CONTENT

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ AOS1/15	**Course name:** Administration of OS
Course type, scope and the method: **Course type:** Practice **Recommended course-load (hours):** **Per week:** 2 **Per study period:** 28 **Course method:** present	
Number of ECTS credits: 2	
Recommended semester/trimester of the course:	
Course level: I., II., N	
Prerequisities:	
Conditions for course completion: The condition for passing the course is successful realization of a project focused on the network services configuration.	
Learning outcomes: The result of the education is an understanding of the theoretical and practical background of Windows and Linux operating systems and selected network services.	
Brief outline of the course: 1. Management of Linux operating system (basic system tools for troubleshooting, system startup, network configuration), 2. File systems (general view), 3. File systems (RAID, LVM), 4. Web hosting services I. (basic concept, APACHE), 5. Web hosting services II. (SQL, HTTPS, security, NGINX), 6. File services I. (SAMBA, NFS), 7. File services II. (FTP), 8. Management of local computer network I. (routing, DHCP), 9. Management of local computer network II. (firewall), 10. VPN, 11. SSH and Proxy, 12. Kernel of the Linux operating system, 13. Administration of the Windows operating system.	
Recommended literature: 1. LPIC-1 Exam 102. LPI [online]. Canada: The Linux Professional Institute, 2021 [cit. 2021-9-22]. Dostupné z: https://learning.lpi.org/en/learning-materials/102-500/, 2. Linux - Dokumentační projekt [online]. 4. Praha: Computer Press, 2007 [cit. 2021-9-22]. Dostupné z: https://i.iinfo.cz/files/root/k/LDP_4.pdf, 3. The LPIC2 Exam Prep [online]. Sue B.V. - Open Sourced, 2021 [cit. 2021-9-26]. Dostupné z: https://lpic2book.github.io/src/	
Course language: Slovak or English	
Notes: Content prerequisites: understanding of fundamental concepts of operating systems, computer networks, basic skill in Linux shell (e.g. bash) and Powershell.	

Course assessment
Total number of assessed students: 35

A	B	C	D	E	FX
60.0	20.0	11.43	0.0	8.57	0.0

Provides: doc. RNDr. JUDr. Pavol Sokol, PhD., RNDr. Tomáš Bajtoš

Date of last modification: 26.09.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ PPPy/18	**Course name:** Advanced programming in Python

Course type, scope and the method:
Course type: Practice
Recommended course-load (hours):
Per week: 2 **Per study period:** 28
Course method: present

Number of ECTS credits: 2

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities: ÚINF/PAZ1a/15

Conditions for course completion:
At least 50 % of the marks in the continuous assessment
A minimum of 50 % marks in the mid-term and end-of-semester practical tests
or
The final project - 100%

Learning outcomes:
Implement solutions to selected problems in Python using available modules. Use and implement non-trivial algorithms to solve selected problems. Use an object-oriented approach to problem solving. Program in Python in an object-oriented manner using Python specifics. Test programs. Implement parallel computing.

Brief outline of the course:
1. Introduction to the environment, basic features of Python, simple and structured data types.
2. Input, output, function definition, lambda function, generator notation, function as parameter, string formatting.
3. Control structures, iterating over data structures, context manager.
4. Exception handling and exception raising. Philosophy of exceptions in Python.
5. Working with files. Serialization and deserialization of data - json and pickle protocol. Text and binary files. Manipulation with files. Open data.
6. Object-oriented programming 1. Design of custom classes, special methods, properties, philosophy of accessing methods and attributes.
7. Object-oriented programming 2. Comparison and differences with Java. Multiple inheritance.
8. Method overloading. Static methods, abstract classes, data class.
9. Decorators, memoization, modules, packages.
10. Code validation (debugging), testing (doctest, unittest), test-driven development.
11. Parallel computing, processes, process triggering and inter-process communication (shared variable, pipe, queue).
12. Graphical program design and implementation.

Recommended literature:
PILGRIM, Mark. Dive into Python 3. 2. United States of America: Apress, 2004. ISBN 978-1430224150. Dostupné také z: https://diveintopython3.net/

SHIPMAN, John W. Tkinter 8.5 reference: a GUI for Python. Socorro, NM 87801: New Mexico Tech Computer Center, 2013. Dostupné také z: https://anzeljg.github.io/rin2/book2/2405/docs/tkinter/tkinter.pdf
LOTT, Steven F. Mastering Object-oriented Python. Birmingham B3 2PB, UK: Packt Publishing, 2014. ISBN 978-1-78328-097-1.

Course language:
Slovak language, knowledge of English language is only required to read documentation of Python.

Notes:

Course assessment
Total number of assessed students: 65

A	B	C	D	E	FX
7.69	13.85	18.46	18.46	24.62	16.92

Provides: PaedDr. Ján Guniš, PhD.

Date of last modification: 10.02.2022

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ ASU1/15	**Course name:** Algorithms and data structures

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 2 / 1 **Per study period:** 28 / 14
Course method: present

Number of ECTS credits: 4

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities: ÚINF/PAZ1a/15 and ÚINF/PAZ1b/15

Conditions for course completion:
Practice activities, homeworks and midterm exam.
Final examination consisting of practice and theoretical test.

Learning outcomes:
Understand and learn algorithmic paradigms and data structures. Analyse time complexity of these algorithms.

Brief outline of the course:
Algorithms' time and space asymptotic complexity. Main Theorem. Amortized complexity.
Brute Force. Backtrack. Divide and Conquer. Dynamic programming. Comparison and non-comparison sort algorithms. Sweep line algorithms. Graph Theory Algorithms.
Data structures – queue, stack, priority queue, heap, prefix sum, binary search trees, interval trees, union & find, trie.

Recommended literature:
1, Laaksonen A.: Guide to Competitive Programming: Learning and Improving Algorithms Through Contests (Undergraduate Topics in Computer Science), Springer, 2017, ISBN 978-3319725468
2, Forišek M., Steinová M.: Explaining Algorithms Using Metaphors. Springer Briefs in Computer Science, Springer (2013), ISBN 978-1-4471-5018-3
3, R. Sedgewick, K. Wayne: Algorithms (4th Edition), Addison-Wesley Professional, 2011, ISBN 978-0321573513, http://algs4.cs.princeton.edu/home/
4, Open Data Structures: http://opendatastructures.org/

Course language:
Slovak or english

Notes:
Content prerequisities:
- programming skills in some programming language (Python/Java/C++/...)
- mathematics:
-- computing with polynomials, logarithmic and exponential functions
-- computing limits of sequences, L'Hospital rule

Course assessment
Total number of assessed students: 184

A	B	C	D	E	FX
13.59	4.35	16.85	25.0	36.96	3.26

Provides: RNDr. Rastislav Krivoš-Belluš, PhD.

Date of last modification: 08.01.2022

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ APS1/15	**Course name:** Applied probability and statistics

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 2 / 2 **Per study period:** 28 / 28
Course method: present

Number of ECTS credits: 5

Recommended semester/trimester of the course:

Course level: I., II., N

Prerequisities: ÚMV/FRPb/19 or ÚMV/MAN2c/22 or ÚMV/MTIb/21 or ÚMV/MTI4b/22 or ÚMV/MTFb/22

Conditions for course completion:
Demonstration of adequate mastery of the content standard of the subject in the ongoing and final evaluation, the ability to formulate a problem in the acquired terminology and solve it within a project.
Written works during the semester, project.
Written and oral exam.

