

Peer-Feedback, Poster-Sessions und OER in ILIAS-Kursräumen

Carsten Gips, Finn Amini Kaveh (HSBI)

21.11.2023



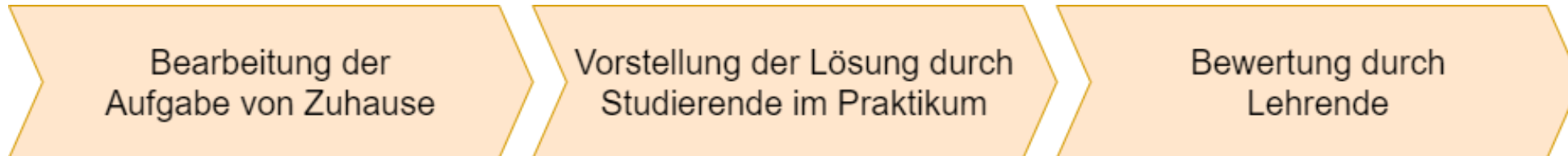
Unless otherwise noted, this work is licensed under CC BY-SA 4.0.

INHALT

1. **Peer-Feedback für die Selbstlernphase**
2. Einrichtung Kursräume
3. OER-Kurse

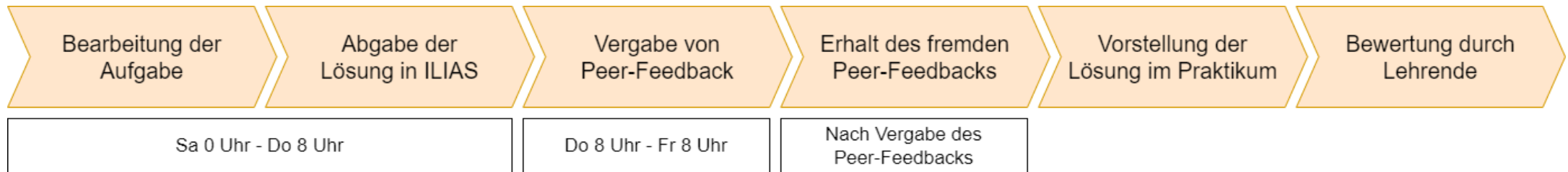
LERNSZENARIO: WIR ENTWICKELN EIN SPIEL

- ┃ Modul Programmiermethoden:
 - ┃ Lehrveranstaltung zum Thema Programmieren mit Java
- ┃ Gamification: Studierende entwickeln in kleinen Teams über das Semester hinweg ein Spiel
- ┃ Wöchentlicher Ablauf des Praktikums:



PEER-FEEDBACK

- ┃ Ziele: Studierende sollen...
 - ┃ Fremde Lösungen (Code) lesen lernen
 - ┃ Fremde Konzepte bewerten lernen
 - ┃ Anregungen für ihre eigenen Lösungen bekommen (Spieleentwicklung!)
- ┃ Organisatorisch: Abgabe zwei Tage früher, Peer-Feedback im ILIAS, danach Praktikum
- ┃ Technisch: ILIAS, Peer-Feedback für Übungen aktiviert



KRITERIENKATALOG - KONZEPT

- ! Welche Aufgaben sollen bearbeitet werden (kurze Benennung)? (Text)
- ! Was soll am Ende entstehen (welche Items, Charakterklassen, ...)? (Text)
- ! Wie gut können Sie die Modellierung nachvollziehen? (5-Sterne-Bewertung)
- ! Was gefällt Ihnen an der Modellierung besonders? (Text)
- ! Geben Sie Hinweise, wie die Modellierung verbessert werden könnte. (Text)
- ! Zusätzliche Bemerkungen – Optional! (Text)
- ! Kein Review – es handelt sich um die Abgabe meines Teams. (Erfüllt Ja/Nein)

KRITERIENKATALOG - IMPLEMENTIERUNG

- Beurteilen Sie die Formatierung des Codes und geben Sie Verbesserungshinweise. (Text)
- Beurteilen Sie die Dokumentation des Codes mit Javadoc und geben Sie Verbesserungshinweise. (Text)
- Was gefällt Ihnen an der Umsetzung besonders? (Text)
- Geben Sie Hinweise, wie die Umsetzung verbessert werden könnte. (Text)
- Zusätzliche Bemerkungen – Optional! (Text)
- Kein Review – es handelt sich um die Abgabe meines Teams. (Erfüllt Ja/Nein)

LESSONS LEARNED

✓ Was lief gut?

