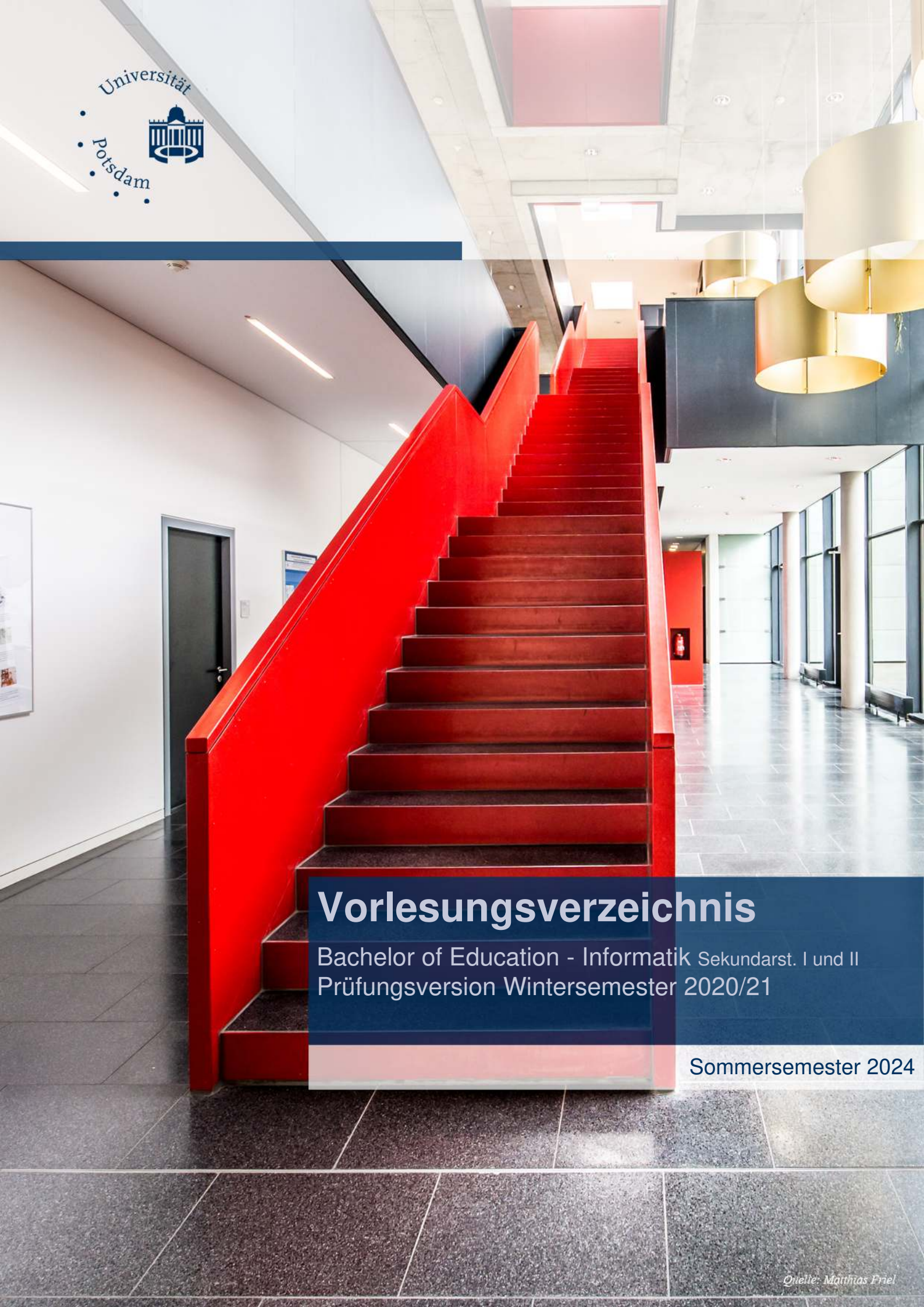


Universität Potsdam

Vorlesungsverzeichnis

Bachelor of Education - Informatik Sekundarst. I und II
Prüfungsversion Wintersemester 2020/21

