Study plan

Name of study plan: Informatics (doctoral)

Faculty/Institute/Others:
Department:
Branch of study guaranteed by the department: Welcome page
Garantor of the study branch:
Program of study: Informatics
Type of study: Doctoral
Required credits: 0
Elective courses credits: 24
Sum of credits in the plan: 24
Note on the plan:

Name of the block: Compulsory elective courses
Minimal number of credits of the block: 0
The role of the block: PV

Code of the group: PI-VSE
Name of the group: All doctoral courses
Requirement credits in the group:
Requirement courses in the group:
Credits in the group: 0
Note on the group: All FIT doctoral courses are included in this group

Code	**Name of the course / Name of the group of courses** ***(in case of groups of courses the list of codes of their members)*** *Tutors, **authors** and guarantors (gar.)*	Completion	Credits	Scope	Semester	Role
PI-AWR.1	**Academic Writing** *Petr Kroha **Petr Kroha** Petr Kroha (Gar.)*	ZK	0	2C	Z	PV
PI-APA	**Advanced Program Analysis** *Jan Vitek **Jan Vitek** Jan Vitek (Gar.)*	ZK	4	3C	Z	PV
PI-ADH	**Algorithms and Data Structures for HPC** *Ivan Šimeček **Ivan Šimeček** Ivan Šimeček (Gar.)*	ZK	4	3C	Z	PV
PI-AKD	**Data Compression Algorithms** *Jan Holub **Jan Holub** Jan Holub (Gar.)*	ZK	4	3C	L	PV
PI-AVG	**Computational genomics algorithms** *Jan Holub **Jan Holub** Jan Holub (Gar.)*	ZK	4	2P+1C	L	PV
PI-AJMIN	**English Language - Discussion on the Dissertation Thesis** *Štěpán Starosta **Štěpán Starosta** Pavel Tvrdík (Gar.)*	ZK	0		Z,L	PV
PI-ANM	**Applied Numerical Mathematics** *Róbert Lórencz **Róbert Lórencz** Róbert Lórencz (Gar.)*	ZK	4	3C	Z,L	PV
PI-ARB	**Arbology** *Jan Janoušek, Bořivoj Melichar **Jan Janoušek** Jan Janoušek (Gar.)*	ZK	4	3C	Z,L	PV
PI-ASP	**Architecture of Symbolic Computers** *Josef Kolář **Josef Kolář** Josef Kolář (Gar.)*	ZK	4	3C	Z,L	PV
PI-CFR	**Computer Assisted Formal Reasoning**	ZK	4	3C	Z,L	PV
PI-EXA	**Experimental algorithmics** *Jan Schmidt **Jan Schmidt** Jan Schmidt (Gar.)*	ZK	4	2P+1C	L	PV
PI-IRT	**Information retrieval** *Petr Kroha **Petr Kroha** Petr Kroha (Gar.)*	ZK	4	3C	L	PV
PI-KP	**Communication protocols** *Jan Janeček **Jan Janeček** Jan Janeček (Gar.)*	ZK	4	3C	L	PV
PI-BCM	**Conceptual Modelling of Behaviour** *Robert Pergl **Robert Pergl** Robert Pergl (Gar.)*	ZK	4	3C	Z,L	PV
PI-KIK	**Quantum Information and Quantum Cryptography**	ZK	4	3C	L	PV
PI-NSV	**Neural Networks and Computational Intelligence** *Pavel Surynek **Pavel Surynek** Pavel Surynek (Gar.)*	ZK	4	3C	L	PV
PI-PRO	**Planning in Robotics** *Pavel Surynek **Pavel Surynek** Pavel Surynek (Gar.)*	ZK	4	3C	L	PV
PI-PPA	**Advanced Parallel Algorithms** *Pavel Tvrdík **Pavel Tvrdík** Pavel Tvrdík (Gar.)*	ZK	4	3C	Z	PV

