TECHNISCHE HOCHSCHULE DEGGENDORF

Module Guide
Artificial Intelligence and Date Science

Faculty Computer Science
Examination regulations 13.11.2020
Date: 30.07.2021 08:12

Table of Contents

AID-01 Artificial Intelligence and Software Development

Module code	AID-01
Module coordination	Prof. Dr. Cezar Ionescu
Course number and name	AID-01 Artificial Intelligence and Software Development
Lecturer	Prof. Dr. Cezar Ionescu
Semester	1
Duration of the module	1 semester
Module frequency	annually
Course type	required course
Level	postgraduate
Semester periods per week (SWS)	4
ECTS	5
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	written ex. 90 min.
Duration of Examination	90 min.
Weight	5/120
Language of Instruction	German

Module Objective

AI software development requires knowledge of multiple programming paradigms. This course introduces the students to functional, logic, and probabilistic programming with applications to AI. Students will experience each paradigm using a different programming language, but they will learn to relate them to the mainstream programming language Python.

After completing the course, the students can

- define the programming paradigms used in AI
- decide which programming paradigm best fits a problem domain
- design solutions using the appropriate programming paradigm
- translate the solution using a mainstream programming language (Python)

Entrance Requirements

Learning Content

- Introduction: AI and software development
- Introduction to functional programming using Haskell
- Symbolic differentiation as an application of functional programming
- Implementing search strategies for board games
- Functional programming in Python
- Property-based testing in Haskell and Python
- Logic and functional programming: dependent types
- Logic programming and SAT/SMT solvers
- Probabilistic programming using WebPPL
- Probabilistic programming in Python

Teaching Methods

lectures, seminars

Recommended Literature

- Programming in Haskell 2nd Ed, Graham Hutton, CUP 2016
- Conceptual Programming with Python, Thorsten Alternkirch and Isaac Triguero, Lulu 2019
- Type-Driven Development with Idris, Edwin Brady, Manning 2017
- Modelling Agents with Probabilistic Programming Languages, Owain Evans et al., electronic
- SAT/SMT by Example, Dennis Yurichev, electronic

AID-02 Theoretical Fundamentals of Artificial Intelligence

Module code	AID-02
Module coordination	Prof. Dr. Andreas Berl
Course number and name	AID-02 Theoretical Fundamentals of Artificial Intelligence
Lecturers	Prof. Dr. Peter Faber Prof. Dr. Peter Jüttner
Semester	1
Duration of the module	1 semester
Module frequency	annually
Course type	required course
Level	postgraduate
Semester periods per week (SWS)	6
ECTS	8
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	written ex. 90 min.
Duration of Examination	90 min.
Weight	8/120
Language of Instruction	English

Module Objective

In this module, students become familiar with the theoretical foundations of computer science such as machine models with applications in complexity theory and predictability theory, formal languages and their hierarchies, etc. They gain an understanding of complexity estimates and verification options, as well as the basic functions of data processing systems. This course lays the foundation for understanding the theoretical background for algorithms and methods inside and outside of AI applications.

Entrance Requirements

Learning Content

Main topics:

Semantics, predictability and complexity theory: Here theoretical (machine) models are used for the formal defi

- Semantics
- Definition of semantics, history
- Semantics of recursive functions (fixed point theory)
- Induction evidence
- Operative semantics
- Axiomatic semantics
- Predictability
- Definition
- Unpredictable functions
- Turing machines and their programming
- Turing predictability
- LOOP, WHILE, GOTO predictability
- Complexity theory
- Definition
- O notation
- Complexity levels

Formal languages and compiler construction for natural language processing: This course illuminates the theoretical background behind formal languages and langage processing.

