

FORTGESCHRITTENE FUNKTIONALE PROGRAMMIERUNG

TEIL A: ABSCHLUSSPROJEKT

Steffen Jost

LFE Theoretische Informatik, Institut für Informatik,
Ludwig-Maximilians Universität, München

08. Januar 2019



1 ABSCHLUSSPROJEKT

- Ausführung (Wdh)
- Anmeldung
- Endabgabe
- Veröffentlichung



ABSCHLUSSPRÜFUNG

Benotung der Veranstaltung durch Programmierprojekt:

- Durchführbar alleine oder in Gruppe mit bis zu 3 Teilnehmern
- Präsentation & Befragung zu Projekt (ca. 30min)
- Abgabe & Präsentation: Ende März/Anfang April (**KW13**)

ZIEL

Projekt soll im Ersatz einer Klausur belegen, dass der Teilnehmer die Inhalte der Vorlesung verstanden hat und anwenden kann.

- Anhand des Codes sollte tieferes Verständnis für mindestens 3–4 Themen der Vorlesung demonstrieren werden können.
- Passende unbehandelte Themen können nach Absprache auch gelten, z.B. andere Kapitel aus den Büchern, andere Frameworks, etc.
- Code anderer Sprachen wird in der Regel nicht berücksichtigt (z.B. JavaScript in Web-Applikationen wird nicht bewertet)

ABSCHLUSSPRÄSENTATION

Alle zur Teilnahme eingeladen!

Ausschluß Öffentlichkeit möglich

Maximal 10–20 Minuten Präsentation, je nach Teilnehmern, inkl.

- Demonstration der Software
- Diskussion des Codes:
 - Wo wurden welche Techniken aus der Vorlesung eingesetzt?
 - Welche interessanten Probleme traten auf?
 - Welche Bibliotheken/Vorlagen wurden benutzt?
- bei 2–3 Teilnehmern: Wer hat was gemacht?

Anschließend Befragung aller Teilnehmer zu Projekt und Code.

Bewertet werden dabei auch:

- Eingesetzte Techniken
- Eleganz und Klarheit des Codes
- Korrektheit
- Umfang



ABSCHLUSSPROJEKTE DER VORJAHRE

www.tcs.ifi.lmu.de/lehre/ws-2015-16/fun/abschlussprojekt

www.tcs.ifi.lmu.de/lehre/ws-2014-15/fun/fun#projekttable

WEITERE MÖGLICHKEITEN

- Kommandozeilen Tools wie im Buch “Real-World Haskell”
- Interpreter/Parser für eine andere Sprache / DSL
Techniken aus Übungen zum Lambda-Kalkül verwendbar
- Parallele Rechenintensive Hilfstools
per Kommandozeile, Web-Interface oder GUI
- Spiel mit GUI, eventuell auch mit (paralleler) AI
- Yesod-Webapp: Spiel, Planer/Verwaltung, ...



HÄUFIGE MISSVERSTÄNDNISSE

- Themen der Vorlesung sind wichtiger als Thema des Projekts!
Z.B. ist eine umfassende, schöne, funktionierende Webapplikationen trotzdem problematisch, wenn diese nur einen minimalen Umfang von Yesod einsetzt!
Tipp: andere Vorlesungsthemen wie z.B. Fehlerbehandlung nicht vernachlässigen!
- Thema „Monaden“ bedeutet mehr, als einmal irgendwo `runDB` und `do` hinzuschreiben!
D.h. entweder essentielle eigene Monaden-Definition, oder essentieller Einsatz von Monaden-Transformern, oder essentieller Einsatz einer Zustandsmonade, usw.
- Thema „Paralleles Rechnen“: nicht vergessen zu demonstrieren, dass tatsächlich eine Beschleunigung erfolgt und wie gut diese ist! (Threadscope, etc.)
- Es sollte kompilieren!
- Lesbarkeit des Codes ist wirklich wichtig!



ANMELDUNG PROJEKTABNAHME PER UNIWORX

Anmeldung per Uni2Work oder UniWorX *bis 4.2.19* durch Abgabe zu Übungsblatt „Abschlussprojekt“.

Eine Textdatei mit folgenden Information abgeben:

- Namen von bis zu drei Teilnehmern

Bitte nur eine Abgabe; Abgabe-Gruppen bilden!

- Einschränkung/Wünsche beim Prüfungstermin

Welche Uhrzeiten/Tage gehen keinesfalls?

Ausschluß der Öffentlichkeit gewünscht?

- Projekttitle und kurze Beschreibung des Projekts

Welche Bezüge zu den Vorlesungsthemen?

Primärer Prüfungszeitraum 25.3.–29.03.2019

Abgabe/Entwerten bis 18.3.19



- Abgabe bis 18.3.19 per Uni2work/UniWorX über Übungsblatt „Endabgabe Abschlußprojekt“
- Erfolgt keine fristgerechte Abgabe, gilt dies als „Entwertet“
- Termine für die Prüfungen werden auf der FFP Homepage öffentlich bekanntgegeben, durch Angabe von Projekttitle und Initialen der Teilnehmer.
- Wer abgibt, aber unentschuldigt nicht zur Prüfung erscheint gilt als „Nicht erschienen“
⇒ Meldet Euch bei Problemen rechtzeitig per eMail!



VERÖFFENTLICHUNG PROJEKT

Wir würden begrüßen, wenn die Abschlußprojekte nach den Prüfungen auf der Vorlesungshomepage veröffentlicht werden könnten.

Dazu sollte in der Prüfungswoche nach Eurer Prüfung eine ZIP-Datei in Uni2work/UniWorX hochgeladen werden, welche unverändert auf der Homepage "Fortgeschrittene Funktionale Programmierung" veröffentlicht werden darf.

Es sollte enthalten sein:

- Schöner Screenshot für die FFP Webgalerie und das IFI-Infosystem
- copyright.txt in der Ihr angebt, was Ihr mit den enthaltenen Dateien erlaubt ist. **Darin muss ausdrücklich der Veröffentlichung über die Vorlesungshomepage zugestimmt werden!**
- Beschreibung des Projektes in einem Absatz (ggf. Anmeldetext)
- *OPTIONAL*: Quellcode oder Link darauf (z.B. github)
- readme.txt oder ähnlich, mit Hilfe zu Kompilation oder Installation

Die Abgabe ist freiwillig! Wer seinen Code nicht veröffentlichen möchte, darf auch einfach nur Screenshot und Beschreibungstext im Archiv abgeben. Die copyright.txt zur Erlaubnis der Veröffentlichung ist in jedem Fall notwendig!