- Studierende haben vielfältige Anregungen bekommen
- Studierende können sich besser im Klassenkontext einordnen
- Studierende haben gelernt, konstruktives Feedback zu formulieren
 - Textfelder statt Checkboxes

✗ Was hätte besser laufen können?

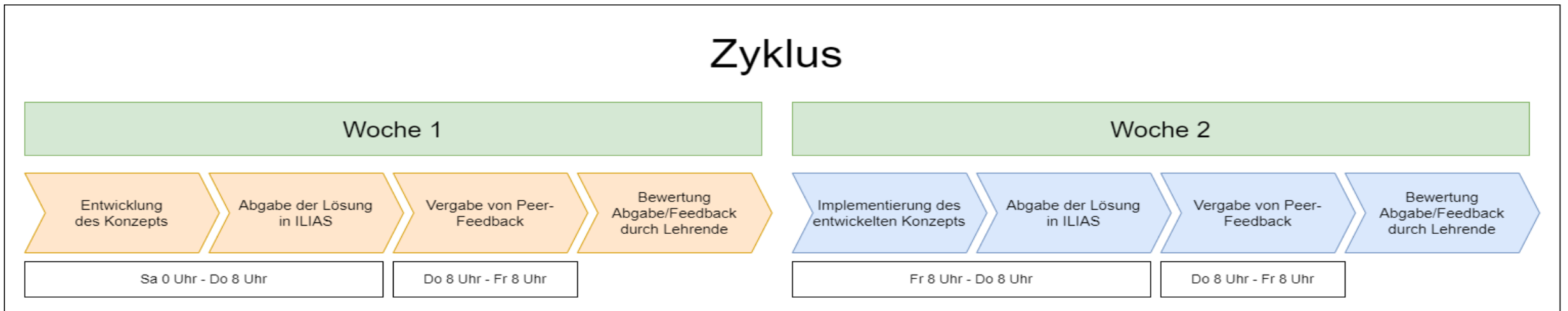
- Einrichtung der Übungsobjekte inklusive des Feedbacks war sehr zeitaufwändig
 - Kriterienkataloge lassen sich nicht kopieren
- Peer-Feedback lässt sich nur für Einzelabgaben konfigurieren
- Die im wöchentlichen Rhythmus verwendete Zeit für das Feedback fehlt an anderer Stelle

INHALT

1. Peer-Feedback für die Selbstlernphase
2. **Einrichtung Kursräume**
3. OER-Kurse

SZENARIO: KONZEPT- UND UMSETZUNGSPHASE

- ┃ Zyklus: Abwechselnd je eine Woche
 - ┃ Erstellung eines Konzepts, und anschließend
 - ┃ Umsetzung des Konzepts
- ┃ Insgesamt 5 Zyklen im Semester
- ┃ Unterschiedliche Feedback-Kataloge
- ┃ Studierende sollten geführt werden, damit sie sich (a) im aktuellen Zyklus und (b) der aktuellen Phase im Zyklus wiederfinden



EINSTELLUNG DER KRITERIENKATALOGE

- Versteckt in den Einstellungen des Übungsobjekts
- Können mehreren Übungseinheiten zugewiesen werden
- Können nicht mehr bearbeitet werden, sobald Feedback-Prozess gestartet ist
- Lassen sich nicht in andere Übungsobjekte kopieren

Erfüllt Ja/Nein

Kriterium hinzufügen

Kriterien

(1 - 7 von 7)