Sommersemester 2024

Quelle: Matthias Friel

Inhaltsverzeichnis

Abkürzungsverzeichnis 4

Pflichtmodule 5

INF-1010 - Grundlagen der Programmierung 5

107988 U - Making Music with Computers - Creative Programming in Python (Rechnerübung) 5

INF-1011 - Algorithmen und Datenstrukturen 6

107959 U - Algorithmen und Datenstrukturen 6

107960 V - Algorithmen und Datenstrukturen 6

INF-1020 - Formale Grundlagen der Informatik 7

INF-1021 - Theoretische Grundlagen: Effiziente Algorithmen 7

108005 VU - Theoretische Informatik II: Effiziente Algorithmen 7

INF-1030 - Maschinenmodelle 8

INF-1031 - Betriebssysteme und Rechnernetze 8

INF-1050 - Daten- und Wissensbasierte Systeme 8

107993 PR - Principles of Data- and Knowledge-Base Systems 9

107994 VU - Principles of Data- and Knowledge-Base Systems 9

INF-1060 - Software Engineering I 9

107998 VU - Research Software Engineering 9

107999 PJ - Research Software Engineering 10

INF-1060 - Software Engineering I (auslaufend) 11

107998 VU - Research Software Engineering 11

107999 PJ - Research Software Engineering 12

INF-6010 - Praxis der Programmierung 12

107992 VU - Praxis der Programmierung 12

INF-DDI-1 - Didaktik der Informatik I 13

107967 VU - Didaktik der Informatik I 13

108184 S - Schulpraktische Studien 13

MAT-1100 - Mathematik für Informatik I 13

MAT-1101 - Mathematik für Informatik II 14

Wahlpflichtmodule 14

INF-1040 - Konzepte paralleler Programmierung 14

107984 V - Konzepte paralleler Programmierung 14

107985 U - Konzepte paralleler Programmierung 14

INF-1080 - Künstliche Intelligenz 14

INF-2010 - Rechnernetze 15

107969 VU - Distributed Systems 15

INF-2030 - Netzbasierte Datenverarbeitung 15

108065 VU - Netzbasierte Datenverarbeitung 15

INF-2031 - Multimediatechnologie 16

INF-2040 - Software Engineering II 16

108003 VU - Software Engineering II 16

Abkürzungen entnehmen Sie bitte Seite 4

INF-2061 - Information und Komplexität **16**

107963 VU - Codierungstheorie 16

Akademische Grundkompetenzen **18**

Glossar **19**

Abkürzungen entnehmen Sie bitte Seite 4

Abkürzungsverzeichnis

Veranstaltungsarten

AG	Arbeitsgruppe
B	Blockveranstaltung
BL	Blockseminar
DF	diverse Formen
EX	Exkursion
FP	Forschungspraktikum
FS	Forschungsseminar
FU	Fortgeschrittenenübung
GK	Grundkurs
HS	Hauptseminar
KL	Kolloquium
KU	Kurs
LK	Lektürekurs
LP	Lehrforschungsprojekt
OS	Oberseminar
P	Projektseminar
PJ	Projekt
PR	Praktikum
PS	Proseminar
PU	Praktische Übung
RE	Repetitorium
RV	Ringvorlesung
S	Seminar
S1	Seminar/Praktikum
S2	Seminar/Projekt
S3	Schulpraktische Studien
S4	Schulpraktische Übungen
SK	Seminar/Kolloquium
SU	Seminar/Übung
TU	Tutorium
U	Übung
UN	Unterricht
UP	Praktikum/Übung
UT	Übung / Tutorium
V	Vorlesung
V5	Vorlesung/Projekt
VP	Vorlesung/Praktikum
VS	Vorlesung/Seminar
VU	Vorlesung/Übung
W	Werkstatt
WS	Workshop

Veranstaltungsrhytmen

wöch.	wöchentlich
14t.	14-täglich
Einzel	Einzeltermin
Block	Block
BlockSa	Block (inkl. Sa)
BlockSaSo	Block (inkl. Sa,So)

Andere

N.N.	Noch keine Angaben
n.V.	Nach Vereinbarung
LP	Leistungspunkte
SWS	Semesterwochenstunden
	Belegung über PULS
PL	Prüfungsleistung
PNL	Prüfungsnebenleistung
SL	Studienleistung
L	sonstige Leistungserfassung

Vorlesungsverzeichnis

Pflichtmodule

INF-1010 - Grundlagen der Programmierung

107988 U - Making Music with Computers - Creative Programming in Python (Rechnerübung)

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
1	U	Do	14:00 - 16:00	wöch.	2.70.0.05	11.04.2024	Florian Reuß

Kommentar

Registrieren Sie sich unbedingt auf PULS sowohl zu Vorlesung/Übung als auch zur Rechnerübung!

Die Zuordnung zu dieser Rechnerübungsgruppe erfolgt über PULS und nicht über den GdP-Moodle-Kurs!

Voraussetzung

Eigene Kopfhörer erforderlich!

Englischkenntnisse erforderlich (Lehrmaterialien auf Englisch).

Literatur

Manaris, B., & Brown, A. R. (2014). Making music with computers: Creative programming in python. Chapman and Hall/CRC.

Leistungsnachweis

In der Rechnerübung zum Modul *Grundlagen der Programmierung* gibt es eine Prüfungsnebenleistung (PNL) zum Abschluss des Moduls (Verbuchung der Leistungspunkte). Die Zulassung zur Prüfung erfolgt unabhängig von dieser PNL. Die PNL wird durch eine Testatleistung im Computerlabor (45–60 Minuten) erbracht und gilt als bestanden, wenn mindestens 50% der geforderten Testatleistung erzielt wurden.

Bemerkung

Making Music with Computers ist eine Einführung in die kreative Programmierung mit der Programmiersprache Python.
Es nutzt aufregende und innovative Aktivitäten zur Erzeugung von Musik, um letztlich Programmierkompetenzen und Computational Thinking zu vermitteln. In dieser Rechnerübung lernen wir, wie wir die Kreativität und das Design der Künste mit der mathematischen Strenge und Formalität der Informatik verbinden können.

In den ersten vier Wochen des Vorlesungszeitraums wird der Umgang mit dem Betriebssystem UNIX/Linux erlernt.
Besuchen Sie in den ersten vier Vorlesungswochen eine der anderen Rechnerübungen! Diese Veranstaltung beginnt erst am 09.11.!

Lerninhalte

Die Übung deckt einen Großteil der Konzepte ab, die in den herkömmlichen Rechnerübungen zur Veranstaltung *Grundlagen der Programmierung* zu finden sind! Zu diesen Konzepten gehören Datentypen, Variablen, Zuweisungen, arithmetische Operatoren, I/O, Algorithmen, Verzweigungen (if-else), Vergleichsoperatoren, Junktoren, Iteration/Schleifen, Rekursion, Listen (Arrays), Funktionen, Modularisierung (Funktionen) sowie Klassen (objektorientierte Programmierung). Darüber hinaus befassen wir uns mit Grundlagen der Musikgeschichte/-theorie, grafischen Benutzeroberflächen (GUIs), ereignisgesteuerter Programmierung, Big Data und MIDI-Programmierung.

Zielgruppe

Studierende im ersten oder zweiten Studienjahr, die sich für Computermusik interessieren und die Grundlagen des Programmierens in einem kreativen Kontext erlernen wollen.

Erwartet werden überdurchschnittliches Interesse und Lernbereitschaft in den Themengebieten Musik & Programmierung. Vorkenntnisse sind nicht vonnöten.

Abkürzungen entnehmen Sie bitte Seite 4

Leistungen in Bezug auf das Modul

PNL 550113 - Rechnerübung (unbenotet)

INF-1011 - Algorithmen und Datenstrukturen

107959 U - Algorithmen und Datenstrukturen

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
1	U	Do	14:00 - 16:00	wöch.	2.70.0.10	11.04.2024	Dr. Henning Bordihn
2	U	Fr	12:00 - 14:00	wöch.	2.70.0.11	12.04.2024	Dr. Henning Bordihn
	Für Lehramtsstudierende.						
3	U	Di	12:00 - 14:00	wöch.	2.70.0.10	09.04.2024	Dr. Henning Bordihn
	Für Lehramtsstudierende.						
4	U	Fr	14:00 - 16:00	wöch.	2.70.0.11	12.04.2024	Dr. Henning Bordihn
5	U	Di	10:00 - 12:00	wöch.	2.70.0.10	09.04.2024	Dr. Henning Bordihn

Voraussetzung

Kenntnisse aus Grundlagen der Programmierung

Leistungsnachweis

Klausur am Schluß der Lehrveranstaltung

Prüfungsnebenleistung zum Abschluss des Moduls: erfolgreiche Bearbeitung von wöchentlichen Programmieraufgaben

Lerninhalte

- Abstrakte Datentypen und Datenstrukturen, insbesondere Sequenzen, Zeiger, Bäume, Mengen und deren Verwendung in Algorithmen
- Analyse von Algorithmen (Asymptotik)
- Algorithmische Prinzipien: Teile und Herrsche, Dynamisches Programmieren, Greedy-Algorithmen
- Algorithmen auf Sequenzen und Graphen, insbesondere Suchen und Sortieren, Bäume, balancierte Bäume, Hashing
- Komplexität von Problemen, NP-Vollständigkeit

Leistungen in Bezug auf das Modul

PNL 550213 - Übung (unbenotet)

107960 V - Algorithmen und Datenstrukturen

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
1	V	Mo	12:00 - 14:00	Einzel	2.70.0.11	08.04.2024	Dr. Henning Bordihn
1	V	Mo	14:00 - 16:00	wöch.	2.27.1.01	08.04.2024	Dr. Henning Bordihn

Voraussetzung

Kenntnisse aus Grundlagen der Programmierung

Leistungsnachweis

Klausur am Schluß der Lehrveranstaltung

Prüfungsnebenleistung zum Abschluss des Moduls: erfolgreiche Bearbeitung von wöchentlichen Programmieraufgaben

Abkürzungen entnehmen Sie bitte Seite 4

Bemerkung

Die Vorlesung findet grundsätzlich montags von 14-16 Uhr statt.