- Introduction and Translators I
- Translators II / Formal Languages I
- Formal languages II / III
- Lexical Analysis I / II
- Syntactic analysis I / II
- Syntactic analysis III / IV
- Syntax directed translation I / II
- Summary, possibly further topics (e.g. on optimizing compiler construction) - wrap-up, possibly further topics (e.g., optimizing compilers)

Teaching Methods

lectures, seminars

Recommended Literature

Basic literature:

- John Longley, Lessons in Formal Programming Language Semantics, University of Edinburgh, 2003
- F.L. Bauer, H. Wössner: Algorithmische Sprache und Programmentwicklung, Springer Verlag 1984 (availble also in English)
- Rudolf Berghammer: Semantik von Programmiersprachen, Logos Verlag, 2001
- Juraj Hromkovic: Theoretische Informatik, Springer Verlag
- Uwe Schöning: Theoretische Informatik - kurz gefasst. Spektrum, 2008
- Hopcroft, Motwani, Ullman: Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, 2001
- Hopcroft, Motwani, Ullman: Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie, Pearson, 2002.
- Compilers Principles, Techniques, and Tools; Aho, Lam, Sethi, Ullmann; 2nd edition; Addison-Wesley; 2007
- Engineering a compiler; Cooper, Torczon; 2nd Edition, Morgan Kaufmann 2012
- Introduction to Automata Theory, Languages, and Comoputation; Hopcroft, Motwani, Ullman; Addison-Wesley; 2001

AID-03 Advanced Machine Learning

Module code	AID-03
Module coordination	Prof. Dr. Christina Bauer
Course number and name	AID-03 Advanced Machine Learning
Lecturer	Prof. Dr. Christina Bauer
Semester	1
Duration of the module	1 semester
Module frequency	annually
Course type	required course
Level	postgraduate
Semester periods per week (SWS)	4
ECTS	5
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	written ex. 90 min.
Duration of Examination	90 min.
Weight	5/120
Language of Instruction	English

Module Objective

The course contains the theoretical foundations of machine learning. Topics include:

- Model assessment: bias-variance tradeoff, curse of dimensionality
- Loss functions, risks, and measures for the goodness of fit
- Universal consistency
- Lower bounds on error rates and rates of convergence
- Cross-validation
- Dimensionality reduction: principal component analysis, independent component analysis, cocktail party problem
- Optimization methods: gradient descent, convex optimization, quasi-Newton methods, Lagrange multipliers

- VC dimensions
- Biased data sets: class imbalance and covariate shift
- Applications to selected machine learning methods: neural networks, support vector machines, k-nearest neighbors, and others

Entrance Requirements

Learning Content

The course contains the theoretical foundations of machine learning. Topics include:
- Model assessment: bias-variance tradeoff, curse of dimensionality
- Loss functions, risks, and measures for the goodness of fit
- Universal consistency
- Lower bounds on error rates and rates of convergence
- Cross-validation
- Dimensionality reduction: principal component analysis, independent component analysis, cocktail party problem
- Optimization methods: gradient descent, convex optimization, quasi-Newton methods, Lagrange multipliers
- VC dimensions
- Biased data sets: class imbalance and covariate shift
- Applications to selected machine learning methods: neural networks, support vector machines, k-nearest neighbors, and others

Teaching Methods

seminar-style lessons with practice

AID-04 Faculty Elective 1 (FWPM)

Module code	AID-04
Module coordination	Prof. Dr. Andreas Berl
Course number and name	AID-04 Faculty Elective (FWPM)
Lecturer	Prof. Dr. Andreas Berl
Semester	1
Duration of the module	1 semester
Module frequency	each semester
Course type	compulsory course
Semester periods per week (SWS)	4
ECTS	5
Workload	Time of attendance: 0 hours Total: 0 hours
Language of Instruction	English

Module Objective

AID-04 Faculty Elective (FWPM)

Type of Examination

written ex. 90 min.