PI-ROZ	**Advanced Pattern Recognition** *Michal Haindl **Michal Haindl** Michal Haindl (Gar.)*	ZK	4	3C	L	PV
PI-PSC	**Programmable Circuits and SoC** *Hana Kubátová **Hana Kubátová** Hana Kubátová (Gar.)*	ZK	4	2P+1C	Z,L	PV
PI-FME.1	**Seminar on Formal Specifications** *Karel Richta **Karel Richta** Karel Richta (Gar.)*	ZK	4	3C	Z,L	PV
PI-SCN	**Seminars on Digital Design** *Petr Fišer **Petr Fišer** Petr Fišer (Gar.)*	ZK	4	2P+1C	Z,L	PV
PI-SWI	**Software Engineering** *Petr Kroha **Petr Kroha** Petr Kroha (Gar.)*	ZK	4	3C	L	PV
PI-SPL	**Satisfiability and Planning** *Pavel Surynek **Pavel Surynek** Pavel Surynek (Gar.)*	ZK	4	3C	Z	PV
PI-STR	**Stringology** *Jan Holub **Jan Holub** Jan Holub (Gar.)*	ZK	4	3C	L	PV
PI-SCM	**Structural Conceptual Modelling** *Robert Pergl **Robert Pergl** Robert Pergl (Gar.)*	ZK	4	3C	Z,L	PV
PI-TGR	**Graph Theory** *Tomáš Valla, Ondřej Suchý **Ondřej Suchý** Ondřej Suchý (Gar.)*	ZK	4	2P+1C	Z	PV
PI-TMN	**Text Mining** *Petr Kroha **Petr Kroha** Petr Kroha (Gar.)*	ZK	4	3C	Z	PV
PI-TPL	**Type Systems for Programming Languages** *Jan Vitek **Jan Vitek** Jan Vitek (Gar.)*	ZK	4	3C	L	PV
PI-ESC	**Embeded SeCurity** *Róbert Lórencz **Róbert Lórencz** Róbert Lórencz (Gar.)*	ZK	4	3C	Z	PV
PI-VAP	**Advanced Computer Architectures** *Pavel Tvrdík **Pavel Tvrdík** Pavel Tvrdík (Gar.)*	ZK	4	3C	L	PV

Characteristics of the courses of this group of Study Plan: Code=PI-VSE Name=All doctoral courses

PI-AWR.1	Academic Writing	ZK	0
PI-APA	Advanced Program Analysis	ZK	4

In the past decade, there have been great advances in the development of automated tools that help programmers find various kinds of quality problems in their code. This includes tools for finding bugs and security vulnerabilities, test generation, fault detection and localization, etc. Many of these tools rely on program analysis to compute an approximation of a program's behavior. In this special topics course, we will study key publications in which static and dynamic program analysis algorithms are used to detect bugs and security vulnerabilities in programs, and how these algorithms are used in other tools that support programmers. Both theoretical properties and practical effectiveness of program analysis algorithms will be studied.

PI-ADH	Algorithms and Data Structures for HPC	ZK	4

The most computation intensive tasks (or memory complex tasks) are solved by large HPC systems. Seven so called "dwarfs" were identified as typical tasks computed by HPC systems. In this subject these "dwarfs" are described (including their variants). Also typical algortihms and advanced data structures for solution will be discussed.

PI-AKD	Data Compression Algorithms	ZK	4

Goals: After completing the course, the students will be able to design special data compression methods or their compositions customized for a given problem. The efficiency of designed solution is evaluated by many parametes not only by the compression ratio. Added value: The students will learn to evaluate advantages and disadvantages of data compression methods and their classes. They also learn to construct cascade methods in order to achieve desired properties of the resulting data compression system.

PI-AVG	Computational genomics algorithms	ZK	4

The course deals with efficient algorithms for various tasks in bioinformatics. One fo such task is an alignment of two or more sequences. Other topic covers algorithms for individual phases of genome assembly. The course also presents compressed data structures for storing and indexing genomes and very fast pattern matching in them. Algorithms for efficient analysis and comparison of genomes.

PI-AJMIN	English Language - Discussion on the Dissertation Thesis	ZK	0

Exam of english language in form of defense of professional study in English. Doctoral student defends his professional work drafted and presented in English before the state committee. Ph.D. student is evaluated in presentation skills, mastery of the language in continuous speech and language skills quickly and correctly respond during the debate.