Der Termin von 12-14 Uhr ist ein einmaliger Sondertermin für Hörer aus dem Studiengang Kognitionswissenschaften. Alle anderen Teilnehmer und Teilnehmerinnen sollen diesen Temin nicht wahrnehmen.

Lerninhalte

- Abstrakte Datentypen und Datenstrukturen und deren Verwendung in Algorithmen, insbesondere Sequenzen, Bäume, Mengen, Datentyp Zeiger
- Analyse von Algorithmen (Asymptotik)
- Algorithmische Entwurfstechniken: Teile und Herrsche, Dynamisches Programmieren, Greedy-Algorithmen
- Algorithmen auf Sequenzen und Graphen, insbesondere Suchen und Sortieren, Suchbäume, balancierte Suchbäume, Hashing, kürzeste Pfade, minimaler Spannbaum
- Komplexität von Problemen, NP-Vollständigkeit

Leistungen in Bezug auf das Modul

PNL 550212 - Vorlesung (unbenotet)

INF-1020 - Formale Grundlagen der Informatik

Für dieses Modul werden aktuell keine Lehrveranstaltungen angeboten

INF-1021 - Theoretische Grundlagen: Effiziente Algorithmen

108005 VU - Theoretische Informatik II: Effiziente Algorithmen

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
Alle	TU	Di	08:00 - 10:00	wöch.	2.70.0.11	09.04.2024	Dr. rer. nat. Sebastian Böhne
Alle	V	N.N.	N.N.	wöch.	N.N.	N.N.	Dr. rer. nat. Sebastian Böhne
	Online asynchron.						
1	U	Do	08:00 - 10:00	wöch.	2.70.0.10	11.04.2024	Dr. rer. nat. Sebastian Böhne
2	U	Fr	08:00 - 10:00	wöch.	2.70.0.11	12.04.2024	Dr. rer. nat. Sebastian Böhne
3	U	Fr	10:00 - 12:00	wöch.	2.70.0.10	12.04.2024	Dr. rer. nat. Sebastian Böhne
	Für Lehramtsstudierende.						

Kommentar

Alle Informationen im Moodle-Kurs "Theoretische Grundlagen: Effiziente Algorithmen (SoSe 2024)" (Kurztitel "TI-II-SoSe2024"). Einschreibeschlüssel bei der ersten Hörsaalübung (09.04.) oder per Anfrage an boehne@uni-potsdam.de

---

Die Theoretische Informatik beschäftigt sich mit den grundlegenden Fragestellungen der Informatik. Hierzu werden Computer- und Automatenmodelle idealisiert und mathematisch untersucht.
Die Automatentheorie und die Theorie der formalen Sprachen (Thema des ersten Semesters) ist grundlegend für die Entwicklung von Programmiersprachen und Compilern. Sie untersucht, mit welchen Techniken welche Arten von Sprachen effizient analysiert werden können.
Die Berechenbarkeitstheorie befasst sich mit den prinzipiellen Grenzen des Berechenbaren und der Relation zwischen verschiedenen Computer- und Programmiermodellen. Die Komplexitätstheorie untersucht Effizienz von Algorithmen im Hinblick auf Platz- und Zeitbedarf und kümmert sich insbesondere um die Frage, wie effizient man bestimmte Probleme lösen kann.
Gliederung der Theoretischen Informatik II:

- Berechenbarkeitstheorie:

- Turingmaschinen
- Loop-, While- und Goto-Programme
- Rekursive Funktionen
- Lambda-Kalkül
- Churchsche These
- Berechenbarkeit, Aufzählbarkeit und Entscheidbarkeit
- Unlösbare Probleme
- Beweistechniken für Unlösbarkeit

- Komplexitätstheorie:

- Konkrete Komplexitätsanalyse
- Komplexitätsklassen
- Handhabbarkeit
- NP-Vollständigkeit
- Satz von Cook
- NP-Vollständigkeit bei konkreten Problemen nachweisen
- Kurzvorstellung weiterer Problemklassen und weiter Methoden

Voraussetzung

Erfolgreiche Teilnahme an Theoretische Informatik I ist sehr zu empfehlen

Literatur

Dirk Hoffmann: Theoretische Informatik

Hopcroft, R. Motwani, J. Ullman: Einfuehrung in die Automatentheorie, Formale Sprachen und Komplexitaetstheorie, Pearson 2002

Michael Sipser: Introduction to the Theory of Computation. 2. Auflage, PWS 2005 J

Leistungsnachweis

Klausur zu Beginn des vorlesungsfreien Zeitraums

Leistungen in Bezug auf das Modul

PNL	550412 - Vorlesung und Übung und Tutorium (unbenotet)

INF-1030 - Maschinenmodelle

Für dieses Modul werden aktuell keine Lehrveranstaltungen angeboten

INF-1031 - Betriebssysteme und Rechnernetze

Für dieses Modul werden aktuell keine Lehrveranstaltungen angeboten

INF-1050 - Daten- und Wissensbasierte Systeme

107993 PR - Principles of Data- and Knowledge-Base Systems

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
1	PR	N.N.	N.N.	wöch.	N.N.	N.N.	Prof. Dr. Torsten Schaub, Francois Laferriere

Kommentar

Moodle course: moodle

Literatur

Principles of Database & Knowledge-Base Systems by Jeffrey D. Ullman W. H. Freeman & Co. New York, NY, USA

Leistungen in Bezug auf das Modul

PNL 550831 - Praktikum (unbenotet)

107994 VU - Principles of Data- and Knowledge-Base Systems

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
Alle	V	Mo	14:00 - 16:00	wöch.	2.70.0.10	08.04.2024	Prof. Dr. Torsten Schaub
1	U	Mo	16:00 - 18:00	wöch.	2.70.0.10	08.04.2024	Francois Laferriere
2	U	Mo	16:00 - 18:00	wöch.	2.70.0.08	08.04.2024	Francois Laferriere
3	U	Mo	16:00 - 18:00	wöch.	2.70.0.09	08.04.2024	Jana Schulz
	Für Lehramtsstudierende.						