AID-05 Faculty Elective 2 (FWPM)

Module code	AID-05
Module coordination	Prof. Dr. Andreas Berl
Course number and name	AID-05 Faculty Elective 2 (FWPM)
Lecturer	Prof. Dr. Andreas Berl
Semester	1
Duration of the module	1 semester
Module frequency	each semester
Course type	compulsory course
Semester periods per week (SWS)	4
ECTS	5
Workload	Time of attendance: 0 hours Total: 0 hours
Language of Instruction	English

Module Objective

AID-06 Language Course: German or Czech

Module code	AID-06
Module coordination	Tanja Mertadana
Course number and name	AID-06 Language Course: German or Czech
Lecturer	Tanja Mertadana
Semester	1
Duration of the module	1 semester
Module frequency	each semester
Course type	required course
Semester periods per week (SWS)	2
ECTS	2
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	written ex. 60 min.
Duration of Examination	60 min.
Language of Instruction	German

Module Objective

AID-07 Information Theory

Module code	AID-07
Module coordination	Dr.-Ing. Ladislav Beránek
Course number and name	AID-07 Information Theory
Lecturer	Prof. Dr. Andreas Berl
Semester	2
Duration of the module	1 semester
Module frequency	annually
Course type	required course
Level	postgraduate
Semester periods per week (SWS)	3
ECTS	4
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	written ex. 90 min.
Duration of Examination	90 min.
Weight	4/120
Language of Instruction	English

Module Objective

The basis of information theory is the study of basic limits in the transmission and storage of information. This course deals with information theory and its applications. These are concepts such as entropy and information; lossless data compression; communication in the presence of noise, channel capacity and channel coding and coding on graphs. In this course, we will introduce these concepts and provide a practical view of their implementation.

Main topics:

1. Probability, Entropy
2. Element of the data compression, the source coding theorem
3. Symbol codes, stream Codes

4. Noisy-channel coding, communication over a noisy channel
5. - 6. Error-Correcting Codes and Real Channels
7. Coding - hash codes, binary codes
8. Good linear codes, message passing
9. - 10. Communication over Constrained Noiseless Channels
11. Elements from probability - exact marginalization (in trellises, in graphs)
12. - 13. Graph codes - low-density parity-check codes, convolutional codes, turbo codes

The content of the seminars is based on lectures. Practical tasks relating the lectures topics will be discussed at the seminars.

Entrance Requirements

Learning Content

This course is taught at the University of South Bohemia by Doc. Ing. Ladislav Beránek, CSc.

The basis of information theory is the study of basic limits in the transmission and storage of information. This course deals with information theory and its applications. These are concepts such as entropy and information; lossless data compression; communication in the presence of noise, channel capacity and channel coding and coding on graphs. In this course, we will introduce these concepts and provide a practical view of their implementation.

Main topics:

1. Probability, Entropy
2. Element of the data compression, the source coding theorem
3. Symbol codes, stream Codes
4. Noisy-channel coding, communication over a noisy channel
5. - 6. Error-Correcting Codes and Real Channels
7. Coding - hash codes, binary codes
8. Good linear codes, message passing
9. - 10. Communication over Constrained Noiseless Channels
11. Elements from probability - exact marginalization (in trellises, in graphs)
12. - 13. Graph codes - low-density parity-check codes, convolutional codes, turbo codes

The content of the seminars is based on lectures. Practical tasks relating the lectures topics will be discussed at the seminars.

Teaching Methods

lectures, seminars

Recommended Literature

Basic literature:

MACKAY, David J. C. *Information theory, inference, and learning algorithms*. Cambridge: Cambridge University Press, 2003. ISBN 978-0-521-64298-9.

Recommended literature:

HOST, S. *Information and Communication Theory*. Hoboken, NJ: Wiley-IEEE Press, 2019. ISBN 978-1119433781.

COVER, T. M. and Joy A. THOMAS. *Elements of information theory*. 2nd ed. Hoboken: Wiley-Interscience, c2006. ISBN 978-0-471-24195-9.

EL-GAMAL, A. and YOUNG-HAN, K. *Network information theory. Primera*. Cambridge: Cambridge Univerisity Press, 2011. ISBN 978-1-107-00873-1.

Study aids:

Materials for lectures and seminars in LMS Moodle.