PI-ANM	Applied Numerical Mathematics	ZK	4
PI-ARB	Arbology	ZK	4

Introduction to tree problems and their effective algorithmic solutions. Algorithms presented on the basis of tree and pushdown automata as models of computation. Practical algorithms used in compiler construction and XML processing are discussed in details.

PI-ASP	Architecture of Symbolic Computers	ZK	4

The course offers a deeper understanding of working principles and internal structure of functional and logical programming systems. A concrete insight is acquired concerning advantages and limits when using such systems as well as concerning specific issues related to their implementation as compared with common imperative programming systems.

PI-CFR	Computer Assisted Formal Reasoning	ZK	4

The goal of this course is to provide the student with the ability to - completely formalize research problems in the field of their Ph.D. study, to - prove the correctness of solutions to such problems, and to - prepare the resulting proofs for publication, while supporting this process using state-of-the-art software tools. The course will take the form of consultations. The teacher will work with the student on concrete research problems from the student's field of research.

PI-EXA	Experimental algorithmics	ZK	4

The course explains experimental evaluation of algorithms and programs, its significance for scientific work and interpretation of its results. Standards of relevance and confidence established in experimental science are transferred to this field.

PI-IRT	Information retrieval	ZK	4
PI-KP	Communication protocols	ZK	4

Students will learn the trends of modern communication protocols development, architectures of selected distributed systems, and formal tools for their specification, modeling and verification.

PI-BCM	Conceptual Modelling of Behaviour	ZK	4

The subject is focused on methodology of conceptual modeling of behavior in the context of business engineering and software engineering. In the course we focus on theoretical and practical aspects of significant approaches to ontological behaviour modeling, such as UFO-B, BORM and DEMO and their application in enterprise engineering, software engineering and ontological analysis of complex domains. Different levels of description of social, socio-technical and technical systems behaviour and their context are discussed.

PI-KIK	Quantum Information and Quantum Cryptography	ZK	4

The module deals with the processing of quantum information, quantum computing, quantum communication from the viewpoint of security, and quantum cryptography. The students learn how specific laws of quantum physics and quantum properties of microscopic world can achieve the objectives classically intractable, or solve some problems more efficiently.