↗ Löschen

Reihenfolge speichern

Position ↑	Titel	Typ	Aktionen
<input type="checkbox"/> 10	Welche Aufgaben sollen bearbeitet werden (kurze Benennung)?	Text	Bearbeiten
<input type="checkbox"/> 20	Was soll am Ende entstehen (welche Items, Charakterklassen, ...)?	Text	Bearbeiten
<input type="checkbox"/> 30	Wie gut können Sie die Modellierung nachvollziehen?	5-Sterne-Bewertung	Bearbeiten
<input type="checkbox"/> 40	Was gefällt Ihnen an der Modellierung besonders?	Text	Bearbeiten
<input type="checkbox"/> 50	Geben Sie Hinweise, wie die Modellierung verbessert werden könnte.	Text	Bearbeiten
<input type="checkbox"/> 60	Zusätzliche Bemerkungen - Optional!	Text	Bearbeiten
<input type="checkbox"/> 70	Kein Review - es handelt sich um die Abgabe meines Teams.	Erfüllt Ja/Nein	Bearbeiten

☐ Alle auswählen

↗ Löschen

Reihenfolge speichern

EINRICHTUNG DES ILIAS-KURSRAUMS

- Seitengestaltung:
 - Für jeden Zyklus wurde ein Block mit den entsprechenden Inhalten angelegt und auf die Übungsobjekte verlinkt
 - Blöcke zeitgesteuert aktiviert und deaktiviert
 - Blöcke zusätzlich farblich unterschiedlich
- Studierende sehen dadurch nur die Kerninformationen sowie den Block für den aktuellen Zyklus
- Vorteile:
 - Intuitives Verständnis der zu erledigen Aufgaben
 - Klare Gliederung

SO SIEHT ES FÜR DEN ADMINISTATOR AUS

Lernkatalog > Campus Minden > Bereich Informatik > Sommersemester 2023 > IFM-2.1 - Programmiermethoden - Gips - SS2023

Tools

Editor

Bearbeitung beenden

Bearbeiten Kopieren, Löschen, ...

Diese Seite wurde für Ihre Bearbeitung gesperrt bis Heute, 18:00. Andere Benutzer können in dieser Zeit keine Änderungen an der Seite vornehmen. Bitte geben Sie die Seite nach der Bearbeitung wieder frei.

☐ **Klicken Sie auf ein Plus-Symbol, um ein neues Element hinzuzufügen.**
☐ **Klicken Sie ein Seitenelement, um es zu bearbeiten.**
☐ **Mit Shift + Klick wählen Sie ein Seitenelement aus und wechseln in den Auswahlmodus.**
☐ **Ziehen Sie ein Seitenelement mit der Maus auf einen Platzhalter um es zu bewegen.**

+

Woche 03: [Zyklus 1: Konzept](#)

Text (Standard)

- Konzept: 20.04. - 08:00 Uhr
- Peer-Feedback: 21.04. - 08:00 Uhr
- Praktikum: 21.04.

+

Woche 04: [Zyklus 1: Implementierung](#)

- Implementierung: 27.04. - 08:00 Uhr
- Peer-Feedback: 28.04. - 08:00 Uhr
- Praktikum: 28.04.

+

Praktikum: Aktueller Zyklus: **Zyklus 2**

+

Woche 05: [Zyklus 2: Konzept](#)

- Konzept: 04.05. - 08:00 Uhr
- Peer-Feedback: 05.05. - 08:00 Uhr
- Praktikum: 05.05.

+

Woche 06: [Zyklus 2: Implementierung](#)

- Implementierung: 11.05. - 08:00 Uhr
- Peer-Feedback: 12.05. - 08:00 Uhr
- Praktikum: 12.05.

+

Praktikum: Aktueller Zyklus: **Zyklus 3**

Aktiv ab: 29. Apr 2023, 00:00 Aktiv bis: 13. Mai 2023, 00:00

Aktiv ab: 13. Mai 2023, 00:00 Aktiv bis: 27. Mai 2023, 00:00

Peer-Feedback, Poster-Sessions und OER in ILIAS-Kursräumen | Hochschule Bielefeld | 13.11.2023 | Seite 12

UND SO FÜR DEN/DIE BENUTZER:IN

Dashboard
 Meine Kurse und Gruppen
 Lernkatalog
 Persönlicher Arbeitsraum
 Kommunikation
 Lerngruppen
 ILIAS Hilfe Wiki

Aufteilung der Abgabeslots im Praktikum:

- [Stundenplangruppe 1: Fr. 09:00 - 10:30 Uhr](#)
- [Stundenplangruppe 2: Fr. 13:30 - 15:00 Uhr](#)
- [Stundenplangruppe 3: Fr. 15:00 - 16:30 Uhr](#)
- [Stundenplangruppe 4: Fr. 10:45 - 12:15 Uhr](#)

Bitte tragen Sie sich und Ihr Team selbstständig ein (nach der ersten Vorlesung).