Kommentar

Moodle course: moodle

Literatur

Principles of Database & Knowledge-Base Systems by Jeffrey D. Ullman W. H. Freeman & Co. New York, NY, USA

Leistungen in Bezug auf das Modul

SL 550821 - Übung (unbenotet)

INF-1060 - Software Engineering I

107998 VU - Research Software Engineering

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
Alle	V	Do	10:00 - 12:00	wöch.	2.70.0.11	11.04.2024	Prof. Dr. Anna-Lena Lamprecht
1	U	Di	12:00 - 14:00	wöch.	2.70.0.01	09.04.2024	Prof. Dr. Anna-Lena Lamprecht
2	U	Di	16:00 - 18:00	wöch.	2.70.0.01	09.04.2024	Prof. Dr. Anna-Lena Lamprecht

Kommentar

Software is vital for contemporary research: The most precious data is worthless without suitable software to process and analyze it. Over the past decade, Research Software Engineering (RSE) has formed as a new discipline to professionalize the development of software for scientific applications.

This course is an introduction to Research Software Engineering. It is intended for students who are already using Python (or a similar programming language) for data analysis, and who want to take their coding and software development skills to the next level. The course covers topics like version control with Git/GitLab, coding standards, development processes, requirements analysis, software architectures and design, testing and error handling, software licensing, software publication and citation, building command-line tools, configurable programs, creating packages, and workflow automation.

The weekly lectures are accompanied by exercise sessions to practice the concepts and techniques discussed. Students will furthermore work on two research software projects during the course, the first individually and the second in an interdisciplinary team. For both, students are invited to bring their own research ideas and problems.

Voraussetzung

The course assumes basic programming skills in Python (e.g. as acquired in "Grundlagen der Programmierung") and builds on that. You should be comfortable doing things like reading data from files and writing loops, conditionals, and functions. If you know another imperative programming language well, you can probably manage to pick up enough Python during the course.

Literatur

The course uses the textbook "Research Software Engineering with Python" (D. Irving et al., 2021, https://merely-useful.tech/py-rse/index.html) and selected additional material (provided in the course).

Leistungsnachweis

Projects and (written or oral) exam.

Bemerkung

The course "Forschungsdatenmanagement/Research Data Management" (taught by Prof. Dr. Lucke) complements this course with a focus on how to manage research data professionally.

Lerninhalte

Learning outcomes of this course include:
- Organize small and medium-sized data science projects.
- Use the Unix shell to efficiently manage your data and code.
- Write Python programs that can be used on the command line.
- Use Git to track and share your work.
- Work productively in a small team where everyone is welcome.
- Enable users to configure your software without modifying it directly.
- Analyse requirements and develop suitable software architectures.
- Organise code in a modular and sustainable way.
- Test your software and know which parts have not yet been tested.
- Find, handle, and fix errors in your code.
- Publish your code and research in open and reproducible ways.
- Create Python packages that can be installed in standard ways.
- Use Make, SnakeMake and other workflow managers to automate complex workflows.

Kurzkommentar

Bitte beachten: Die Belegung dieses Kurses als Modul INF-1060 ist **nur** für den BSc-Studiengang Kognitionswissenschaften vorgesehen.

Zielgruppe

Students from all disciplines who have at least basic programming skills (preferably in Python) and want to learn more about conducting research software projects professionally.

Leistungen in Bezug auf das Modul

PNL 550942 - Vorlesung und Übung (unbenotet)

107999 PJ - Research Software Engineering

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
1	PJ	N.N.	N.N.	wöch.	N.N.	N.N.	Prof. Dr. Anna-Lena Lamprecht

Kurzkommentar

Bitte beachten: Die Belegung dieses Kurses als Modul INF-1060 ist **nur** für den BSc-Studiengang Kognitionswissenschaften vorgesehen.

Leistungen in Bezug auf das Modul

SL 550941 - Projekt (unbenotet)

INF-1060 - Software Engineering I (auslaufend)

Dieses Modul gilt, aufgrund einer Änderungssatzung, nur noch für Studierende, die das Modul vor dem 01.10.2022 begonnen haben. Das Modul läuft spätestens am 30.09.2024 aus.

107998 VU - Research Software Engineering

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
Alle	V	Do	10:00 - 12:00	wöch.	2.70.0.11	11.04.2024	Prof. Dr. Anna-Lena Lamprecht
1	U	Di	12:00 - 14:00	wöch.	2.70.0.01	09.04.2024	Prof. Dr. Anna-Lena Lamprecht
2	U	Di	16:00 - 18:00	wöch.	2.70.0.01	09.04.2024	Prof. Dr. Anna-Lena Lamprecht

Kommentar

Software is vital for contemporary research: The most precious data is worthless without suitable software to process and analyze it. Over the past decade, Research Software Engineering (RSE) has formed as a new discipline to professionalize the development of software for scientific applications.

This course is an introduction to Research Software Engineering. It is intended for students who are already using Python (or a similar programming language) for data analysis, and who want to take their coding and software development skills to the next level. The course covers topics like version control with Git/GitLab, coding standards, development processes, requirements analysis, software architectures and design, testing and error handling, software licensing, software publication and citation, building command-line tools, configurable programs, creating packages, and workflow automation.

The weekly lectures are accompanied by exercise sessions to practice the concepts and techniques discussed. Students will furthermore work on two research software projects during the course, the first individually and the second in an interdisciplinary team. For both, students are invited to bring their own research ideas and problems.

Voraussetzung

The course assumes basic programming skills in Python (e.g. as acquired in "Grundlagen der Programmierung") and builds on that. You should be comfortable doing things like reading data from files and writing loops, conditionals, and functions. If you know another imperative programming language well, you can probably manage to pick up enough Python during the course.

Literatur

The course uses the textbook "Research Software Engineering with Python" (D. Irving et al., 2021, https://merely-useful.tech/py-rse/index.html) and selected additional material (provided in the course).

Leistungsnachweis

Projects and (written or oral) exam.

Bemerkung

The course "Forschungsdatenmanagement/Research Data Management" (taught by Prof. Dr. Lucke) complements this course with a focus on how to manage research data professionally.

Lerninhalte

Learning outcomes of this course include:
- Organize small and medium-sized data science projects.
- Use the Unix shell to efficiently manage your data and code.
- Write Python programs that can be used on the command line.
- Use Git to track and share your work.
- Work productively in a small team where everyone is welcome.
- Enable users to configure your software without modifying it directly.
- Analyse requirements and develop suitable software architectures.
- Organise code in a modular and sustainable way.
- Test your software and know which parts have not yet been tested.
- Find, handle, and fix errors in your code.
- Publish your code and research in open and reproducible ways.
- Create Python packages that can be installed in standard ways.
- Use Make, SnakeMake and other workflow managers to automate complex workflows.