Learning outcomes:
After completing the course, the student is able to apply the acquired concepts and techniques of probability theory and mathematical statistics in formulating hypotheses within the considered models and analysis of data dependencies, and use the appropriate software.

Brief outline of the course:
1) Random event, probability and conditional probability.
2) Probability distribution laws.
3) Characteristics of position, variability and dependence.
4) Basic discrete and continuous distributions.
5) The law of large numbers and the central limit theorem.
6) Random sample. Initial analytical and geometric analysis of data.
7) Quantiles, basic distributions and basic theorem of mathematical statistics.
8) Theory of estimates, method of moments and maximum likelihood. Hypothesis testing.
9) Tests on distribution parameters and goodness-of-fit tests.
10) Modeling of dependencies and noise. Least squares method and smoothing.
11) Polynomial regression models.
12) Pseudorandom quantities and Monte Carlo methods.

Recommended literature:
- Cs. Török: Úvod do teórie pravdepodobnosti a matematickej štatistiky, Košice, 1992
- M.R.Spiegel, J.J.Schiller, R.A.Srinivasan, Probability and Statistics, McGraw Hill, 2009
- J. Maindonald, W.J. Braun, Data Analysis and Graphics Using R – an Example-Based Approach, CAMBRIDGE UNIVERSITY PRESS, 2010

Course language:
Slovak or english

Notes:
Face to face or online teaching.
Content prerequisites:
the basics of differential, integral and matrix calculus

Course assessment
Total number of assessed students: 90

A	B	C	D	E	FX
16.67	15.56	24.44	12.22	30.0	1.11

Provides: doc. RNDr. Csaba Török, CSc.

Date of last modification: 23.11.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ UUI1/15	**Course name:** Artificial Intelligence and Cognitive Science

Course type, scope and the method:
Course type: Lecture
Recommended course-load (hours):
Per week: 2 **Per study period:** 28
Course method: present

Number of ECTS credits: 3

Recommended semester/trimester of the course:

Course level: II., N

Prerequisities:

Conditions for course completion:
Home work and written tests.
Final exam - written or oral.

Learning outcomes:
The goal of the course is to provide an overview of the extensive field of artificial intelligence and cognitive science. The student can opt to study individually a selected topic from the literature.

Brief outline of the course:
1. Definition and goals of Artificial intelligence and Cognitive Science. Natural intelligence. Intlligence of a machine vs. humnan agent.
2. Knowledge representation in AI (semantic networks, frames), reasoning.
3. Problem solving in state space - uninformed vs informed search, depth-first vs. breadth-first search.
4. Planning and decision making, logic constraints programming, machine learning.
5. Computer vision - image recognition (feature vs structure scene analysis), preprocessing, representation and description of image, object recognition.
6. Natural language processing, artificial neural networks, knowledge systems (structure, characteristics, feedforward vs feedback propagatiion during inference).
7. Genetic algorithms and artificial life, distributed AI and multiagent stystems.
8. Visual perception and cognition.
9. Auditory perception and cognition.
10. Memory, learning and attention.
11. language, thinking and consciousness.
12. Emotions, motivation, attention.
13. Motor system and crossmodal interactions.

Recommended literature:
1. Russell S.J., Norvig P: Artificial Intelligence: A Modern Approach (2nd Edition), Prentice Hall, 2002, ISBN: 0137903952
2. Negnevitsky Michael: Artificial Intelligence: A Guide to Intelligent Systems (2nd Edition), Addison Wesley, 2004, ISBN: 0321204662
3. Poeppel D., Mangun G., Gazzaniga M. (ed.): The Cognitive Neurosciences. 6th ed. MIT Press.

2020. ISBN-13: 978-0262043250

Course language:
Slovak or english

Notes:
Content prerequisities:
basic programing, neurobiology, cognitive psychology, or instructor's consent

Course assessment
Total number of assessed students: 98

A	B	C	D	E	FX
61.22	19.39	13.27	4.08	2.04	0.0

Provides: doc. Ing. Norbert Kopčo, PhD.

Date of last modification: 23.11.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/ AFJ1a/15	**Course name:** Automata and formal languages

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 2 / 1 **Per study period:** 28 / 14
Course method: present

Number of ECTS credits: 4

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities:

Conditions for course completion:
Oral examination.

Learning outcomes:
To provide theoretical background for studying computer science in general, by giving the necessary knowledge in theory of automata.

Brief outline of the course:
1: Chomsky hierarchy of grammars: alphabet, symbol (letter, character), transitive closure, word (string), empty word (empty string), length of a string, concatenation, language, grammar, nonterminal symbol, terminal symbol, initial nonterminal (initial symbol), grammar rule, derivation step, language generated by a grammar, Chomsky hierarchy of grammars - phrase-structure, context sensitive, context free, regular
2: Deterministic finite state automata: finite state automaton, state, input symbol, output symbol, initial state, transition function, output function, examples of automata and their graphic representation, generalized transition and output functions and their basic properties
3: Reduction of automata I: equivalent automata, minimal (optimal) automaton, reachable state, properties of reachable states, elimination of unreachable states
4: Reduction of automata II: equivalent states, k-equivalent states, properties of equivalence and k-equivalence, relation between k-equivalence and (k+1)-equivalence, partitioning the state set into equivalence classes, elimination of equivalent states
5: Reduction of automata III: proof of correctness, unambiguity, and optimality of reduced automaton, testing equivalence of two automata
6: Deterministic finite state acceptors: basic definitions, language recognized by a finite state acceptor, common properties of acceptors and automata with an output, minimizing a finite state acceptor
7: Operations with regular languages: complement, intersection, union, difference, symmetric difference, testing of emptiness, inclusion, equality, and disjointness for regular languages
8: Nondeterministic finite state acceptors: definition, transition function, language recognized by a nondeterministic acceptor, elimination of nondeterminism
9: epsilon-acceptors: definition, properties, elimination of epsilon-transitions

10: Regular grammars: regular grammar, extended regular grammar, transformation of acceptor to a regular grammar, transformation of extended regular grammar to an epsilon-acceptor
11: Regular expressions I: basic properties, transformation of regular expression to an epsilon-acceptor
12: Regular expressions II: regular equations, valid algebraic manipulations with regular expressions, solving an equation with a single unknown variable, solving a system of regular equations, transformation of acceptor to a regular expression
13: Another constructions: review of transformations among various representations, an example of a direct transformation of a grammar to a regular expression, closure of the class of regular languages under another language operations – concatenation and Kleene star, mirror image
14: Another operations: homomorphism and inverse homomorphism, a context-free language that is not regular

Recommended literature:

J.E. Hopcroft, R.Motwani, J.D. Ullman: Introduction to automata theory, languages, and computation, Addison-Wesley, 2001.
J. Shallit: A second course in formal languages and automata theory, Cambridge University press, 2009.
M. Sipser: Introduction to the theory of computation, Thomson Course Technology, 2006.

Course language:
Slovak or English

Notes:

Course assessment
Total number of assessed students: 895

A	B	C	D	E	FX
26.59	18.21	23.46	17.09	9.83	4.8

Provides: prof. RNDr. Viliam Geffert, DrSc., RNDr. Dominika Pališínová, RNDr. Juraj Šebej, PhD.