AID-08 Mathematics for Artificial Intelligence and Data Science

Module code	AID-08
Module coordination	Dr.-Ing. Jan Valdman
Course number and name	AID-08 Mathematics for Artificial Intelligence and Data Science
Lecturer	Prof. Dr. Andreas Berl
Semester	2
Duration of the module	1 semester
Module frequency	annually
Course type	required course
Level	postgraduate
Semester periods per week (SWS)	4
ECTS	6
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	course assessment, written ex. 90 min.
Duration of Examination	90 min.
Weight	6/120
Language of Instruction	English

Module Objective

The course Mathematical Foundations and some Applications

1 - Linear Algebra: matrices, linear independence, row echelon form

2 - Analytic Geometry: norms, inner products, orthogonality, orthogonal projections

3 - Matrix Decompositions: eigenvalues, eigenvectors, Cholesky-, eigen-, singular-decompositions

4 - Vector Calculus: partial Differentiation and Gradients, Higher-Order Derivatives, Multivariate Taylor Series

5 - Optimization: Gradient descent, Constrained Optimization and Lagrange Multipliers, Convex Optimization

6 - Probability and Distributions: Bayes' Theorem, Gaussian distribution,

7 - Applications: Linear Regression, Principal Component Analysis

Entrance Requirements

Learning Content

This course is taught at the University of South Bohemia by Doc. Dr. rer. nat. Ing. Jan Valdman

The course Mathematical Foundations and some Applications

1 - Linear Algebra: matrices, linear independence, row echelon form

2 - Analytic Geometry: norms, inner products, orthogonality, orthogonal projections

3 - Matrix Decompositions: eigenvalues, eigenvectors, Cholesky-, eigen-, singular-decompositions

4 - Vector Calculus: partial Differentiation and Gradients, Higher-Order Derivatives, Multivariate Taylor Series

5 - Optimization: Gradient descent, Constrained Optimization and Lagrange Multipliers, Convex Optimization

6 - Probability and Distributions: Bayes' Theorem, Gaussian distribution,

7 - Applications: Linear Regression, Principal Component Analysis

Recommended Literature

Basic literature:

Mathematics for Machine Learning by Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong, ISBN 978-1108470049

Recommended literature:

Linear Algebra and Its Applications by David C. Lay, Steven R. Lay, Judi J. McDonald, ISBN 978-0321982384

Convex Optimization by Stephen Boyd, Lieven Vandenberghe , ISBN 978-0521833783

AID-09 Computational Intelligence

Module code	AID-09
Module coordination	Dr.-Ing. Miroslav Skrbek
Course number and name	AID-09 Computational Intelligence
Lecturer	Prof. Dr. Andreas Berl
Semester	2
Duration of the module	1 semester
Module frequency	annually
Course type	required course
Level	postgraduate
Semester periods per week (SWS)	0
ECTS	4
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	course assessment
Weight	3/120
Language of Instruction	English

Module Objective

This course is focused on advanced topic of computation intelligence algorithms. Students obtain practical experience with advanced recurrent neural networks, reinforcement learning, fuzzy systems and nature inspired optimization. The emphasis is on practical experience and individual student projects.

Entrance Requirements

Learning Content

Lectures:

1. Introduction, overview of neural networks
2. Advanced feed-forward neural network architectures

3. Recurrent and LSTM neural networks
4. Neural Turing machines
5. Fuzzy systems and neural fuzzy systems
6. Reinforcement learning, actor-critic architecture
7. Evolutionary techniques and nature inspired optimization

Labs: in the labs the students obtain practical experiences in advance neural network architectures and nature inspired optimization techniques. Benefits of commonly used neural frameworks will be used. The students pass through a series of laboratory tasks and s final project.

Remarks

Student evaluation:

Evaluation of student tasks (small range tasks). Evaluation of final project. Written test covering knowledge from labs and lectures.