<table>
<tr><td>PI-NSV</td><td>Neural Networks and Computational Intelligence</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Theoretical foundations of neural networks with a focus on advanced paradigms and the use of neural networks as a model for data analysis and data mining. Networks with dynamically generated topology during learning developed on the principles of inductive modeling. Evolutionary techniques and nature-inspired optimization. Principles of machine learning, deep neural networks and deep learning.</td></tr>
<tr><td>PI-PRO</td><td>Planning in Robotics</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The course covers theoretical aspects of planning in robotics from an abstract level known from classical planning to motion planning directly executable on robotic hardware. Abstract symbolic planning and robotics are linked together in this subject, so we will show how to create symbolic plans and refine them through geometric motion planning to the level of control of robotic hardware. The focus will be on (but not limited to) algorithms for creating classical plans by forward state search, planning with time and resources, planning under uncertainty, probabilistic planning. The course will smoothly continue with specific robotic aspects of planning, i.e. motion planning and reflecting the true plan execution in contrast to the ideal abstract plan, geometric representations of working and configuration spaces, combinatorial and probabilistic methods for pathfinding in configuration spaces, location and mapping techniques, motion planning with differential constraints. Planning and coordinating multiple robots will be important aspect that we will focus on. The course is focused on algorithmic techniques for generating plans, not on execution of plans by robots. It is therefore recommended to further verify the theoretical knowledge in practice in some of the robotic simulators or on real robots in the faculty laboratory.</td></tr>
<tr><td>PI-PPA</td><td>Advanced Parallel Algorithms</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The students learn complex parallel algorithms and techniques to assess their correctness efficiency and optimality.</td></tr>
<tr><td>PI-ROZ</td><td>Advanced Pattern Recognition</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Lectures follow up the fundamental course Pattern Recognition 1 (MI-ROZ). The fundamentals of statistical pattern recognition based on multidimensional models, contextual classification and recent pattern recognition applications in the area of machine perception will be explained in the lectures.</td></tr>
<tr><td>PI-PSC</td><td>Programmable Circuits and SoC</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Students will obtain the knowledge and practical skills in the state-of-the-art SoC and NoC design methods.</td></tr>
<tr><td>PI-FME.1</td><td>Seminar on Formal Specifications</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Students learn how to evaluate pros and cons of formal specifications and how to work with tools supporting such formalisms, and also how to design and evaluate specification prototypes.</td></tr>
<tr><td>PI-SCN</td><td>Seminars on Digital Design</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">This subject deals with problems of realization and implementation of digital circuits - both combinational and sequential. Basic means of description of digital circuits and basic logic synthesis and optimization algorithms are described. Basics of EDA (Electronic Design Automation) systems are given, together with combinatorial problems emerging in EDA.</td></tr>
<tr><td>PI-SWI</td><td>Software Engineering</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The course assumes knowledges discussed at FIT CTU courses Software Engineering I. and Software Engineering II. including projects working experiences. A prerequisite for understanding the lectures is a good knowledge of object-oriented programming and modeling. The knowledges will be extended to modern methods, eg. Adaptive Programming, Aspect-oriented programming. Some of the modern concepts are explained in more detail or perspective and in context: mainly the use of and respect for the principles of software engineering for creating requirements , modeling and design of information systems.</td></tr>