Fragetypen-Demo: Die [Fragetypen-Demo](#) ist online. Hier können Sie nach Belieben den Umgang mit den am häufigsten (in PM) genutzten Fragetypen üben.

Bei Fragen oder Anmerkungen sprechen Sie mich gern an oder schicken Sie mir eine kurze E-Mail ([Gips, Carsten \[cgips\]](#)). Viel Spass in PM :-)

Praktikum: Aktueller Zyklus: **Zyklus 5**

Woche 11: [Zyklus 5: Konzept](#)

- Konzept: 15.06. - 08:00 Uhr
- Peer-Feedback: 16.06. - 08:00 Uhr
- Praktikum: 16.06.

Woche 12: [Zyklus 5: Implementierung](#)

- Konzept: 22.06. - 08:00 Uhr
- Peer-Feedback: 23.06. - 08:00 Uhr
- Praktikum: 23.06.

Abgelaufene Zyklen:

- **Woche 03:** [Zyklus 1: Konzept](#)
- **Woche 04:** [Zyklus 1: Implementierung](#)
- **Woche 05:** [Zyklus 2: Konzept](#)
- **Woche 06:** [Zyklus 2: Implementierung](#)
- **Woche 07:** [Zyklus 3: Konzept](#)
- **Woche 08:** [Zyklus 3: Implementierung](#)
- **Woche 09:** [Zyklus 4: Konzept](#)
- **Woche 10:** [Zyklus 4: Implementierung](#)

POSTER-GALERIE – MODUL: KI

Dashboard

Meine Kurse und Gruppen

Lernkatalog

Persönlicher Arbeitsraum

Kommunikation

Lerngruppen

ILIAS Hilfe Wiki

Modellierung

Die Fitnessfunktion ist der Schlüssel zur Lösung dieses Problems. Jedes Teilproblem kann als Teilproblem betrachtet werden.

Jedes Teilproblem enthält Vorwissen, Raum, Namen und Semester.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Experimente

Wenn jede Lösung auf 4 Stunden abgelegt wird, ist der Algorithmus langwieriger, um ein Fitness zu finden.

Wenn die Anzahl der Räume reduziert wird, wird es schwieriger, die Lösung zu finden.

Räume, Namen und Semester sind vorgegeben.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Ergebnisse

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Fazit

Die Behandlung des Problems als CSP verläuft bei der Implementierung auf komplexer Weise.

Der genetische Algorithmus ist für dieses Problem geeignet.

Da die Population so groß ist, ändert sich das Ergebnis nicht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Stundenplanung mit «Genetic Algorithm»

Anil Cemal Yasar – 200503022 Türkisch-Deutsche Universität

Zusammenfassung

Die Stundenplanung ist ein Problem, das die Stundenplanung mit der Fitnessfunktion ist der Schlüssel zur Lösung dieses Problems. Jedes Teilproblem kann als Teilproblem betrachtet werden.

Jedes Teilproblem enthält Vorwissen, Raum, Namen und Semester.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Einführung

Die Stundenplanung ist ein Problem, das die Stundenplanung mit der Fitnessfunktion ist der Schlüssel zur Lösung dieses Problems. Jedes Teilproblem kann als Teilproblem betrachtet werden.

Jedes Teilproblem enthält Vorwissen, Raum, Namen und Semester.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Ergebnis

Die Stundenplanung ist ein Problem, das die Stundenplanung mit der Fitnessfunktion ist der Schlüssel zur Lösung dieses Problems. Jedes Teilproblem kann als Teilproblem betrachtet werden.

Jedes Teilproblem enthält Vorwissen, Raum, Namen und Semester.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Stundenplanproblemlösung mit genetischem Algorithmus

Musab Nail Cakerek

Einführung für Stundenplanproblem

Es gibt jedes Semester neue Stundenpläne. Wir müssen ein so abstraktes Problem lösen, dass es ein System ist, bei dem wir es nicht jedes Semester neu anfangen müssen.