Date of last modification: 23.11.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/ KKV1/21	**Course name:** Classical and quantum computations

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 3 / 2 **Per study period:** 42 / 28
Course method: present

Number of ECTS credits: 6

Recommended semester/trimester of the course:

Course level: II., N

Prerequisities:

Conditions for course completion:
Successful completion of the subject is conditioned by proper acquisition of basic concepts, algorithms and models and demonstrating the ability to apply them creatively. The acquisition of knowledge takes place:
- continuously during the semester in the form of partial assignments,
- a written test during the semester,
- a written test at the exam,
- oral exam.

In order to receive an evaluation, it is necessary to obtain at least 50% of points from each of the three parts (assignments during the semester, written part of the exam, oral part of the exam). The detailed evaluation method is published in the AIS.

Learning outcomes:
By completing the subject, the student will get:
- knowledge of the classification and design of probabilistic algorithms,
- basic knowledge of the principles of quantum computers and their differences compared to classical computing models,
- knowledge and skills about the design and functioning of quantum computing and become familiar with the most well-known algorithms,

= basic quantum computer programming skills.

Brief outline of the course:
1. Introduction to quantum quantum computers. Basics of classical complexity theory.
2. Boolean circuits and their basic properties.
3. Probability algorithms.
4. BPP class and probability testing.
5. Basic properties of circuits and Fermat's test.
6. Miller - Rabin's test and the position of the BPP class in the hierarchy of complexity models.
7. Introduction to quantum computing and mathematical foundations of quantum theory.
8. Spectral representation of self-adjoint operators.
9. Quantum states and Hilbert vector spaces.
10. Basic quantum operators and basic quantum algorithms.

11. Quantum teleportation, superdense coding and Grover's algorithm.
12. Fourier transformation.
13. Shor's algorithm.

Recommended literature:

1. BERMAN,G.P., DOOLEN,G.D., MAINIERI, R., TSIFRINOVIC, V.I. Introduction to Quantum Computers. World Scientific, 2003.
2. GRUSKA, J. Quantum Computing. McGraw-Hill, 1999.
3. JOHNSON, G. A Shortcut Through Time: The Path to the Quantum Computer, Knopf 2003.
4. KITAEV, A.Y., SHEN, A.H., VYALYI, M.N. Classical and Quantum Computation. American Mathematical Society, 2002.
5. NIELSEN, M.A., CHUANG, I.L. Quantum Computation and Quantum Information. Cambridge University Press, 2000.
6. HIRVENSALO, M., Quantum Computing, Springer 2004

Course language:
Slovak or english

Notes:
Content prerequisites:
Linear algebra, Group theory, Probability theory, Theory of algorithms, Introduction to quantum computers.

Course assessment
Total number of assessed students: 83

A	B	C	D	E	FX
26.51	40.96	15.66	4.82	2.41	9.64

Provides: prof. RNDr. Gabriel Semanišin, PhD., RNDr. Marek Semjan

Date of last modification: 25.07.2022

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ UNV1/15	**Course name:** Computational and cognitive neuroscience I

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 2 / 2 **Per study period:** 28 / 28
Course method: present

Number of ECTS credits: 5

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities:

Conditions for course completion:
Midterm exam
Final exam consisting of written and/or oral part

Learning outcomes:
Overview anatomy, physiology, and cognitive processes in the human brain with focus on computational aspects of cognition and computational tools used in neuroscience.

Brief outline of the course:
1. Intro to neural and cognitive science
2. Overview of anatomy and physiology of the central nervous system (CNS)
3. Methods of study in neuroscience. Sensory, motor and associative brain areas.
4. Neuron: anatomy, types, action potential
5. Propagation of signals in the neuron, neural coding.
6. Synaptic transmission and plasticity - neural basis of learning and memory.
7. Psychology of memory and learning.
8. Vision: Intro. Perception of brightness, edges, color. Model BCS/FCS. Perception of size and sitance.
9. Hearing and auditory cognition.
10. Language, psycholinguistics, speech perception and production.
11. Attention.
12. Crossmodal interaction (vision, hearing, touch).
13. Reasoning and decision making.

Recommended literature:
1. Poeppel D., Mangun G., Gazzaniga M. (ed.): The Cognitive Neurosciences. 6th ed. MIT Press. 2020. ISBN-13: 978-0262043250
2. Dayan P and LF Abbott: Theoretical Neuroscience - Computational and Mathematical Modeling of Neural Systems. MIT Press, 2005 ISBN-13: 978-0262541855
3. Thagard P: Mind: Introduction to Cognitive Science, 2nd Edition. Bradford Books. ISBN-13⁀: ⁀978-0262701099

Course language:

Slovak or English

Notes:
Content prerequisites:
Algebra, programming (Matlab).

Course assessment
Total number of assessed students: 32

A	B	C	D	E	FX
18.75	21.88	25.0	21.88	9.38	3.13

Provides: doc. Ing. Norbert Kopčo, PhD., RNDr. Keerthi Kumar Doreswamy, Ing. Udbhav Singhal, Mgr. Ondrej Spišák

Date of last modification: 08.01.2022

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ VKN/15	**Course name:** Computational and cognitive neuroscience II
Course type, scope and the method: **Course type:** Lecture / Practice **Recommended course-load (hours):** **Per week:** 2 / 2 **Per study period:** 28 / 28 **Course method:** present	
Number of ECTS credits: 5	
Recommended semester/trimester of the course:	
Course level: II., N	
Prerequisities:	
Conditions for course completion: Midterm exam Final exam consisting of written and/or oral part	
Learning outcomes: Advanced topics in computational and cognitive neuroscience, and in the tools used in neuroscience.	
Brief outline of the course: 1. Intro: Cognitive psychology, neural modeling. Theme 1: Topics in cognitive and neural science 2. Neural basis of vision 3. Visual object recognition and visual scene analysis 4. Auditory cognition. Echo suppression. Auditory scene analysis 5. Cortical sound processing. 6. Other topics in the study of brain and main: thinking, consciousness, emotions, motivation Topic 2: Modeling in cognitive and neural science 7. Intro 8. Connectionism, STM and LTM modeling 9. Additive and shunting neural networks. 10. Learning rule Outstar. 11. Adaptive resonance theory. 12. Statistical and decision-theory modeling Topic 3: Current research at UPJS 13. Invited lecture	
Recommended literature: 1. KANDEL, E. R., SCHWARTZ, J. H. and JESSELL, T.M.: Principles of Neural Science. McGraw-Hill, 2021 ISBN-13: 978-1259642234 2. Dayan P and LF Abbott: Theoretical Neuroscience - Computational and Mathematical Modeling of Neural Systems. MIT Press, 2005 ISBN-13: 978-0262541855 3. Thagard P: Mind: Introduction to Cognitive Science, 2nd Edition. Bradford Books. ISBN-13 : 978-0262701099	

4. HERTZ, J., KROGH, A. and PALMER R. G.: Introduction to the theory of neural computation. Addison-Wesley 1991 ISBN-13: 978-0201515602

Course language:
Slovak or English

Notes:
Content prerequisites:
basics of neurobiology, cognitive psychology, linear algebra and differential equations, programing, or instructor's consent

Course assessment
Total number of assessed students: 9

A	B	C	D	E	FX
44.44	11.11	22.22	11.11	11.11	0.0

Provides: doc. Ing. Norbert Kopčo, PhD., RNDr. Keerthi Kumar Doreswamy, Ing. Udbhav Singhal, Mgr. Ondrej Spišák

Date of last modification: 08.01.2022

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ VYZ1/15	**Course name:** Computational complexity

Course type, scope and the method:
Course type: Lecture
Recommended course-load (hours):
Per week: 2 **Per study period:** 28
Course method: present

Number of ECTS credits: 4

Recommended semester/trimester of the course:

Course level: II., N

Prerequisities:

Conditions for course completion:
Oral examination.