Kurzkommentar

Bitte beachten: Die Belegung dieses Kurses als Modul INF-1060 ist **nur** für den BSc-Studiengang Kognitionswissenschaften vorgesehen.

Zielgruppe

Students from all disciplines who have at least basic programming skills (preferably in Python) and want to learn more about conducting research software projects professionally.

Leistungen in Bezug auf das Modul

PNL 550912 - Vorlesung und Übung (unbenotet)

107999 PJ - Research Software Engineering

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
1	PJ	N.N.	N.N.	wöch.	N.N.	N.N.	Prof. Dr. Anna-Lena Lamprecht

Kurzkommentar

Bitte beachten: Die Belegung dieses Kurses als Modul INF-1060 ist **nur** für den BSc-Studiengang Kognitionswissenschaften vorgesehen.

Leistungen in Bezug auf das Modul

PNL 550913 - Projekt (unbenotet)

INF-6010 - Praxis der Programmierung

107992 VU - Praxis der Programmierung

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
Alle	V	Fr	10:00 - 12:00	wöch.	2.27.1.01	12.04.2024	Dr. Henning Bordihn
1	U	Di	10:00 - 12:00	wöch.	2.70.0.01	09.04.2024	Dr. Henning Bordihn, Jana Schulz
	Für Lehramtsstudierende.						
2	U	Mi	14:00 - 16:00	wöch.	2.70.0.01	10.04.2024	Dr. Henning Bordihn
3	U	Mi	16:00 - 18:00	wöch.	2.70.0.01	10.04.2024	Dr. Henning Bordihn
4	U	Mo	12:00 - 14:00	wöch.	2.70.0.01	08.04.2024	Dr. Henning Bordihn, Jana Schulz
	Für Lehramtsstudierende.						
4	U	Di	14:00 - 16:00	Einzel	2.70.0.11	21.05.2024	Dr. Henning Bordihn, Jana Schulz
	Für Lehramtsstudierende.						
5	U	Do	12:00 - 14:00	wöch.	2.70.0.01	11.04.2024	Dr. Henning Bordihn

Voraussetzung

Kenntnisse aus Grundlagen der Programmierung

Leistungsnachweis

Klausur am Ende des Vorlesungszeitraums

Prüfungsnebenleistung zur Zulassung zur Prüfung: zwei Testate (Programmieraufgaben)

Prüfungsnebenleistung zum Abschluss des Moduls: zwei Programmierprojekte

Lerninhalte

Programmierung in einer imperativ-prozeduralen Programmiersprache wie beispielsweise C,
Objektorientierte Programmierung, beispielsweise in der Programmiersprache Java,
Implementierung von Algorithmen und Datenstrukturen

Leistungen in Bezug auf das Modul

PNL 555513 - Rechnerübung (unbenotet)

INF-DDI-1 - Didaktik der Informatik I

107967 VU - Didaktik der Informatik I

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
1	U	Do	12:00 - 14:00	wöch.	2.70.0.08	11.04.2024	N.N.
1	V	Do	14:00 - 16:00	wöch.	2.70.0.08	11.04.2024	N.N.

Kommentar

http://www.informatikdidaktik.de/Lehre/ddi1

Leistungsnachweis

Regelmäßige und aktive (!) Mitarbeit in den Übungen. Eine Abschlussnote wird bei erfolgreicher Teilnahme an einem Prüfungsgespräch erteilt.

Leistungen in Bezug auf das Modul

PNL 555331 - Einführung in die Didaktik der Informatik I (unbenotet)

108184 S - Schulpraktische Studien

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
1	S	N.N.	N.N.	Block	N.N.	N.N.	Alexander Hacke

Kommentar

Ausgangspunkt. Wie lernt man zu unterrichten? Neben einer genauen Kenntnis über die didaktischen und methodischen Hintergründe des Unterrichtens benötigt man vor allem Erfahrung. Erste Erfahrungen im Unterrichten können in dieser Veranstaltung erworben werden. Wir werden schrittweise in die Didaktik und Methodik des Informatikunterrichts einführen, zunächst beobachtend, dann experimentell unter Laborbedingungen, und schließlich in der Realität an einer Partnerschule in der näheren Umgebung. Lernziele: * Unterrichtsbeobachtung * Unterrichtsanalyse * Unterrichtsvorbereitung (fachwissenschaftliche und didaktische Analyse von Unterrichtsgegenständen) * Unterrichten im Kleinen (Microteaching) und im Großen (an der Partnerschule) * Unterrichtsauswertung * Einblick in die Wirklichkeit des Informatikunterrichts

Voraussetzung

GdP1, GdP2, RNB 1 u. 2, DdI 1

Leistungsnachweis

- Microteaching - Vorbereitende Ausarbeitung der Unterrichtsplanung - Abschlussbericht

Bemerkung

Termin nach Aushang/Homepage

Leistungen in Bezug auf das Modul

PNL 555332 - Fachdidaktische Tagespraktika (SPS) und Vorbereitungs-, Begleit- und Nachbereitungsseminar zu den Fachdidaktischen Tagespraktika (unbenotet)

MAT-1100 - Mathematik für Informatik I

Abkürzungen entnehmen Sie bitte Seite 4

Für dieses Modul werden aktuell keine Lehrveranstaltungen angeboten

MAT-1101 - Mathematik für Informatik II

Für dieses Modul werden aktuell keine Lehrveranstaltungen angeboten

Wahlpflichtmodule

INF-1040 - Konzepte paralleler Programmierung

107984 V - Konzepte paralleler Programmierung

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
1	V	Mi	10:00 - 12:00	wöch.	2.25.F1.01	10.04.2024	Prof. Dr. Bettina Schnor

Kommentar

Der Kurs besteht aus Vorlesung und Übung. An der richtigen Darstellung in PULS wird noch gearbeitet.

Für weitere Informationen siehe auch die Webseite https://www.cs.uni-potsdam.de/bs/teaching/docs/courses/

Voraussetzung

Vorlesung Grundlagen Betriebssysteme und Rechnernetze

Leistungsnachweis

mindesten 50% der Hausaufgabenpunkte, um zur Klausur zugelassen zu werden. Die Note ergibt scih aus der Klausurnote.

Bemerkung

Mit Beginn der Einschreibefrist in PULS ist auch die Einschreibung zum zugehörigen Moodle-Kurs "Konzepte paralleler Programmierung" über diesen Link möglich und erforderlich: https://moodle2.uni-potsdam.de/course/view.php?id=33419

107985 U - Konzepte paralleler Programmierung

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
1	U	Do	14:00 - 16:00	wöch.	2.70.0.11	11.04.2024	Petra Vogel, Prof. Dr. Bettina Schnor, Max Schrötter

Kommentar

Achtung! Der Kurs besteht aus Vorlesung und Übung!