Problemlösung: 4 Räume, 4 Dozenten, 11 Vorlesungen, 3 Semester

Lösungsweg – Genetischer Algorithmus

Genetische Algorithmen werden häufig verwendet, um komplexe Probleme zu lösen, die nicht durch traditionelle Algorithmen gelöst werden können. In diesem Fall wird der genetische Algorithmus verwendet, um einen Stundenplan zu erstellen, der alle Anforderungen erfüllt.

Beispiel Stundenplan

Zeit	Montag	Dienstag	Mittwoch	Donnerstag	Freitag
08:00					
10:00					
12:00					
14:00					
16:00					
18:00					

Modellierung und Implementierung

Operationen: Kreuzung, Mutation, Selektion, etc.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Fazit

Die Stundenplanung ist ein Problem, das die Stundenplanung mit der Fitnessfunktion ist der Schlüssel zur Lösung dieses Problems. Jedes Teilproblem kann als Teilproblem betrachtet werden.

Jedes Teilproblem enthält Vorwissen, Raum, Namen und Semester.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Stundenplan mit Genetische Algorithmus

Yusuf Ziya AK

Mit der Fitness-Funktion könnte das Stundenplanproblem durch genetische Algorithmen gelöst werden.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

1. Einführung

Die Stundenplanung ist ein Problem, das die Stundenplanung mit der Fitnessfunktion ist der Schlüssel zur Lösung dieses Problems. Jedes Teilproblem kann als Teilproblem betrachtet werden.

Jedes Teilproblem enthält Vorwissen, Raum, Namen und Semester.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Stundenplanung mit CSP

Nihan Bekarlar

Ein Stundenplanungsproblem kann als zwei aufeinanderfolgende Kartenfärbeprobleme angesehen werden

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Eigenschaften des Problems

Die Stundenplanung ist ein Problem, das die Stundenplanung mit der Fitnessfunktion ist der Schlüssel zur Lösung dieses Problems. Jedes Teilproblem kann als Teilproblem betrachtet werden.

Jedes Teilproblem enthält Vorwissen, Raum, Namen und Semester.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Modellierung als Kartenfärbungsproblem

Die Stundenplanung ist ein Problem, das die Stundenplanung mit der Fitnessfunktion ist der Schlüssel zur Lösung dieses Problems. Jedes Teilproblem kann als Teilproblem betrachtet werden.

Jedes Teilproblem enthält Vorwissen, Raum, Namen und Semester.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Ergebnis

Die Stundenplanung ist ein Problem, das die Stundenplanung mit der Fitnessfunktion ist der Schlüssel zur Lösung dieses Problems. Jedes Teilproblem kann als Teilproblem betrachtet werden.

Jedes Teilproblem enthält Vorwissen, Raum, Namen und Semester.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Wieder ist, dass der zufällig erzeugte Raum während der Mutation am besten ist, was dem Stundenplan entspricht.

Peer-Feedback, Poster-Sessions und OER in ILIAS-Kursräumen | Hochschule Bielefeld | 13.11.2023 | Seite 14

DIGITAL LEARNING SCOUTS VON DIGIKOS

- Die Entwicklung und Umsetzung der Konzepte wurde unterstützt durch das DigikoS Projekt. Dazu gehörten...
 - Die Einrichtung des Kursraumes in ILIAS
 - Die Konfiguration des Peer-Feedbacks
 - Die Konfiguration der Objekte für die Zyklen
 - Die Umsetzung der Poster-Galerie
 - Die Hilfe bei der Entwicklung von Umfragen zur Evaluation der eingesetzten Techniken