Learning outcomes:
To give students theoretical background in computational complexity and theory of NP-completeness.

Brief outline of the course:
1: Introduction: the notion of computational complexity, computational time, computational model, example - the problem of sorting, computational complexity as an asymptotic function
2: Basic computational models: RAM and RASP computers, the cost of an elementary step on these computers, single-tape Turing machine, multi-tape Turing machine, nondeterministic variants of these computational models, transformations among these models with respect to the time complexity
3: The classes P and NP: basic definitions, presenting (un)undirected graphs on the input, 3COL – the set of all 3-colorable graphs is in NP, 2COL - the set of all 2-colorable graphs is in P, SAT – the set of satisfiable Boolean formulas is in NP, CNF-SAT - Boolean formulas in conjunctive normal form
4: Variants of P and NP: decision problem, the problem of finding a solution, optimization problem, polynomial conversions among different variants
5: NP-completeness: reducibility in polynomial time and its transitivity, definition of the NP-completeness and its basic properties
6: NP-completeness of SAT
7: Variants of SAT: 3CNF-SAT - satisfiability of Boolean formulas in 3-conjunctive normal form, kCNF-SAT, CNF-SAT - satisfiability in k-conjunctive (conjunctive) normal form, 2CNF-SAT is in P
8: 3COL and its variants: 3COL (the problem of coloring vertices of a graph with 3 colors) in NP-complete, consequently: for each k>3, kCOL (the problem of coloring with k colors) is NP-complete as well
9: Colorability of a planar graph with three colors: presenting a planar graph on the input, the proof of NP-completeness, coloring with a larger number of colors
10: Another NP-complete problems: Exact set cover, Clique, Vertex cover

11: Hamiltonian path: Hamiltonian path in a directed and in undirected graph
12: Subset-sum-like problems: Subset Sum - the problem of whether any subset of the integers sum to precisely a target sum, Partition - the problem of whether a given multiset of positive integers can be partitioned into two subsets with equal sums, a "more relaxed" version of Partition - achieving an approximate equality of the sums, distribution of tasks among K parallel processors
13: Beyond P a NP: a review of the basic complexity classes - L, NL, P, NP, PSpace , NPSpace, ExpTime, NExpTime, …, simulation of (non)deterministic space in (non)deterministic time, conversions in opposite directions
14: PSpace: QBF - true quantified Boolean formulas, prenex normal form, Pspace completeness of QBF, PSpace = NPSpace

Recommended literature:

1. J.E. Hopcroft, R.Motwani, J.D. Ullman: Introduction to automata theory, languages, and computation, Addison-Wesley, 2007.
2. M. Sipser: Introduction to the Theory of Computation, Thomson, 2nd edition, 2006.
3. L.A.Hemaspaandra, M.Ogihara: Complexity theory companion, EATCS series, texts in computer science, Springer-Verlag, 2002.
4. S. Arora, B. Barak: Computational Complexity: A Modern Approach, Cambridge Univ. Pess, 2009. 5. G.Brassard, P.Bradley: Fundamentals of algorithmics, Prentice Hall, 1996.
6. D.P.Bovet, P.Crescenzi: Introduction to the theory of complexity, Prentice Hall, 1994.
7. C. Calude and J. Hromkovič: Complexity: A Language-Theoretic Point of View, in G. Rozenberg and A. Salomaa, Handbook of Formal Languages II, Springer, 1997.

Course language:
Slovak or english

Notes:
Content prerequisities:
Basic notions from the theory of automata and formal languages.
Basic skills in programming and design of algorithms (in any programming language).
Basics knowledge in mathematical logic, set theory, and graph theory.

Course assessment
Total number of assessed students: 357

A	B	C	D	E	FX
57.7	15.41	12.04	7.28	7.28	0.28

Provides: prof. RNDr. Viliam Geffert, DrSc.

Date of last modification: 23.11.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/ ARP1/15	**Course name:** Computer architecture

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 2 / 1 **Per study period:** 28 / 14
Course method: present

Number of ECTS credits: 4

Recommended semester/trimester of the course:

Course level: I., II., N

Prerequisities:

Conditions for course completion:
Homeworks, active participation in laboratory exercises, final written exam. Final oral examination.

Learning outcomes:
Obtain detailed information about the technical implementation of modern computer systems. Understand the principles of organization of work of processor and computer on concrete examples. Gain basic experience with programming at the level of machine instructions (Assembler language). Understand the current way a computer communicates with I / O devices. Students will get acquainted with the components of current computers, with their properties, connection, principle of operation and possibilities of use. They will be able to make informed decisions about the purchase of computer equipment, identify computer failures; make simpler repairs by replacing modules, including setting them correctly.

Brief outline of the course:
Milestones in computer organization, fundamental limitations. The representation of numbers and the implementation of floating point arithmetic. Combinatorial and sequential circuits, memory organization, RAMs and ROMs. Digital logic level architecture, data path timing, machine cycle. The microarchitecture level, microinstructions and microinstruction control. The instruction set architecture level, data types, addressing modes, instruction types. Instruction execution, pipelining, cache memory. I/O controllers, ports, interrupts, direct memory access. Multicore architectures, processor virtualization. Device drivers, operating system kernel, device-independent software. Laboratory practices and tutorials.

Recommended literature:
1. W. Stallings: Computer Organization and Architecture, Pearson, 2018
2. J. Ledin: Modern Computer Architecture and Organization, Packt Publishing, 2020
3. E. Upton, J. Duntemann, R. Roberts, T. Mamtora, B. Everard: Learning Computer Architecture with Raspberry Pi, Wiley, 2016

Course language:
Slovak or English

Notes:

Content prerequisities: understanding of fundamental concepts of computer architecture and design within the scope of a standard undergraduate course.
The course is not organized annually.

Course assessment
Total number of assessed students: 60

A	B	C	D	E	FX
16.67	18.33	16.67	23.33	18.33	6.67

Provides: doc. RNDr. Jozef Jirásek, PhD., RNDr. Juraj Šebej, PhD.

Date of last modification: 23.11.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/PSIN/15	**Course name:** Computer network Internet

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 3 / 1 **Per study period:** 42 / 14
Course method: present

Number of ECTS credits: 5

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities: ÚINF/PAZ1a/15 or ÚINF/PRG1/15

Conditions for course completion:
Activity at excercises (max 18 points), home work (max 18 points), test (max 30 points).
Verbal exam (min 25 points, max 50 points). Required minimum for passing the course is 55 points.

