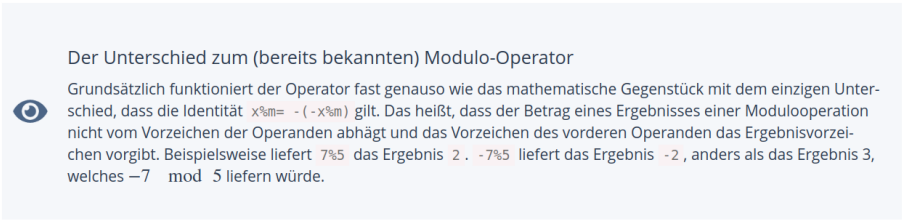

Abbildung 3: Das Skript beinhaltet auch nicht klausurrelevante Inhalte, der für Studierende, die Dinge schon etwas genauer wissen wollen oder die schon Vorwissen haben, gedacht ist. Diese Inhalte befinden sich in Vertiefungswissen-Blöcke, die über „Vertiefungswissen verbergen“ ausgeblendet werden können.