INHALT

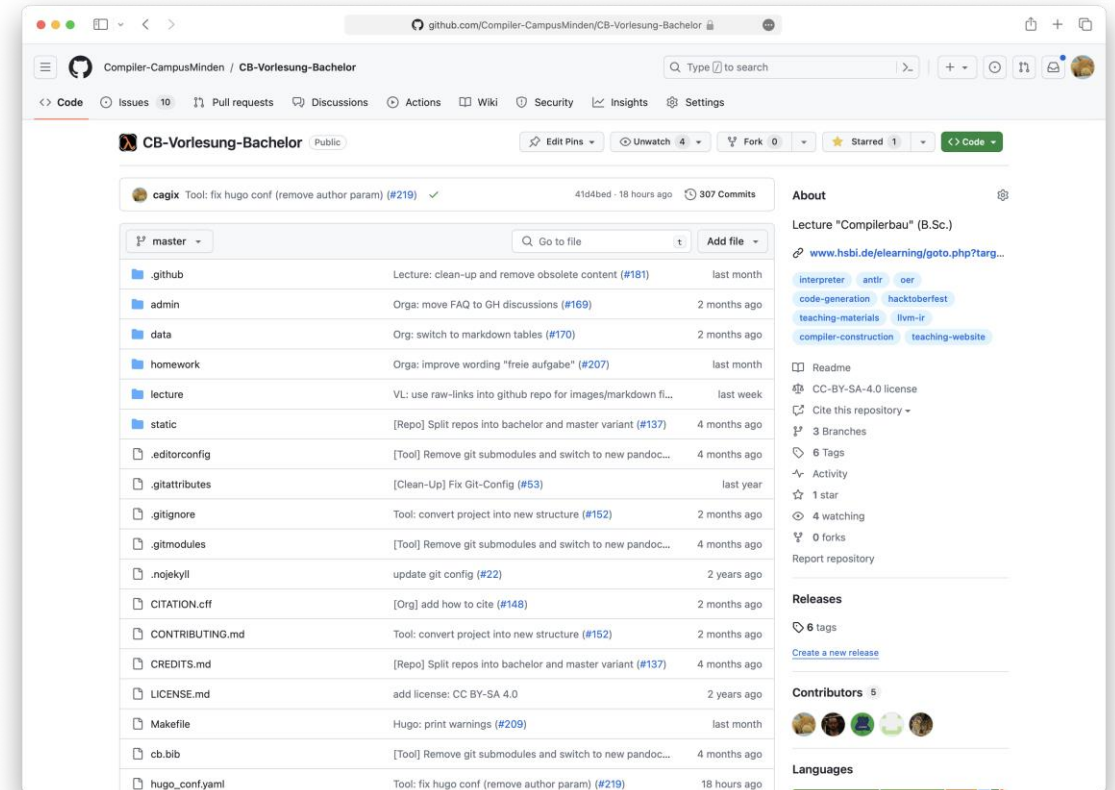
1. Peer-Feedback für die Selbstlernphase
2. Einrichtung Kursräume
3. **OER-Kurse**

KONZEPT

- I Prof. Gips hat aktuell **vier Module als OER** überarbeitet und bereitgestellt
 - I Entwicklung und Versionierung der Inhalte auf GitHub als Markdown-Quellen
 - I **Selbstentwickelte Toolchain** erstellt daraus automatisch **Folien (PDF)** sowie interaktive **Lehrunterlagen (HTML-Lernmodul)**
- I Kurse werden im **offenen Bereich im ILIAS** angelegt: Beschreibung, Lernmodul, Link zum aktuellen geschlossenen Kursraum
- I Geschlossener Kursraum für Durchführung des Moduls: Organisation der Studierenden, Abgaben im Praktikum, ... => Link in den offenen Kursraum für das Lernmodul

SCREENSHOT GITHUB COMPILERBAU

- Öffentlich zugänglich
- Beinhaltet alle bereitgestellten Lehrinhalte
- Lehrinhalte zusätzlich in Form von Videos auf Youtube und im Medienportal
- Selbsttests in ILIAS (versteckt im offenen Kursraum)



OFFENER KURSRAUM COMPILERBAU

- Für jeden zugänglich
- Darstellung aller fachlichen Inhalte im Lernmodul
- Umfasst eine Beschreibung des Moduls, sowie den Ablauf des jeweiligen Semesters
- Versteckt: Selbsttests