Learning outcomes:
Students will get the informations about principles and achitecture of Internet. They will understand the principles of ISO/OSI layers reference model for network communication. They will understand the meaning and usage of terms protocol, service, interface. They will analyze the parameters of communication channels, understand the function of interconnection devices (hub, switch, router). They will understand the structure of IP packets, addressing and how packets are transmitted, the principle of routing protocols and the creation of routing tables. They will understand the priciples of acknowledged TCP transport transmission and its implementation. They will know how to use the interface of UDP and TCP protocols in a program code. They will understand the basic application protocols of the Internet.

Brief outline of the course:
1. Introduction to computer networks, internet connection types, delay and loss in packet-switched networks, ISO OSI reference model and TCP/IP protocols family.
2. Application layer: Web and HTTP, protocol FTP ,e-mail and protocols SMTP, POP3, IMAP,
3. Application layer: domain names and DNS, Peer-to-peer applications. Security in computer networks.
4. Transport layer: services, multiplexing and demultiplexing, protocol UDP, reliable data transfer
5. Transport layer: connection oriented transport protocol TCP, flow and congestion control.
6. Network Layer: Internet protocol IPv4, virtual circuit and datagram networks, packet fragmentation, routing table, application protocol DHCP
7. Network Layer: network address translation NAT, ICMP protocol, internet protocol IPv6
8. Network Layer: routing algorithms and protocols, broadcast and multicast routing
9. Link layer: error detection, multiple access methods CSMA/CD and CSMA/CA, Ethernet, frames, protocols ARP and RARP, link layer addressing
10. Link Layer and wireless and mobile networks: hub, switch, virtual LAN, 802.11 Wireless LAN, Bluetooth 802.15, WiMAX 802.16, Mobile IP, mobility in GSM
11. Physical Layer: Communication channels parameters, digital and analog encoding.

Recommended literature:

1. J. F. Kurose, Keith W. Ross: Computer Networking: A Top-Down Approach, 7. edition, 2016
2. A. S. Tanenbaum: Computer Networks, 5. edition, Pearson, 2010
3. W. Stallings: Local and Metropolitan Area Networks, Prentice Hall, 2000
4. E. Comer, R.E. Droms: Computer Networks and Internets, Prentice Hall, 2003
5. W. R. Stevens: TCP/IP Illustrated, Vol.1: The Protocols, Addison-Wesley, 1994

Course language:

Slovak or English

Notes:

Content prerequisities: basic programming skills in Java

Course assessment

Total number of assessed students: 843

A	B	C	D	E	FX
9.49	5.58	12.46	16.37	36.42	19.69

Provides: RNDr. Peter Gurský, PhD., doc. RNDr. JUDr. Pavol Sokol, PhD.

Date of last modification: 04.01.2022

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/ KRP1/15	**Course name:** Cryptographic protocols

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 2 / 2 **Per study period:** 28 / 28
Course method: present

Number of ECTS credits: 4

Recommended semester/trimester of the course:

Course level: I., II., N

Prerequisities:

Conditions for course completion:
Homeworks, active participation in laboratory exercises, presentation of a selected topic at a course seminar. Final written exam.

Learning outcomes:
Understand the problems of designing secure cryptographic protocols for authentication and key management. Know the ways to compromise them and be able to apply methods of proving their correctness. Control some automated verification tools. Understand and be able to apply advanced cryptographic techniques in various application fields - signature schemes, electronic banking, electronic voting. Orientation in current problems of implementation of cryptographic protocols.

Brief outline of the course:
Authentication and key establishment using shared and public key cryptography, key agreement protocols, conference key agreement, zero-knowledge protocols, provable security. Protocol architecture and formal definition, goals for authentication and key establishment, formal verification. Digital signature, implementation, trust distribution.
The final seminar with presentations on selected current topics - electronic banking, electronic voting, secure communication ...

Recommended literature:
1. Colin Boyd, Anish Mathuria: Protocols for Authentication and Key Establishment, Springer, 2020
2. Douglas R. Stinson, Maura B. Paterson: Cryptography: Theory and Practice, Fourth Edition, Chapman & Hall/CRC, 2018
3. Paul C. van Oorschot: Computer Security and the Internet: Tools and Jewels, Springer, 2020
4. Peter Ryan, Steve Schneider: Modeling and Analysis of Security Protocols, Addison-Wesley, 2001

Course language:
Slovak or English

Notes:

Content prerequisities: understanding of fundamental cryptographic concepts and primitives (as taught in the course KRS/15 or in the scope of the textbook "Understanding Cryptography" by Christof Paar and Jan Pelzl).
The course is not organized annually.

Course assessment
Total number of assessed students: 27

A	B	C	D	E	FX
29.63	7.41	14.81	29.63	14.81	3.7

Provides: doc. RNDr. Jozef Jirásek, PhD., RNDr. Rastislav Krivoš-Belluš, PhD.

Date of last modification: 08.01.2022

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/ KRS/15	**Course name:** Cryptographic systems and their applications

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 3 / 2 **Per study period:** 42 / 28
Course method: present

Number of ECTS credits: 6

Recommended semester/trimester of the course:

Course level: I., II., N

Prerequisities:

Conditions for course completion:
Homeworks, midterm written exam, active participation in laboratory exercises.
Final written exam, possibly oral exam.

Learning outcomes:
This course covers the basic knowledge in understanding and using cryptography. The main focus is on definitions, theoretical foundations, and rigorous proofs of security, with some programming practice. Topics include symmetric and public key encryption, message integrity, hash functions, block cipher design and analysis, number theory, and digital signatures. The course also provides an introduction to cryptographic protocols for authentication and key management, including PKI and certificates.

Brief outline of the course:
Classical cryptography, basic information theory, cryptoanalysis, security of classical ciphers. Symmetric ciphers - stream ciphers, block ciphers (DES, AES), modes of operation. Asymmetric ciphers - RSA, Elgamal, elliptic curve cryptosystems. Hash functions, message authentication codes, digital signatures. Authentication, key establishment and distribution, certificates.

Recommended literature:
1. PAAR, Ch., PELZL, J.: Understanding Cryptography, Springer 2010.
2. STINSON, D. R.. PATERSON, M. B.: Cryptography: Theory and Practie. CRC Press, 2018.
3. MAO, W. Modern Cryptography: Theory and Practice. Prentice Hall, 2003.
4. MENEZES, A., OORSCHOT, P. van, VANSTONE, S.: Handbook of Applied Cryptography. CRC Press, 1996.
5. SCHNEIER, B.: Applied Cryptography, 20th Edition, John Wiley & Sons Inc., 2015

Course language:
Slovak or English

Notes:
Content prerequisities: basic number theory and algebra, basic programming

Course assessment
Total number of assessed students: 119

A	B	C	D	E	FX
14.29	9.24	14.29	13.45	31.93	16.81

Provides: doc. RNDr. Jozef Jirásek, PhD., RNDr. Rastislav Krivoš-Belluš, PhD.

Date of last modification: 08.01.2022

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ USU/19	**Course name:** Introduction to machine learning

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 2 / 2 **Per study period:** 28 / 28
Course method: present

Number of ECTS credits: 5

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities:

Conditions for course completion:
Creating a project focused on the application of machine learning algorithms in a selected application domain. Continuous written work focused on the preparation, processing and interpretation of data using machine learning methods. Successful completion of an oral exam focused on selected machine learning methods.