Weitere Informatonen siehe Webseite https://www.cs.uni-potsdam.de/bs/teaching/courses/

Voraussetzung

Vorlesung Grundlagen Betriebssysteme und Rechnernetze

Leistungsnachweis

mindesten 50% der Hausaufgabenpunkte, um zur Klausur zugelassen zu werden. Die Note ergibt scih aus der Klausurnote.

Bemerkung

Mit Beginn der Einschreibefrist in PULS ist auch die Einschreibung zum zugehörigen Moodle-Kurs "Konzepte paralleler Programmierung"erforderlich.

Leistungen in Bezug auf das Modul

PNL 550712 - Vorlesung und Übung (unbenotet)

INF-1080 - Künstliche Intelligenz

Für dieses Modul werden aktuell keine Lehrveranstaltungen angeboten

INF-2010 - Rechnernetze

107969 VU - Distributed Systems

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
1	V	Di	12:00 - 14:00	wöch.	2.70.0.09	09.04.2024	Prof. Dr. Sukanya Bhowmik
1	U	Mi	12:00 - 14:00	wöch.	2.70.0.08	10.04.2024	Petra Vogel, Prof. Dr. Sukanya Bhowmik

Kommentar

Die Vorlesung gibt eine Einführung in die Konzepte verteilter Systeme. Themengebiete sind u.a. Kommunikation (RPC, Publish/Subscribe, Multicast, REST) in Verteilten Systemen, verteilte Dateisysteme, Synchronisationstechniken für verteilte Anwendungen und Lastverteilung (Webserver, Cloud Computing).

Für weitere Informationen siehe auch die Webseite https://www.cs.uni-potsdam.de/bs/teaching/docs/courses/

Start der Veranstaltung ist in der zweiten Vorlesungswoche: Vorlesung am 26.4.22, Übung am 27.4.22!

Voraussetzung

Grundlagen Betriebssysteme und Rechnernetze

Leistungsnachweis

Hat man mindestens 50% der Hausaufgabenpunkte erreicht, wird man zur Klausur zugelassen.

Bemerkung

Mit Beginn der Einschreibefrist in PULS ist auch die Einschreibung zum zugehörigen Moodle-Kurs "Verteilte Systeme" über diesen Link möglich und erforderlich: https://moodle2.uni-potsdam.de/course/view.php?id=32996 **Achtung! Erst ab 19.4.2022!**

Leistungen in Bezug auf das Modul

PNL 551212 - Vorlesung und Übung (unbenotet)

INF-2030 - Netzbasierte Datenverarbeitung

108065 VU - Netzbasierte Datenverarbeitung

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
Alle	V	Mo	08:00 - 10:00	wöch.	2.70.0.11	08.04.2024	Prof. Dr.-Ing. Ulrike Lucke
1	U	Do	14:00 - 16:00	wöch.	2.70.0.09	11.04.2024	Dr. rer. nat. Tobias Moebert
2	U	Fr	10:00 - 12:00	wöch.	2.70.0.11	12.04.2024	Dr. rer. nat. Tobias Moebert

Links:

Moodle https://moodle2.uni-potsdam.de/course/edit.php?id=40206

Kommentar

Aktuelle (Multimedia-)Applikationen setzen i.d.R. nicht nur leistungsfähige Clients voraus, sondern sind für den Einsatz in Rechnernetzen konzipiert. Häufig handelt es sich um parallelisierte Anwendungen. Neben Servern und Netzwerktechnik erfordert dies spezielle netzbasierte Architekturen, um die Interoperabilität der einzelnen Komponenten in heterogenen Umgebungen zu gewährleisten. Die Vorlesung geht nach einer Einführung in die Thematik auf ausgewählte Konzepte ein, wie z.B. Grid Computing, Peer-to-Peer Kommunikation oder Service-Orientierte Architekturen. In der begleitenden Übung werden die vorgestellten Konzepte vertiefend betrachtet und an einem Programmierbeispiel selbst analysiert.

Voraussetzung

Netzwerk- und Betriebssystem-Kenntnisse

Bemerkung

Bitte melden Sie sich im Moodle-Kurs an:
https://moodle2.uni-potsdam.de/course/edit.php?id=40206
Dort finden Sie alle weiteren Materialien und Informationen zum Kurs.

Leistungen in Bezug auf das Modul
PNL 551512 - Vorlesung und Übung (unbenotet)

INF-2031 - Multimediatechnologie

Für dieses Modul werden aktuell keine Lehrveranstaltungen angeboten

INF-2040 - Software Engineering II

108003 VU - Software Engineering II

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
Alle	V	Do	12:00 - 14:00	wöch.	2.70.0.11	11.04.2024	Prof. Dr. Anna-Lena Lamprecht
Alle	V	Do	12:00 - 14:00	Einzel	2.70.0.09	27.06.2024	Prof. Dr. Anna-Lena Lamprecht
1	U	Mo	14:00 - 16:00	wöch.	2.70.0.01	08.04.2024	Prof. Dr. Anna-Lena Lamprecht
2	U	Mo	16:00 - 18:00	wöch.	2.70.0.01	08.04.2024	Prof. Dr. Anna-Lena Lamprecht

Leistungen in Bezug auf das Modul
PNL 551712 - Vorlesung und Übung (unbenotet)

INF-2061 - Information und Komplexität

107963 VU - Codierungstheorie

Gruppe	Art	Tag	Zeit	Rhythmus	Veranstaltungsort	1.Termin	Lehrkraft
1	V	Do	16:00 - 18:00	wöch.	2.70.0.09	11.04.2024	Prof. Dr. Michael Gössel
1	U	Fr	14:00 - 16:00	wöch.	2.70.0.08	12.04.2024	Alexander Benjamin Glätzer, Prof. Dr. Michael Gössel, Alexander Benedict Behrens
1	U	Fr	14:00 - 16:00	Einzel	Online.Veranstalt	28.06.2024	Prof. Dr. Michael Gössel, Alexander Benjamin Glätzer

Kommentar

Sprache: Deutsch/Englisch je nach Fähigkeiten der Teilnehmer und Teilnehmerinnen

Die Vorlesung Codierungstheorie führt in die Grundlagen der Fehlererkennung und Fehlerkorrektur von Daten unter Verwendung von fehlererkennenden und fehlercorrigierenden Codes ein. In der Codierungstheorie werden mathematische Begriffe und Ergebnisse der linearen Algebra und der Theorie endlicher Körper unmittelbar in technische Lösungen umgesetzt, was ein tiefes Verständnis und eine goße Schönheit technischer Lösungen ermöglicht.