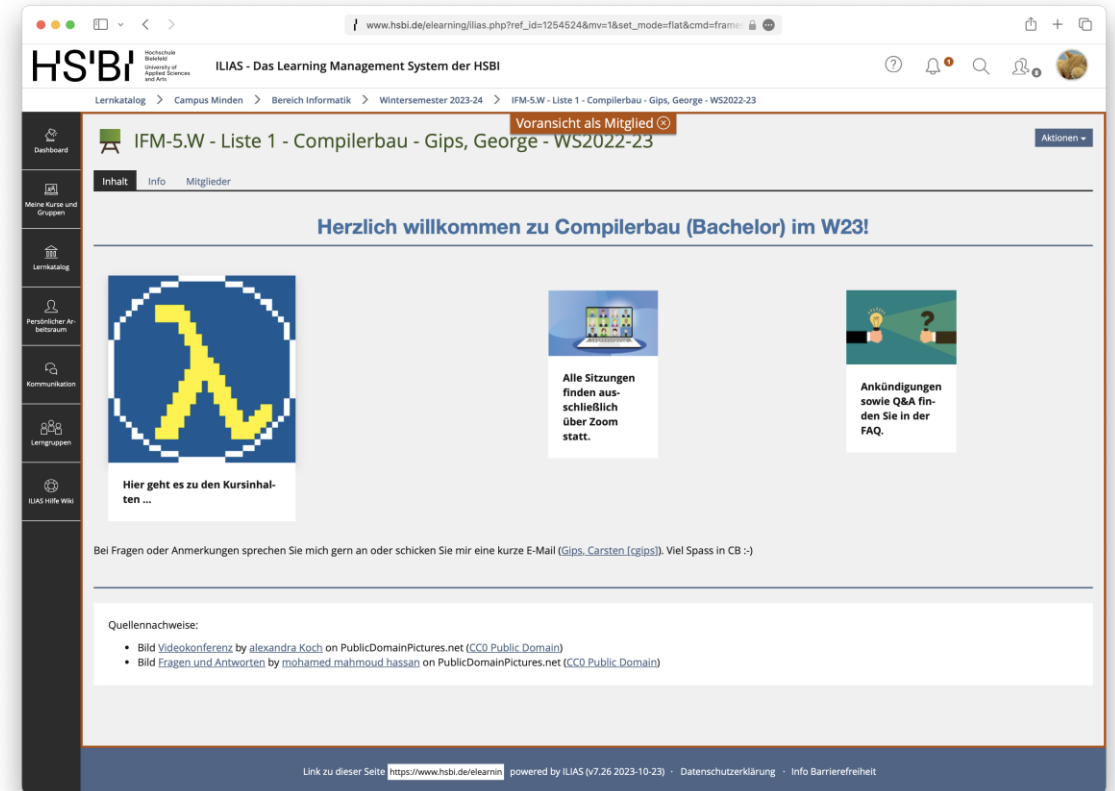
The screenshot shows the ILIAS web interface for the 'Compilerbau' course. The browser address bar shows the URL: `www.hsbi.de/elearning/ilias.php?ref_id=1089779&mv=1&set_mode=flat&cmd=frame`. The page header includes the HS'BI logo and the text 'ILIAS - Das Learning Management System der HSBI'. Below the header, there is a navigation bar with links: 'Lernkatalog > Offene Angebote | OER an der HSBI > Informatik > Compilerbau'. The main content area is titled 'Compilerbau' and includes a sub-header '(We CAN virtuOWL / NAC, Kooperation mit University of Edmonton/Canada)'. A sidebar on the left contains icons for 'Dashboard', 'Meine Kurse und Gruppen', 'Lernkatalog', 'Persönlicher Arbeitsraum', 'Kommunikation', 'Lerngruppen', and 'ILIAS Hilfe-Wiki'. The main content area features a large diagram illustrating the compilation process. The diagram shows a flow from 'Quellcode' (Source Code) to 'Objektcode' (Object Code) and finally to 'Ausgabe' (Output). The 'Quellcode' section contains a code snippet:

```
#include <stdio.h>
int main() {
    printf("Hello World");
}
```

. The 'Objektcode' section shows a diagram of the compilation process, including 'Lexical Analyse', 'Syntax Analyse', 'Semantische Analyse', 'Intermediäre Code-Generierung', 'Code-Optimierung', and 'Code-Übersetzung'. The 'Ausgabe' section shows a diagram of the output, including 'Ausgabe' and 'Ausgabe'. Below the diagram, there is a welcome message: 'Herzlich willkommen zu Compilerbau!'. At the bottom, there is a small logo and the text 'Hier geht es zu den Kursinhalten ...'. A footer note states: 'Unless otherwise noted, this work by BC George, Carsten Gips, and contributors is licensed under CC BY-SA 4.0.'