<tr><td>PI-SPL</td><td>Satisfiability and Planning</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The course offers a modern perspective on solving problems in artificial intelligence through satisfiability in logic (SAT) and finite domain constraint satisfaction problems (CSP). Satisfiability in logic, especially propositional logic, currently represents one of the most sophisticated approaches to state space search. We will discuss advanced techniques used in systematic solvers based on CDCL (conflict-driven clause learning), techniques for encoding pseudo-Boolean and cardinality constraints, symmetry-breaking techniques, satisfiability in first order logic theories, SAT modulo theories (SMT), and tractable cases where satisfiability has polynomial time complexity will also be discussed. We will emphasize the use of logic and satisfiability in the pivotal topic of symbolic artificial intelligence, namely in classical planning. In a closely related area of constraint satisfaction, we will focus on constraint propagation techniques, algorithms for maintaining consistencies such as arc or path consistency, filtering algorithms for global cardinality constraints, and problem modeling in CSP, especially planning. We give a unified view of CSP and SAT with strong emphasis on explanation of algorithmic principles.</td></tr>
<tr><td>PI-STR</td><td>Stringology</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Algorithms for processing and searching in text. Algorithms presented on the basis of finite automata as models of computation. Principles of processing compressed text and parallel algorithms</td></tr>
<tr><td>PI-SCM</td><td>Structural Conceptual Modelling</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The course is focused on the methodology of structural conceptual modelling in the context of information engineering and software engineering. In this course we focus on theoretical and practical aspects of significant approaches to modelling ontological structures such as modal logic, descriptive logic and their application in languages such as OntoUML, Alloy and OWL. The focus is on model-driven engineering and ontological analysis of complex domains. Methods and tools of verification, validation and simulation of structural ontological models, model transformation and code generation are discussed.</td></tr>
<tr><td>PI-TGR</td><td>Graph Theory</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Attention will be paid both to structural issues and to questions of algorithmization and computational complexity of basic optimization problems restricted to special graph classes, aiming at determining the boundary between polynomially solvable and NP-hard variants of the problems.</td></tr>
<tr><td>PI-TMN</td><td>Text Mining</td><td>ZK</td><td>4</td></tr>
<tr><td>PI-TPL</td><td>Type Systems for Programming Languages</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">A type system is a static method for imposing constraints on legal programs in order to guarantee their safe execution, which would prevent some class of execution errors prior to running the program, whilst a semantics specifies what the program will do when executed. Type systems in languages like Java and C# provide a lightweight tool for identifying syntactic errors as well as erroneous uses of data and illegal memory accesses. More sophisticated type systems can be used to guarantee a multitude of other properties, including reasoning about memory management and resource usage, confidentiality and integrity of data, atomicity in concurrent programs, safe execution of untrusted code. This course gives an introduction to the main ideas and methodologies behind type systems and semantics, and a practical exploration of typed features for commonly used statically typed programming languages. This course will be assessed through written assignments and a final project that involves programming.</td></tr>
<tr><td>PI-ESC</td><td>Embeded SeCurity</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Familiarization of students with the theoretical and practical aspects of embedded security. Design methods of hardware cryptographic primitives for embedded systems. Understanding of the origin of vulnerabilities when designing digital circuits for embedded systems. Methods of elimination of these vulnerabilities.</td></tr>
<tr><td>PI-VAP</td><td>Advanced Computer Architectures</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Students will learn the mechanisms for multilevel branch prediction, speculative instruction execution, and speculative data prefetching techniques in ILP processors. The second part is on memory hierarchy systems, memory consistency models, and memory coherence protocols in parallel computer systems with virtual shared distributed memory. The third part is devoted to synchronization mechanisms in parallel systems with distributed memory.</td></tr>
</table>