Durch die extreme Verkleinerung elektronischer Bauelemente insbesondere im Speicherbereich nimmt deren Fehleranfälligkeit ständig zu, weshalb Fehlererkennung und Fehlerkorrektur insbesondere für sicherheitskritische Anwendungen, aber nicht nur für diese, von wachsender Bedeutung ist. Z. B. durch das Internet der Dinge sind zunehmend fehlersichere Datenübertragungen auch zwischen Geräten erforderlich

In der Vorlesung werden die folgenden linearen Codes detailliert behandelt: Paritätscode, Hamming-Code, Hsiao-Code, zyklische Code, BCH-Codes und Reed-Solomon-Codes, Low-Density-Parity Codes. Auf nichtlineare Codes wird kurz eingegangen. Die Möglichkeiten und Grenzen der Fehlererkennung und der Fehlerkorrektur und auch der damit zusammenhängenden ethischen Probleme werden ausführlich besprochen.

Nach einem erfolgreichen Abschluss der Vorlesung sind die Teilnehmer/innen in der Lage, fehlererkennende und fehlerkorrigierende Code anzuwenden und auf konkrete praktische Probleme anzupassen.

Lösungen der Übungsaufgaben werden von den Studenten vorgestellt und diskutiert (in Deutsch oder Englisch). 50% der Aufgaben müssen richtig gelöst werden, um die Veranstaltung positiv bewertet zu bestehen.

Eine regelmäßige Teilnahme (80 %) wird erwartet.

The solutions of the exercises will be presented (in German, or depending on the participants in English) by the students and dicussend. 50 % of the exercises have to be correctly solved by a studend to be qualified for the examination which can be done in Geman and English.

Voraussetzung

Grundkenntnisse in Mathematik, insbesondere Elementare Lineare Algebra. Grundkentnisse der Theorie endlicher Körper sind von Vorteil, aber nicht Voraussetzung, da diese in der Vorlesung eingeführt werden.

Von Vorteil sind ebenfalls Grundkenntnisse in technischer Informatik, die auch in der Vorlesung/Übung gelernt werden können.

Literatur

Die Standars-Ergebnisse der Codierungstheorie sind in einer Vielzahl von Lehrbüchern dargestellt.

Beispielsweise in

Rohling, H. "Einführung in die Informations-und Codierungstheorie", Teubner, 1995

Lin, S. and Costello, "Error Control Coding", 2. Auflage, 2004, Person Education und Prentice-Hall, preisswerte 1. Auflage, 1983 (Für die Vorlesung sind nur ausgewählte Abschnitte Wichtig)

E. Fujiwara "Code-Design for Dependable Systems", 2006, Wiley,preiswerte vorige Auflage als Rao, T. and Fujiwara, E " Eror Control Coding for Computer Systems", Prentice Hasll 1989 (Für die Vorlesung sind nur ausgewählte Abschnitte Wichtig)

spezielle Literatur, insbesonderre auch Patente, zu einzelnen Themen wird in der Vorlesung/Übung genannt

Leistungsnachweis

Die Bewertung der Veranstaltung erfolgt in diesem Semester dadurch, dass jeder Teilnehmer einen ca 30 minütigen Vortrag in einer Uebung zu einem Thema haelt, das die Vorlesung ergänzt (Publikation oder Patent). Ausserdem erfolgt eine mündliche Prüfung. Weiterhin ist erforderlich, dass 50 % der Punkte der Übungsaufgaben erreicht werden, die wöchentlich abzugeben sind. Der Vortrag ist in Deutsch (möglichst) oder falls erfordelich in Englisch möglich. Die mündliche Prüfung kann auf Wunsch in Deutsch, oder Englisch erfolgen. Zur Erarbeitung des Vortrages ist eine persönliche Konsultation des/der Vortragenden (on-line, zoom) von ca 1 Stunde mit dem Dozenten vorgesehen.

Vortrag und mündliche Prüfung werden jeweils mit 50 % gewertet.

Bemerkung

Die Vorlesung erfolgt in Prasenz, es wird eine On-line-Übertragung angestrebt.

Lerninhalte

Kenntis und Verständnis bekannter Codes und neuerer Codes,

Fähigkeit zum eigenständigen Lösen praktischer Aufgaben der Codierungstheorie und zum Literaturstudium wissenschaftliche Arbeiten und von Patenten unter Verwendung der in der Veranstaltung gelernten Grundbegriffe,

Fähigkeit zum wissenschaftlichen Arbeit etwa im Rahmen einer Bacherlor- oder Masterarbeit, und zur Anwendung im Beruf.

Grundlegendes Verständnis der Möglichkeiten der Fehlererkennung und Fehlerkorrektkur, die sich aus dem Wahrscheinlichkeitscharakter der auftretenden Fehler ergeben und der daraus resultierenden ethischen Probleme für das eigene Tun oder Nichttun.

Kurzkommentar

siehe Bemerkungen

Zielgruppe

Bachelor und Master-Studenten, die in der Lage sein wollen, Datenübertragung und Datenspeicherung fehlertolerant unter Verwendung von fehlererkennenden und fehlerkorrigierenden Codes zu sichern, oder die auf dem Gebiet der Codierungsthorie wissenschaftlich arbeiten und weiterentwickeln wollen und/ oder beabsichtigen, ihre Bacherlor-oder Master-Arbeit auf dem Gebiet der Codierungstheorie zu schreiben.

Ein Interesse an der Umsetzung theoretischer Resultate in technische Lösungen oder in algorithmische Lösungen wird erwartet.

Leistungen in Bezug auf das Modul

SL	552112 - Vorlesung und Übung (unbenotet)

Akademische Grundkompetenzen

Für dieses Modul werden aktuell keine Lehrveranstaltungen angeboten

Glossar

Die folgenden Begriffserklärungen zu Prüfungsleistung, Prüfungsnebenleistung und Studienleistung gelten im Bezug auf Lehrveranstaltungen für alle Ordnungen, die seit dem WiSe 2013/14 in Kranft getreten sind.