Learning outcomes:
Theoretical knowledge in the area of machine learning. Basic concepts of machine learning. Basic machine learning algorithms.

Brief outline of the course:
1. Basic concepts of machine learning.
2. Basic characteristics of data, types of attributes, characteristics for individual attributes, dependence between attributes.
3. Data sources and their acquisition. Determining the target task.
4. Preparation and cleaning of data, missing values, incorrect inputs.
5. Classification tasks
6. Selected classification methods
7. Evaluation of models - true positive, false positive, true negative, false negative examples.
8. Classification accuracy indicators.
9. Cluster analysis.
10. Association rules.
11. Prediction tasks and selected prediction methods
12. Prediction accuracy indicators.

Recommended literature:
1. AGGARWAL, Charu C. Data mining: a textbook. Cham: Springer, 2015. ISBN 978-3-319-14141-1.
2. ALPAYDIN, Ethem. Introduction to machine learning. 3rd ed. Massachusetts: MIT Press, 2014. ISBN 978-0-262-02818-9.
3. RASCHKA, Sebastian, Mirjalili, Vahid. Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2, 3rd Edition, Packt Publishing Ltd., 2019. ISBN 978-1789955750.

4. WITTEN, I. H., Eibe FRANK a Mark A. HALL. Data mining: practical machine learning tools and techniques. 4th ed. Amsterdam: Morgan Kaufmann, 2017. Morgan Kaufman series in data management systems. ISBN 9780128042915.

Course language:
Slovak or English

Notes:
Content prerequisites:
Basics of programming in Python, or another alternative programming language suitable for data analysis

Course assessment
Total number of assessed students: 20

A	B	C	D	E	FX
90.0	10.0	0.0	0.0	0.0	0.0

Provides: doc. RNDr. Ľubomír Antoni, PhD.

Date of last modification: 20.09.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/ UNS1/15	**Course name:** Introduction to neural networks

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 2 / 2 **Per study period:** 28 / 28
Course method: present

Number of ECTS credits: 5

Recommended semester/trimester of the course:

Course level: I., II., N

Prerequisities:

Conditions for course completion:
The condition for passing the course is the realization of a project with the application of neural networks, successful completion of two written tests in the field of neural networks, their basic types, and genetic algorithms, as well as successful completion of the written and oral part of the exam.

Learning outcomes:
The result of the education is an understanding of the basic principles of neural networks and genetic algorithms. The student will gain the ability to apply the acquired knowledge in intelligent data analysis and also work with a selected tool for modeling neural networks.

Brief outline of the course:
1. Basic concept arising from biology. Linear threshold units, polynomial threshold units, functions calculable by threshold units.
2. Perceptrons. Linear separable objects, adaptation process (learning), convergence of perceptron learning rule, higher order perceptrons.
3. Forward neural networks, hidden neurons, adaptation process (learning), backpropagation method.
4. Recurrent neural networks. Hopfield neural networks, properties, associative memory model, energy function, learning, optimization problems (business traveler problem).
5. Model of gradually created network. ART network, architecture, operations, initialization phase, recognition phase, search and adaptation phase. Use of the ART network.
6. Applications of studied models in solving practical problems.
7. Written test I.
8. Motivation to model genetic elements. Genetic algorithm. Application of genetic algorithms.
9. Genetic programming, root trees, Read's linear code. Basic stochastic optimization algorithms: blind algorithm and climbing algorithm. Forbidden search method.
10. Genetic and evolutionary programming with typing, examples of use. Grammatical evolution.
11. Special techniques of evolutionary computations. Selection mechanisms in evolutionary algorithms.
12. Use of genetic algorithms in training neural networks. Artificial life.
13. Written test II.

Recommended literature:

1. AGGARWAL, Charu C. Neural networks and deep learning: a textbook. Cham: Springer, 2018. ISBN 978-3319944623.
2. KVASNIČKA, Vladimír. Úvod do teórie neurónových sietí. [Slovenská republika]: IRIS, 1997. ISBN 80-88778-30-1.
3. KVASNIČKA, Vladimír. Evolučné algoritmy. Bratislava: Vydavateľstvo STU, 2000. Edícia vysokoškolských učebníc. ISBN 80-227-1377-5.
4. MITCHEL, Melanie. An Introduction to Genetic Algorithms. Cambridge: MIT Press, 2002. ISBN 0-262-63185-7.
5. SINČÁK, Peter, ANDREJKOVÁ, G. Úvod do neurónových sietí, I. diel, Košice: ELFA, 1996. ISBN 808878638X

Course language:

Slovak or English

Notes:

Content prerequisites:
Basics of programming in Python, or another alternative programming language suitable for data analysis

Course assessment

Total number of assessed students: 472

A	B	C	D	E	FX
17.16	17.58	22.25	17.8	21.19	4.03

Provides: doc. RNDr. Ľubomír Antoni, PhD., RNDr. Šimon Horvát, PhD.

Date of last modification: 23.11.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/ZLI/21	**Course name:** Linux basics

Course type, scope and the method:
Course type: Practice
Recommended course-load (hours):
Per week: 2 **Per study period:** 28
Course method: present

Number of ECTS credits: 2

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities:

Conditions for course completion:
The condition for passing the course is: 1. Homeworks (50% of the total number of points), 2. Written final theoretical exam (25% of the total number of points), 3. Written final practical exam (25% of the total number of points).

Learning outcomes:
The result of the education is an understanding of the theoretical and practical background for studying computer science, by giving the necessary knowledge in the usage of Unix/Linux operating systems.

Brief outline of the course:
1. Introduction to Unix/Linux systems, 2. Linux ommand line, 3. Text processing tools, 4. Managing files, 5. Managing users, groups and rights, 6. Managing processes, 7. Managing software and packages, 8. Administering the system - system booting, jobs, logging,9. Basic networking, 10. Managing network interfaces, 11. Managing disk partitions, 12. Exam.

Recommended literature:
1. LPIC-1 Exam 101. LPI [online]. Canada: The Linux Professional Institute, 2021 [cit. 2021-9-22]. Dostupné z: https://learning.lpi.org/en/learning-materials/101-500/, 2. LPIC-1 Exam 102. LPI [online]. Canada: The Linux Professional Institute, 2021 [cit. 2021-9-22]. Dostupné z: https://learning.lpi.org/en/learning-materials/102-500/, 3. Linux - Dokumentační projekt [online]. 4. Praha: Computer Press, 2007 [cit. 2021-9-22]. Dostupné z: https://i.iinfo.cz/files/root/k/LDP_4.pdf.

Course language:
Slovak or English

Notes:

Course assessment
Total number of assessed students: 107

A	B	C	D	E	FX
44.86	14.95	17.76	7.48	5.61	9.35

Provides: doc. RNDr. JUDr. Pavol Sokol, PhD., RNDr. Richard Staňa
Date of last modification: 04.01.2022
Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/ LAD1/15	**Course name:** Logical aspects of databases

Course type, scope and the method:
Course type: Lecture
Recommended course-load (hours):
Per week: 2 **Per study period:** 28
Course method: present

Number of ECTS credits: 4

Recommended semester/trimester of the course:

Course level: II., N

Prerequisities:

Conditions for course completion:
Satisfiable understanding of basic concepts.

Learning outcomes:
Ability to correctly formalize databases.