GESCHLOSSENER KURSRAUM COMPILERBAU

- Ausschließlich zugänglich für Studierende, die das Modul Compilerbau belegen
- Beinhaltet keine fachlichen Inhalte
- Verlinkt auf den offenen Kursraum



Quelle: Videokonferenz by alexandra Koch on PublicDomainPictures.net (CC0 Public Domain)

Quelle: Fragen und Antworten by mohamed mahmoud hassan on PublicDomainPictures.net (CC0 Public Domain)

SO SIEHT ES AUS: LERNMODUL COMPILERBAU

IFM 5.21: COMPILERBAU (WINTER 2023/24)

```
#include <stdio.h>

int main() {
    printf("Hello World");
}
```

Kursbeschreibung

Der Compiler ist das wichtigste Werkzeug in der Informatik. In der Königsdisziplin der Informatik schließt sich der Kreis, hier kommen die unterschiedlichen Algorithmen und Datenstrukturen und Programmiersprachenkonzepte zur Anwendung.

In diesem Modul geht es um ein grundlegendes Verständnis für die wichtigsten Konzepte im Compilerbau. Wir schauen uns dazu relevante aktuelle Tools und Frameworks an und setzen diese bei der Erstellung eines kleinen Compiler-Frontends für *Mini-Python* ein.

Überblick Modulinhalt

1. Lexikalische Analyse: Scanner/Lexer
 - Reguläre Sprachen
 - Generierung mit ANTLR
2. Syntaxanalyse: Parser
 - Kontextfreie Grammatiken (CFG)
 - LL-Parser (Top-Down-Parser)
 - Generierung mit ANTLR
3. Semantische Analyse: Attributierte Grammatiken und Symboltabellen

ANTLR GENERIEREN

Lexik: Erzeugen eines Token-Stroms aus einem Zeichenstrom

Definition wichtiger Begriffe

Typische Muster für Erstellung von Token

Hello World

Hinweis zur Grammatik (Regeln)

ANTLR einrichten

"Hello World" übersetzen und ausführen

Generierte Dateien und Klassen

Bedeutung der Ausgabe

ANTLR-Grammatik für die Lexer-Generierung

Greedy und Non-greedy Lexer-Regeln

Verhalten des Lexers: 1. Längster Match

Verhalten des Lexers: 2. Reihenfolge

Verhalten des Lexers: 3. Non-greedy Regeln

Attribute und Aktionen

Attribute bei Token (Auswahl)

Aktionen mit den Lexer-Regeln

Wrap-Up

werden. Diese Aktionen müssen in der Zielprogrammiersprache formuliert werden, da sie in die generierte Lexerklasse in die jeweiligen Methoden eingebettet werden.

Videos (YouTube)

- VL Lexer mit ANTLR
- Demo ANTLR Basics
- Demo Verhalten Lexer-Regeln
- Demo Lexer-Regeln mit Aktionen

Videos (HSBI-Medienportal)

- VL Lexer mit ANTLR

Lernziele

- (K3) Lexer-Regeln in ANTLR formulieren und einsetzen
- (K2) Verhalten des Lexers: längste Matches, Reihenfolge
- (K3) Nutzung von Lexer-Aktionen

Lexer: Erzeugen eines Token-Stroms aus einem

OER-KURSE (ÜBERBLICK)

- Künstliche Intelligenz: [ILIAS](#), [GitHub](#)
- Programmiermethoden: [ILIAS](#), [GitHub](#)
- Compilerbau: [ILIAS](#), [GitHub](#)
- Concepts of Programming Languages: [ILIAS](#), [GitHub](#)

Vielen Dank für Ihre
Aufmerksamkeit!