Prüfungsleistung

Prüfungsleistungen sind benotete Leistungen innerhalb eines Moduls. Aus der Benotung der Prüfungsleistung(en) bildet sich die Modulnote, die in die Gesamtnote des Studiengangs eingeht. Handelt es sich um eine unbenotete Prüfungsleistung, so muss dieses ausdrücklich („unbenotet“) in der Modulbeschreibung der fachspezifischen Ordnung geregelt sein. Weitere Informationen, auch zu den Anmeldemöglichkeiten von Prüfungsleistungen, finden Sie unter anderem in der Kommentierung der BaMa-O

Prüfungsnebenleistung

Prüfungsnebenleistungen sind für den Abschluss eines Moduls relevante Leistungen, die – soweit sie vorgesehen sind – in der Modulbeschreibung der fachspezifischen Ordnung beschrieben sind. Prüfungsnebenleistungen sind immer unbenotet und werden lediglich mit "bestanden" bzw. "nicht bestanden" bewertet. Die Modulbeschreibung regelt, ob die Prüfungsnebenleistung eine Teilnahmevoraussetzung für eine Modulprüfung oder eine Abschlussvoraussetzung für ein ganzes Modul ist. Als Teilnahmevoraussetzung für eine Modulprüfung muss die Prüfungsnebenleistung erfolgreich vor der Anmeldung bzw. Teilnahme an der Modulprüfung erbracht worden sein. Auch für Erbringung einer Prüfungsnebenleistungen wird eine Anmeldung vorausgesetzt. Diese fällt immer mit der Belegung der Lehrveranstaltung zusammen, da Prüfungsnebenleistung im Rahmen einer Lehrveranstaltungen absolviert werden. Sieht also Ihre fachspezifische Ordnung Prüfungsnebenleistungen bei Lehrveranstaltungen vor, sind diese Lehrveranstaltungen zwingend zu belegen, um die Prüfungsnebenleistung absolvieren zu können.

Studienleistung

Als Studienleistung werden Leistungen bezeichnet, die weder Prüfungsleistungen noch Prüfungsnebenleistungen sind.

Impressum

Herausgeber
Am Neuen Palais 10
14469 Potsdam

Telefon: +49 331/977-0
Fax: +49 331/972163
E-mail: presse@uni-potsdam.de
Internet: www.uni-potsdam.de

Umsatzsteueridentifikationsnummer
DE138408327

Layout und Gestaltung
jung-design.net

Druck
14.3.2024

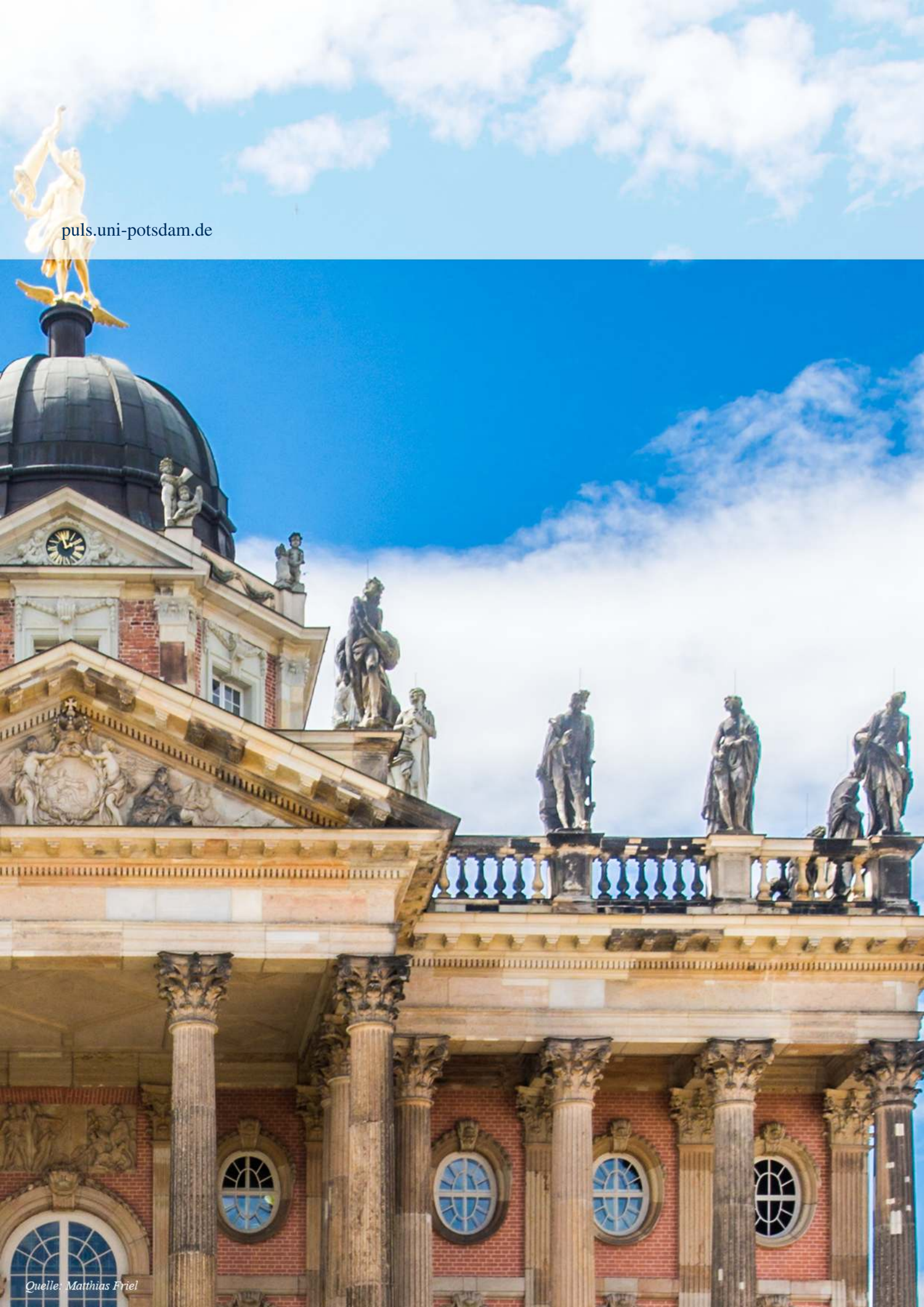
Rechtsform und gesetzliche Vertretung
Die Universität Potsdam ist eine Körperschaft des Öffentlichen Rechts. Sie wird gesetzlich vertreten durch Prof. Oliver Günther, Ph.D., Präsident der Universität Potsdam, Am Neuen Palais 10, 14469 Potsdam.

Zuständige Aufsichtsbehörde
Ministerium für Wissenschaft, Forschung und Kultur des Landes Brandenburg
Dortustr. 36
14467 Potsdam

Inhaltliche Verantwortlichkeit i. S. v. § 5 TMG und § 55 Abs. 2 RStV
Referat für Presse- und Öffentlichkeitsarbeit
Referatsleiterin und Sprecherin der Universität
Silke Engel
Am Neuen Palais 10
14469 Potsdam
Telefon: +49 331/977-1474
Fax: +49 331/977-1130
E-mail: presse@uni-potsdam.de

Die einzelnen Fakultäten, Institute und Einrichtungen der Universität Potsdam sind für die Inhalte und Informationen ihrer Lehrveranstaltungen zuständig.

puls.uni-potsdam.de



Quelle: Matthias Friel