List of courses of this pass:

<table>
<tr><th>Code</th><th>Name of the course</th><th>Completion</th><th>Credits</th></tr>
<tr><td>PI-ADH</td><td>Algorithms and Data Structures for HPC</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The most computation intensive tasks (or memory complex tasks) are solved by large HPC systems. Seven so called "dwarfs" were identified as typical tasks computed by HPC systems. In this subject these "dwarfs" are described (including their variants). Also typical algortihms and advanced data structures for solution will be discussed.</td></tr>
<tr><td>PI-AJMIN</td><td>English Language - Discussion on the Dissertation Thesis</td><td>ZK</td><td>0</td></tr>
<tr><td colspan="4">Exam of english language in form of defense of professional study in English. Doctoral student defends his professional work drafted and presented in English before the state committee. Ph.D. student is evaluated in presentation skills, mastery of the language in continuous speech and language skills quickly and correctly respond during the debate.</td></tr>
<tr><td>PI-AKD</td><td>Data Compression Algorithms</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Goals: After completing the course, the students will be able to design special data compression methods or their compositions customized for a given problem. The efficiency of designed solution is evaluated by many parametes not only by the compression ratio. Added value: The students will learn to evaluate advantages and disadvantages of data compression methods and their classes. They also learn to construct cascade methods in order to achieve desired properties of the resulting data compression system.</td></tr>
<tr><td>PI-ANM</td><td>Applied Numerical Mathematics</td><td>ZK</td><td>4</td></tr>
<tr><td>PI-APA</td><td>Advanced Program Analysis</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">In the past decade, there have been great advances in the development of automated tools that help programmers find various kinds of quality problems in their code. This includes tools for finding bugs and security vulnerabilities, test generation, fault detection and localization, etc. Many of these tools rely on program analysis to compute an approximation of a program's behavior. In this special topics course, we will study key publications in which static and dynamic program analysis algorithms are used to detect bugs and security vulnerabilities in programs, and how these algorithms are used in other tools that support programmers. Both theoretical properties and practical effectiveness of program analysis algorithms will be studied.</td></tr>
<tr><td>PI-ARB</td><td>Arbology</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Introduction to tree problems and their effective algorithmic solutions. Algorithms presented on the basis of tree and pushdown automata as models of computation. Practical algorithms used in compiler construction and XML processing are discussed in details.</td></tr>
<tr><td>PI-ASP</td><td>Architecture of Symbolic Computers</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The course offers a deeper understanding of working principles and internal structure of functional and logical programming systems. A concrete insight is acquired concerning advantages and limits when using such systems as well as concerning specific issues related to their implementation as compared with common imperative programming systems.</td></tr>
<tr><td>PI-AVG</td><td>Computational genomics algorithms</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The course deals with efficient algorithms for various tasks in bioinformatics. One fo such task is an alignment of two or more sequences. Other topic covers algorithms for individual phases of genome assembly. The course also presents compressed data structures for storing and indexing genomes and very fast pattern matching in them. Algorithms for efficient analysis and comparison of genomes.</td></tr>
<tr><td>PI-AWR.1</td><td>Academic Writing</td><td>ZK</td><td>0</td></tr>
<tr><td>PI-BCM</td><td>Conceptual Modelling of Behaviour</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The subject is focused on methodology of conceptual modeling of behavior in the context of business engineering and software engineering. In the course we focus on theoretical and practical aspects of significant approaches to ontological behaviour modeling, such as UFO-B, BORM and DEMO and their application in enterprise engineering, software engineering and ontological analysis of complex domains. Different levels of description of social, socio-technical and technical systems behaviour and their context are discussed.</td></tr>
<tr><td>PI-CFR</td><td>Computer Assisted Formal Reasoning</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The goal of this course is to provide the student with the ability to - completely formalize research problems in the field of their Ph.D. study, to - prove the correctness of solutions to such problems, and to - prepare the resulting proofs for publication, while supporting this process using state-of-the-art software tools. The course will take the form of consultations. The teacher will work with the student on concrete research problems from the student's field of research.</td></tr>
<tr><td>PI-ESC</td><td>Embeded SeCurity</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Familiarization of students with the theoretical and practical aspects of embedded security. Design methods of hardware cryptographic primitives for embedded systems. Understanding of the origin of vulnerabilities when designing digital circuits for embedded systems. Methods of elimination of these vulnerabilities.</td></tr>
<tr><td>PI-EXA</td><td>Experimental algorithmics</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The course explains experimental evaluation of algorithms and programs, its significance for scientific work and interpretation of its results. Standards of relevance and confidence established in experimental science are transferred to this field.</td></tr>
<tr><td>PI-FME.1</td><td>Seminar on Formal Specifications</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Students learn how to evaluate pros and cons of formal specifications and how to work with tools supporting such formalisms, and also how to design and evaluate specification prototypes.</td></tr>
<tr><td>PI-IRT</td><td>Information retrieval</td><td>ZK</td><td>4</td></tr>
<tr><td>PI-KIK</td><td>Quantum Information and Quantum Cryptography</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The module deals with the processing of quantum information, quantum computing, quantum communication from the viewpoint of security, and quantum cryptography. The students learn how specific laws of quantum physics and quantum properties of microscopic world can achieve the objectives classically intractable, or solve some problems more efficiently.</td></tr>
<tr><td>PI-KP</td><td>Communication protocols</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Students will learn the trends of modern communication protocols development, architectures of selected distributed systems, and formal tools for their specification, modeling and verification.</td></tr>
<tr><td>PI-NSV</td><td>Neural Networks and Computational Intelligence</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">Theoretical foundations of neural networks with a focus on advanced paradigms and the use of neural networks as a model for data analysis and data mining. Networks with dynamically generated topology during learning developed on the principles of inductive modeling. Evolutionary techniques and nature-inspired optimization. Principles of machine learning, deep neural networks and deep learning.</td></tr>
<tr><td>PI-PPA</td><td>Advanced Parallel Algorithms</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The students learn complex parallel algorithms and techniques to assess their correctness efficiency and optimality.</td></tr>
<tr><td>PI-PRO</td><td>Planning in Robotics</td><td>ZK</td><td>4</td></tr>
<tr><td colspan="4">The course covers theoretical aspects of planning in robotics from an abstract level known from classical planning to motion planning directly executable on robotic hardware. Abstract symbolic planning and robotics are linked together in this subject, so we will show how to create symbolic plans and refine them through geometric motion planning to the level of control of robotic hardware. The focus will be on (but not limited to) algorithms for creating classical plans by forward state search, planning with time and resources, planning under uncertainty, probabilistic planning. The course will smoothly continue with specific robotic aspects of planning, i.e. motion planning and reflecting the true plan execution in contrast to the ideal</td></tr>
</table>