Brief outline of the course:
1.-3. Basic concepts of logic – a symbol, a term, a formula, an interpretation
4. Formalization of a table and a database
5. Conjunctive queries
6. Conjunctive calculus
7. Relations between conjunctive calculus and conjunctive queries
8.-10. Relational algebra
11.-12. Relations of different models of databases

Recommended literature:
https://ics.upjs.sk/~krajci/skola/vyucba/ucebneTexty/LAD-presentation.pdf

Course language:
Slovak

Notes:

Course assessment
Total number of assessed students: 93

A	B	C	D	E	FX
44.09	18.28	17.2	10.75	7.53	2.15

Provides: prof. RNDr. Stanislav Krajči, PhD.

Date of last modification: 23.11.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/ MTL/15	**Course name:** MATLAB and neurocognition

Course type, scope and the method:
Course type: Practice
Recommended course-load (hours):
Per week: 2 **Per study period:** 28
Course method: present

Number of ECTS credits: 2

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities:

Conditions for course completion:
Written quizes, midterm and final exam.

Learning outcomes:
Intro to programming in MATLAB with focus on its usage in neural and cognitive Science.

Brief outline of the course:
1. Intro to Matlab
2. Navigation, interaction, variables, vectors, matrices, scripts, toolboxes
3. Interaction with humans in behaviroal experiments
4. Auditory and visual stimulus generation
5. Analysis and visualization of behavioral data
6. Analysis of neurophysiological data
7. Analysis of neuroimaging data.
8. Cognitive and neural modeling in Matlab
9. Auditory modeling tools
10. Visual modeling tools
11. Tools for modeling of learning
12. Tools for psychological experiments

Recommended literature:
1. Wallisch P, et al. MATLAB for Neuroscientists: An Introduction to Scientific Computing in MATLAB. Academic Press 2008. ISBN-13: 978-0123838360
2. Stork D, Yom-Tow E: Computer Manual in MATLAB to accompany Pattern Classification, 2nd Edition, Wiley, 2004 ISBN-13: 978-0471429777
3. Dayan P and LF Abbott: Theoretical Neuroscience - Computational and Mathematical Modeling of Neural Systems. MIT Press, 2005 ISBN-13: 978-0262541855

Course language:
Slovak or English

Notes:
Content prerequisities:
basic programing skills or instructor's consent

Course assessment
Total number of assessed students: 11

A	B	C	D	E	FX
18.18	27.27	18.18	36.36	0.0	0.0

Provides: doc. Ing. Norbert Kopčo, PhD., RNDr. Keerthi Kumar Doreswamy, Ing. Udbhav Singhal, Mgr. Ondrej Spišák

Date of last modification: 08.01.2022

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/ PDS1/21	**Course name:** Parallel and distributed systems

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 2 / 2 **Per study period:** 28 / 28
Course method: present

Number of ECTS credits: 5

Recommended semester/trimester of the course:

Course level: II., N

Prerequisities:

Conditions for course completion:
Home assignments, class project from tutorials, midterm written exam. Final written and oral exam.

Learning outcomes:
Understand the principles, basic problems and algorithms of parallel programming. Be able to implement synchronization procedures and manage and use interprocess communication. Master the basics of GPU programming. Understand the differences between parallel and distributed computational models. Master basic distributed algorithms and know how to implement them. Understand the problems of creating a distributed system environment and know how to solve them. Be able to use distributed environments in practical applications.

Brief outline of the course:
Parallel architectures, parallel computational model, access to shared memory. Basic algorithms, scaling, optimality. Effective methods of parallel search and sorting. Working in a GPU environment. Distributed computational model, communication protocols, characteristics of distributed systems. Intercomputer communication, distributed synchronization algorithms, transactions, termination and deadlock detection. Consistency issues with distributed memory sharing. Distributed application environment. Reliable calculations in an environment with errors.

Recommended literature:
1. J. JáJá: An Introduction to Parallel Algorithms, Addison-Wesley, 1992, ISBN 0-201-54856-9
2. P. Sanders, K. Mehlhorn, M. Dietzfelbinger, R. Dementiev: Sequential and Parallel Algorithms and Data Structures, Springer, 2019
3. Sukumar Ghosh: Distributed Systems and Algorithms (Second Edition), CRC Press 2014
4. M. Raynal: Distributed Algorithms for Message-Passing Systems, Springer, 2013
5. Gerard Tel: Introduction to Distributed Algorithms, Cambridge University Press, 2001

Course language:
Slovak or English

Notes:
Content prerequisities: basic of concurrent programming, basic of operating system principles

Course assessment
Total number of assessed students: 47

A	B	C	D	E	FX
19.15	6.38	17.02	17.02	25.53	14.89

Provides: doc. RNDr. Jozef Jirásek, PhD., RNDr. Rastislav Krivoš-Belluš, PhD., Bc. Marián Dvorský, RNDr. Ladislav Mikeš, PhD.

Date of last modification: 23.11.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ PMO1/15	**Course name:** Proces modelling

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 2 / 2 **Per study period:** 28 / 28
Course method: present

Number of ECTS credits: 5

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities: ÚINF/PAZ1b/15 and ÚINF/DBS1a/15 and ÚINF/SWI1a/15

Conditions for course completion:
The interim evaluation is based on the evaluation of partial tasks within the solution of the semester project.
The final assessment is given on the basis of the interim assessment and the result of the exam. On the exam, it is required to prove the ability to orient oneself in the presented issue, to master the theoretical foundations of process modeling, basic skills for the creation and interpretation of process models.
The evaluation is awarded if the student gets at least 50% of the possible points from each part of the exam. Detailed requirements are given in the AIS.

Learning outcomes:
By completing the subject, the student:
- acquires knowledge about the theoretical starting points and basics of process modeling,
- can master the basic principles of creating process models
- get familiar with standard languages for process modeling
- will gain practical experience in creating models using selected modeling tools.

Brief outline of the course:
1. Introduction to process modeling.
2. Development of approaches to the development of large software systems.
3. Theoretical foundations of process modeling.
4. Petri nets.
5. Process orchestration.
6. Choreography of processes.
7. Selected properties of processes and process models.
8. Architectures of process models.
9. Methodologies and standards.

Recommended literature:
1. Ehrig, H.; Juhas, G.; Padberg, J.; Rozenberg, G. (Eds.), Advances in Petri Nets, Lecture Notes in Computer Science , Vol. 2128 (2001)

2. Eshuis, R. ; Wieringa R.: Comparing Petri Net and Activity Diagram Variants for Workflow Modelling – A Quest for Reactive Petri Nets, [dostupné online http://is.tm.tue.nl/staff/heshuis/pnt.pdf]
3. Madison D., Process Mapping, Process Improvement and Process Management, Paton Press 2005
4. Weske, M. Business Process Management, Springer 2007
5. White S.A., Miers D., Fischer L., BPMN Modeling and Reference Guide, Future Strategies Inc., Lighthouse Pt 2008
6. White:, S.A. Process Modeling Notations and Workflow Patterns, [available online http://www.omg.org/bp-corner/bp-files/Process_Modeling_Notations.pdf]

Course language:
Slovak or English

Notes:
Content prerequisities: programming, bases of software engineering and database management systems, bases of project management

Course assessment
Total number of assessed students: 54

A	B	C	D	E	FX
16.67	24.07	27.78	18.52	7.41	5.56

Provides: prof. RNDr. Gabriel Semanišin, PhD.