Recommended Literature

Basic literature:

[1] Charu C. Aggarwal: Neural Networks and Deep Learning: A Textbook. Springer; 1st ed. 2018. ISBN: 978-3319944623

[2] Simeon Kostadinov: Recurrent Neural Networks with Python Quick Start Guide: Sequential learning and language modeling with TensorFlow. Packt Publishing, November, 2018. ISBN: 978-1789132335

[3] Lotfi A Zadeh (Author), Rafik A Aliev: Fuzzy Logic Theory and Applications: Part I and Part II. WSPC December 5, 2018. ISBN: 978-9813238176

[4] Applied Reinforcement Learning with Python: With OpenAI Gym, Tensorflow, and Keras, Paperback Apress, 1st edition, August, 2019. ISBN: 978-1484251263

Recommended literature:

[5] Xin-She Yang: Nature-Inspired Optimization Algorithms, 2nd Edition. Academic Press, September 2020. 978-0128219867

AID-10 Distributed Algorithms

Module code	AID-10
Module coordination	Dr.-Ing. Jan Fesl
Course number and name	AID-10 Distributed Algorithms
Lecturer	Prof. Dr. Andreas Berl
Semester	2
Duration of the module	1 semester
Module frequency	annually
Course type	required course
Level	postgraduate
Semester periods per week (SWS)	3
ECTS	4
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	course assessment, written ex. 90 min.
Duration of Examination	90 min.
Weight	4/120
Language of Instruction	English

Module Objective

Main topics:

1. Structure of distributed systems and applications, communication methods.
2. Procedural communication - XML-RPC, CORBA, Java RMI, SOAP, Node.js, Go
3. Model of distributed computation, asynchrony/synchrony.
4. Snapsots in distributed computation, logical time.
5. Election mechanisms in distributed systems.
6. Explicit access mechanisms.
7. Deadlock detection - device sharing, communication.
8. Termination detection in distributed algorithms.

9. Distributed shared memory - data sharing and replication.
10. Support and use of P2P, DHT technologies.
11. Agent based applications, mobility.
12. Cloud technologies, application methods.

Entrance Requirements

Learning Content

Main topics:
1. Structure of distributed systems and applications, communication methods.
2. Procedural communication - XML-RPC, CORBA, Java RMI, SOAP, Node.js, Go
3. Model of distributed computation, asynchrony/synchrony.
4. Snapsots in distributed computation, logical time.
5. Election mechanisms in distributed systems.
6. Explicit access mechanisms.
7. Deadlock detection - device sharing, communication.
8. Termination detection in distributed algorithms.
9. Distributed shared memory - data sharing and replication.
10. Support and use of P2P, DHT technologies.
11. Agent based applications, mobility.
12. Cloud technologies, application methods.

Teaching Methods

lecture, seminar

Remarks

students must complete semestral task

Recommended Literature

Basic literature:
Lynch N.A.: Distributed Algorithms.
Recommended literature:
Tel G.: Introduction to Distributed Algorithms.

Kshemkalyani A.D., Singhal M.: Distributed Computing: Principles, Algorithms, and Systems.

Coulouris G., Dellimore J., Kindberg T.: Distributed Systems.

AID-11 Advanced Data Storrages and Analyses

Module code	AID-11
Module coordination	Dr. Miloš Prokýšek
Course number and name	AID-11 Advanced Data Storrages and Analyses
Lecturer	Prof. Dr. Andreas Berl
Semester	2
Duration of the module	1 semester
Module frequency	annually
Course type	required course
Level	postgraduate
Semester periods per week (SWS)	4
ECTS	6
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	course assessment, written ex. 90 min.
Duration of Examination	90 min.
Weight	6/120
Language of Instruction	English

Module Objective

1 Relational and noSQL data storages
1 Datawarehouse
1 Star, Snowflake and Data Vault patterns
1 ETL, OLAP, OLTP
1 Distributed database systems
1 CAP theorem
1 Master-slave, mirroring, sharding
1 NoSQL database systems
1 Key-value
1 Column oriented
1 Document databases

1 Graph databases
1 Time-series databases
1 Large datasets
1 Velocity, variability, volume
1 Unstructured data
1 ELT processing, curated data
1 Stream data processing
1 Buffering
1 Distribution
1 Storing
1 Real-time processing
1 Data mining
1 Data sources and datatypes
1 Data matrix
1 Data storages
1 Level of Míry podobnosti, metody shlukové analýz
1 Basic data models
1 Linear and log-linear regression
1 Data modeling
1 Decission trees, associaton rules
1 Classificators
1 k-NN
1 naive bayes classificator
1 Data lakes
1 Distributed filesystems
1 Hadoop - family solutions

Entrance Requirements

Learning Content

This course is taught at the University of South Bohemia by doc. Ing. Ladislav Beránek, CSc., PhDr. Miloš Prokýšek, Ph.D.