abstract plan, geometric representations of working and configuration spaces, combinatorial and probabilistic methods for pathfinding in configuration spaces, location and mapping techniques, motion planning with differential constraints. Planning and coordinating multiple robots will be important aspect that we will focus on. The course is focused on algorithmic techniques for generating plans, not on execution of plans by robots. It is therefore recommended to further verify the theoretical knowledge in practice in some of the robotic simulators or on real robots in the faculty laboratory.			
PI-PSC	Programmable Circuits and SoC	ZK	4
Students will obtain the knowledge and practical skills in the state-of-the-art SoC and NoC design methods.			
PI-ROZ	Advanced Pattern Recognition	ZK	4
Lectures follow up the fundamental course Pattern Recognition 1 (MI-ROZ). The fundamentals of statistical pattern recognition based on multidimensional models, contextual classification and recent pattern recognition applications in the area of machine perception will be explained in the lectures.			
PI-SCM	Structural Conceptual Modelling	ZK	4
The course is focused on the methodology of structural conceptual modelling in the context of information engineering and software engineering. In this course we focus on theoretical and practical aspects of significant approaches to modelling ontological structures such as modal logic, descriptive logic and their application in languages such as OntoUML, Alloy and OWL. The focus is on model-driven engineering and ontological analysis of complex domains. Methods and tools of verification, validation and simulation of structural ontological models, model transformation and code generation are discussed.			
PI-SCN	Seminars on Digital Design	ZK	4
This subject deals with problems of realization and implementation of digital circuits - both combinational and sequential. Basic means of description of digital circuits and basic logic synthesis and optimization algorithms are described. Basics of EDA (Electronic Design Automation) systems are given, together with combinatorial problems emerging in EDA.			
PI-SPL	Satisfiability and Planning	ZK	4
The course offers a modern perspective on solving problems in artificial intelligence through satisfiability in logic (SAT) and finite domain constraint satisfaction problems (CSP). Satisfiability in logic, especially propositional logic, currently represents one of the most sophisticated approaches to state space search. We will discuss advanced techniques used in systematic solvers based on CDCL (conflict-driven clause learning), techniques for encoding pseudo-Boolean and cardinality constraints, symmetry-breaking techniques, satisfiability in first order logic theories, SAT modulo theories (SMT), and tractable cases where satisfiability has polynomial time complexity will also be discussed. We will emphasize the use of logic and satisfiability in the pivotal topic of symbolic artificial intelligence, namely in classical planning. In a closely related area of constraint satisfaction, we will focus on constraint propagation techniques, algorithms for maintaining consistencies such as arc or path consistency, filtering algorithms for global cardinality constraints, and problem modeling in CSP, especially planning. We give a unified view of CSP and SAT with strong emphasis on explanation of algorithmic principles.			
PI-STR	Stringology	ZK	4
Algorithms for processing and searching in text. Algorithms presented on the basis of finite automata as models of computation. Principles of processing compressed text and parallel algorithms			
PI-SWI	Software Engineering	ZK	4
The course assumes knowledges discussed at FIT CTU courses Software Engineering I. and Software Engineering II. including projects working experiences. A prerequisite for understanding the lectures is a good knowledge of object-oriented programming and modeling. The knowledges will be extended to modern methods, eg. Adaptive Programming, Aspect-oriented programming. Some of the modern concepts are explained in more detail or perspective and in context: mainly the use of and respect for the principles of software engineering for creating requirements , modeling and design of information systems.			
PI-TGR	Graph Theory	ZK	4
Attention will be paid both to structural issues and to questions of algorithmization and computational complexity of basic optimization problems restricted to special graph classes, aiming at determining the boundary between polynomially solvable and NP-hard variants of the problems.			
PI-TMN	Text Mining	ZK	4
PI-TPL	Type Systems for Programming Languages	ZK	4
A type system is a static method for imposing constraints on legal programs in order to guarantee their safe execution, which would prevent some class of execution errors prior to running the program, whilst a semantics specifies what the program will do when executed. Type systems in languages like Java and C# provide a lightweight tool for identifying syntactic errors as well as erroneous uses of data and illegal memory accesses. More sophisticated type systems can be used to guarantee a multitude of other properties, including reasoning about memory management and resource usage, confidentiality and integrity of data, atomicity in concurrent programs, safe execution of untrusted code. This course gives an introduction to the main ideas and methodologies behind type systems and semantics, and a practical exploration of typed features for commonly used statically typed programming languages. This course will be assessed through written assignments and a final project that involves programming.			
PI-VAP	Advanced Computer Architectures	ZK	4
Students will learn the mechanisms for multilevel branch prediction, speculative instruction execution, and speculative data prefetching techniques in ILP processors. The second part is on memory hierarchy systems, memory consistency models, and memory coherence protocols in parallel computer systems with virtual shared distributed memory. The third part is devoted to synchronization mechanisms in parallel systems with distributed memory.			

For updated information see http://bilakniha.cvut.cz/en/FF.html
Generated: day 2024-08-14, time 18:12.