Mgr. Jakub Geyer

1 Relational and noSQL data storages
1 Datawarehouse
1 Star, Snowflake and Data Vault patterns
1 ETL, OLAP, OLTP
1 Distributed database systems
1 CAP theorem
1 Master-slave, mirroring, sharding
1 NoSQL database systems

1	Key-value
1	Column oriented
1	Document databases
1	Graph databases
1	Time-series databases
1	Large datasets
1	Velocity, variability, volume
1	Unstructured data
1	ELT processing, curated data
1	Stream data processing
1	Buffering
1	Distribution
1	Storing
1	Real-time processing
1	Data mining
1	Data sources and datatypes
1	Data matrix
1	Data storages
1	Level of Míry podobnosti, metody shlukové analýz
1	Basic data models
1	Linear and log-linear regression
1	Data modeling
1	Decission trees, associaton rules
1	Classificators
1	k-NN
1	naive bayes classificator
1	Data lakes
1	Distributed filesystems
1	Hadoop - family solutions

Teaching Methods

lectures, practice

Remarks

Criteria for obtaining the course credit: Defense of semestral work
Criterium for passing the exam: Oral examination with two theoretical topics

Recommended Literature

Basic Literature:

Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, Vipin Kumar. Introduction to Data Mining (2nd edition). 2018.

AID-12 Parallel Programming and Computing

Module code	AID-12
Module coordination	Dr. Milan P#edota
Course number and name	AID-12 Parallel Programming and Computing
Lecturer	Prof. Dr. Andreas Berl
Semester	2
Duration of the module	1 semester
Module frequency	annually
Course type	required course
Level	postgraduate
Semester periods per week (SWS)	3
ECTS	4
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	course assessment, written ex. 90 min.
Duration of Examination	90 min.
Weight	4/120
Language of Instruction	English

Module Objective

Students will learn the following skills:

setting of access to a parallel cluster, solving of tasks in MPI and later in OpenMP (calculation of a sum of numbers and a factorial, calculation of a definite integral, calculation of energy of a configuration of molecules including execution on multiple computers and efficiency determination. Further tasks practise numerous commands and skills. In?OpenCL the students work with large matrices and calculate the energy of molecules. Parallel sorting of data represents te final task independently soled by students.

Entrance Requirements

Learning Content

This course is taught at the University of South Bohemia by doc. RNDr. Milan Predota, Ph.D.

1. Introduction to parallel computing. What is parallel computing? Why do we need parallel computers? Parallelization strategies. Acquaintance with a simple parallel program. Commands for creating and managing a parallel environment.

2.-3. Parallelization in MPI, basic commands (MPI_Bcast, MPI_Reduce, MPI_Allreduce). Different approaches to parallelization of loops.

4. Working with arrays (MPI_Scatter, MPI_Gather).

5. Commands of the C language for working with files, input/output in parallel programs, comparison of different approaches.

6. Timing of the program, communication vs. computing time - optimalization of number of threads.

7. Point to point parallelization in MPI (MPI_Send, MPI_Recv).

8. Management of tasks, execution of parallel tasks, queue submission systems.

9. Differences in parallelization with respect to cooperation of threads during the calculation. Shared vs. distributed memory, MPI vs. OpenMP parallel environments.

10. Parallelization in OpenMP, basic commands (pragma, parallelization of loops, reduction).

11. Parallelization in OpenMP, advanced commands (pragma single/master, critical/ atomic/ordered, parallelization of sections).

12.-13. Programming on the graphic cards (GPU) using OpenCL, host + kernel, passing of arguments, setting of the number of GPU threads.

Practical exercises follow the content of lectures. Setting of access to a parallel cluster, solving of tasks in MPI and later in OpenMP (calculation of a sum of numbers and a factorial, calculation of a definite integral, calculation of energy of a configuration of molecules including execution on multiple computers and efficiency determination. Further tasks practise numerous commands and skills. In OpenCL the students work with large matrices and calculate the energy of molecules. Parallel sorting of data represents te final task independently soled by students.

Teaching Methods

Lectures, practise

Remarks

Criteria for obtaining the course credit (zápocet): Solving at least 70% of taskds solved during exercises with the help of the teacher. Independent solution of two larger individual tasks (at least 50%). Criteria for passing the exam: At least 50% knowledge of the topic contained in the two drawn questions.

Recommended Literature

Basic Literature:

E-learning materials of this course by M. Predota, https://elearning.jcu.cz/course/view.php?id=1830

Recommended Literature:

OpenCL, https://www.khronos.org/opencl/

OpenMP, https:www.//openmp.org/

Barbara Chapman, Gabriele Jost, Ruud Van Der Pas: Using OpenMP: Portable Shared Memory Parallel Programming, the MIT Press 2007

Michael J. Quinn: Parallel Programming in C with MPI and Open MP, McGraw HIll Higher Education, 2003.

Rohit Chandra, Leo Dagum, Dave Kohr, Dor Maydan: Parallel Programming in OpenMP, Morgan Kaufmann, 2000

William Gropp, Ewing Lusk and Anthony Skjellum: Using MPI - 3rd edition: Portable Parallel Programming with the Mesage Passing Interface (Scientific and Engineering Computation), The MIT Press, 2014.

Study Aids:

E-Learning materials for this course by M. Predota: https://elearning.jcu.cz/course/view.php?id=1830

AID-13 Language Course: German or Czech

Module code	AID-13
Module coordination	Tanja Mertadana
Course number and name	AID-13 Language Course: German or Czech
Lecturer	Tanja Mertadana
Semester	2
Duration of the module	1 semester
Module frequency	
Course type	required course
Level	
Semester periods per week (SWS)	2
ECTS	2
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	written ex. 60 min.
Duration of Examination	60 min.
Weight	2/120
Language of Instruction	German

Module Objective

Entrance Requirements

AID-14 Internship

Module code	AID-14
Module coordination	Prof. Dr. Andreas Berl
	Germany
Course number and name	AID-14 Internship
Lecturer	Prof. Dr. Andreas Berl
Semester	3
Duration of the module	1 semester
Module frequency	as required
Course type	required course
Level	
Semester periods per week (SWS)	0
ECTS	20
Workload	Time of attendance: 0 hours virtual learning: 600 hours Total: 600 hours
Weight	20/120
Language of Instruction	English

Module Objective

The aim of the internship is to use and verify the acquired knowledge and skills in the practical conditions of a particular corporate environment. At the same time, students will test their own ability to solve a range of practical problems in the field of applied informatics. This solution of practical problems as well as understanding the connection with the operation of the company will help anchor theoretical knowledge and clarify specific areas of interest in the field of AI, data sciences and computer systems.

Part of the course is an introductory seminar for the presentation of companies, at the end of the internship students will prepare a final report as a condition for granting credit.

Internships in individual technology companies are contractually secured.

Entrance Requirements

Remarks

Credit is given based on attendance, evaluation of the responsible employee of the company where

the internship takes place. Student's final report.

Recommended Literature

Basic literature:

Martin Gary E., Arnold B. Danzig, William F. Wright, Richard A. Flanary: School Leader Internship (2016). Routledge; 4 edition , 218 pages. ISBN: 978-1138824010

AID-15 Faculty Elective 3 (FWPM)

Module code	AID-15
Module coordination	Prof. Dr. Andreas Berl
Course number and name	AID-15 Faculty Elective 3 (FWPM)
Lecturer	Prof. Dr. Andreas Berl
Semester	3
Duration of the module	1 semester
Module frequency	each semester
Course type	compulsory course
Semester periods per week (SWS)	0
ECTS	5
Workload	Time of attendance: 0 hours Total: 0 hours
Language of Instruction	English

Module Objective

AID-16 Faculty Elective 4 (FWPM)

Module code	AID-16
Module coordination	Prof. Dr. Andreas Berl
Course number and name	AID-16 Faculty Elective 4 (FWPM)
Lecturer	Prof. Dr. Andreas Berl
Semester	3
Duration of the module	1 semester
Module frequency	each semester
Course type	required course
Semester periods per week (SWS)	0
ECTS	5
Workload	Time of attendance: 0 hours Total: 0 hours
Language of Instruction	English

Module Objective

AID-17 Advanced Topics in AI

Module code	AID-17
Module coordination	Prof. Dr. Christina Bauer
Course number and name	AID-17 Advanced Topics in AI
Lecturer	Prof. Dr. Christina Bauer
Semester	4
Duration of the module	1 semester
Module frequency	annually
Course type	required course
Level	postgraduate
Semester periods per week (SWS)	4
ECTS	5
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	written student research project
Weight	5/120
Language of Instruction	English

Module Objective

The purpose of this course is to provide students with hands-on and real-world development experience. They will have the opportunity to review some cutting-edge research papers and to then turn them in concrete software/hardware outcomes. As these projects are completed in teams, students will also have the opportunity to elaborate on their social and language skills. At the end of the term, students will present their projects at an in-house R&D fair which will be open to the public.

Content and learning goals:

- Implementing contemporary research papers from the fields of artificial intelligence, machine learning, computer vision, natural language processing and others.
- Using modern high-end hardware, such as GPUs clusters and cloud services.

- Utilizing an agile process framework such as Scrum.
- Understanding and using modern industrial software development tools such as work package trackers, code revision systems, debuggers, profilers and others.
- Presenting R&D outcomes to stakeholders at different levels, such as fellow students, faculty members and practitioners and executives.

Entrance Requirements

Learning Content

The purpose of this course is to provide students with hands-on and real-world development experience. They will have the opportunity to review some cutting-edge research papers and to then turn them in concrete software/hardware outcomes. As these projects are completed in teams, students will also have the opportunity to elaborate on their social and language skills. At the end of the term, students will present their projects at an in-house R&D fair which will be open to the public.

Content and learning goals:

- Implementing contemporary research papers from the fields of artificial intelligence, machine learning, computer vision, natural language processing and others.
- Using modern high-end hardware, such as GPUs clusters and cloud services.
- Utilizing an agile process framework such as Scrum.
- Understanding and using modern industrial software development tools such as work package trackers, code revision systems, debuggers, profilers and others.
- Presenting R&D outcomes to stakeholders at different levels, such as fellow students, faculty members and practitioners and executives.

Teaching Methods

project and seminars

Recommended Literature

Basic:

- C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
- I. Goodfellow, Y. Bengio and A. Courville, Deep Learning, MIT Press, 2016.

Study aids:

- High-end GPUs

- Cloud services
- Development boards
- Mobile robots and drones
- Hardware manuals

AID-18 Master Thesis

Module code	AID-18
Module coordination	Prof. Dr. Andreas Berl
Course number and name	AID-18 Master Thesis
Lecturer	Prof. Dr. Andreas Berl
Semester	4
Duration of the module	1 semester
Module frequency	each semester
Course type	required course
Semester periods per week (SWS)	0
ECTS	20
Workload	Time of attendance: 0 hours Total: 0 hours
Language of Instruction	English

Module Objective

AID-19 Master Seminar

Module code	AID-19
Module coordination	Prof. Dr. Andreas Berl
Course number and name	AID-19 Master Seminar
Lecturer	Prof. Dr. Andreas Berl
Semester	4
Duration of the module	1 semester
Module frequency	each semester
Course type	required course
Level	postgraduate
Semester periods per week (SWS)	0
ECTS	5
Workload	Time of attendance: 0 hours Total: 0 hours
Type of Examination	oral examination
Weight	5/120
Language of Instruction	English

Module Objective

Elaboration of thesis for the final work

Entrance Requirements