Date of last modification: 25.07.2022

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ SPP1b/15	**Course name:** Programming environments in schools II

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 2 / 2 **Per study period:** 28 / 28
Course method: present

Number of ECTS credits: 4

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities: ÚINF/SPP1a/15

Conditions for course completion:
Conditions for ongoing evaluation:
1. Educational software or game programmed in the Scratch environment,
2. A programming etude created for learning of programming in the MIT App Inventor environment.
3. Educational or assistive software programmed in the MIT App Inventor environment.
4. A programmed project using the BBC micro: bit kit.
Conditions for successful completion of the course:
Obtaining at least 50% of points for ongoing assignments.

Learning outcomes:
After completing this course, students are able to:
a) get an overview of educational programming environments,
b) acquire programming skills in selected educational programming environments,
c) develop the ability to design and program educational software for devices using their sensors and actuators.

Brief outline of the course:
1. Teaching algorithmization and programming in primary and secondary school - objectives, content, textbooks and methodological materials. Algorithmic computer games.
2. Programming in the Scratch environment.
3. Programming in the Scratch environment.
4. Programming in the Scratch environment.
5. Programming of mobile devices in the MIT App Inventor environment.
6. Programming of mobile devices in the MIT App Inventor environment.
7. Programming of mobile devices in the MIT App Inventor environment.
8. Programming of mobile devices in the MIT App Inventor environment.
9. Programming of mobile devices in the MIT App Inventor environment.
10. Programming BBC micro: bit kits in MS MakeCode environment.
11. Programming BBC micro: bit kits in MS MakeCode environment.
12. Overview of educational programming initiatives and development environments.

Recommended literature:

BELL, Charles A., 2017. Micropython for the internet of things: a beginner's guide to programming with Python on microcontrollers. New York, NY: Springer Science+Business Media. ISBN 9781484231227.
GUTSCHANK, Jörg et al., 2019. Coding in STEM Education [online]. Berlin: Science on Stage Deutschland e.V., 76 p. [cited 2021-7-10]. ISBN 978-3-942524-58-2. Available from: https://www.science-on-stage.eu/sites/default/files/material/coding_in_stem_education_en_2nd_edition.pdf
ŠNAJDER, Ľubomír, Gabriela LOVÁSZOVÁ, Viera MICHALIČKOVÁ and Ján GUNIŠ, 2020. Programovanie mobilných zariadení [online]. Bratislava: Centrum vedecko-technických informácií SR, 300 p. [cited 2020-11-30]. ISBN 978-80-89965-63-2. Available from: https://registracia.itakademia.sk/media/themes/nip-pmz.pdf
WOLBER, David, 2014. App Inventor: Vytvořte si vlastní aplikaci pro Android. Brno: Computer Press. ISBN 978-80-251-4195-3.
LOVÁSZOVÁ, Gabriela, Jana GALBAVÁ, Viera PALMÁROVÁ and Monika TOMCSÁNYIOVÁ, 2010. Ďalšie vzdelávanie učiteľov základných škôl a stredných škôl v predmete informatika: Malé programovacie jazyky. Bratislava: Štátny pedagogický ústav. ISBN 978–80–8118–066–8.
CODE.ORG. Learn today, build a brighter tomorrow.
Code.org [online]. [cited 2021-7-13]. Available from: https://code.org/
THE LIFELONG KINDERGARTEN GROUP AT MIT MEDIA LAB. Scratch - Imagine, Program, Share [online]. [cited 2021-7-13]. Available from: https://scratch.mit.edu/
MASSACHUSETTS INSTITUTE OF TECHNOLOGY. MIT App Inventor
Explore MIT App Inventor [online]. [cited 2021-7-13]. Available from: http://appinventor.mit.edu/
MICRO:BIT EDUCATIONAL FOUNDATION. BBC micro:bit [online]. [cited 2021-7-13]. Available from: https://microbit.org/
SPY O.Z. Učíme s Hardvérom [online]. [cited 2021-7-13]. Available from: https://www.ucimeshardverom.sk/

Course language:
Slovak or English

Notes:
By default, teaching is carried out face to face. If this is not possible (eg due to a pandemic), teaching is provided at a distance through video conferencing programs and LMS.

Course assessment
Total number of assessed students: 20

A	B	C	D	E	FX
25.0	20.0	15.0	20.0	5.0	15.0

Provides: doc. RNDr. Ľubomír Šnajder, PhD.

Date of last modification: 01.08.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ PRO1a/15	**Course name:** Project I.

Course type, scope and the method:
Course type: Practice
Recommended course-load (hours):
Per week: 4 **Per study period:** 56
Course method: present

Number of ECTS credits: 4

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities:

Conditions for course completion:

Learning outcomes:
Practical skills in full-stack technologies on the backend-frontend principle for REST API, databases and SPA frontend.

Brief outline of the course:
1. Creation of project documentation using Markdown and Asciidoc
2. Versioning of source codes via git and the GitLab platform
3. Continuous integration and delivery (CI/CD) via GitLab Pipelines
4. Database migration scripts and deployment to production
5. Securing the backend REST API using HTTP Basic (Spring Boot and Spring Security)
6. Securing the backend REST API using OAuth via an authorization server (Keycloak)
7. Application containerization via Docker
8. Custom docker image and integration into CI/CD
9. Testing applications via Testcontainers
10. Frontend and backend integration via API Gateway and loadbalancer (Traefik)

Recommended literature:
1. Study literature tied to the selected project (according to the client's recommendation)
2. Joost Evertse. Mastering GitLab 12: Implement DevOps culture and repository management solutions. Packt Publishing Ltd, 2019. ISBN 1789534062
3. Lauren#iu Spilcă. Spring Security in Action. Manning, október 2020. ISBN 9781617297731
4. Thomas Vitale. Cloud Native Spring in Action. Manning, november 2022. ISBN 9781617298424
5. Jeff Nickoloff, Stephen Kuenzli. Docker in Action, Second Edition. Manning, október 2019. ISBN 9781617294761

Course language:
Slovak or English

Notes:
content prerequisities: programming skills, basics of shell scripts in Linux

Course assessment
Total number of assessed students: 125

A	B	C	D	E	FX
69.6	9.6	7.2	9.6	3.2	0.8

Provides: RNDr. Róbert Novotný, PhD., RNDr. Peter Gurský, PhD.

Date of last modification: 25.11.2022

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/ PRO1b/15	**Course name:** Project II.

Course type, scope and the method:
Course type: Practice
Recommended course-load (hours):
Per week: 4 **Per study period:** 56
Course method: present

Number of ECTS credits: 4

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities:

Conditions for course completion:
Presentation of the results achieved in solving a specific problem. Uploading a software work. Preparation of materials for the promotion of the final work.

Learning outcomes:
Acquire the way of working on the software work with agile methodology, communication in the software team, solving problems of computer systems administration in all phases of their life cycle.

Brief outline of the course:
Work in a 4-5 member team on the development, testing of a software product under the guidance of a mentor from software companies. Improving with continuous integration and working with git in command lines. Software development using Agile methodology.
1. Introduction to the Software Project, team building.
2. Presentation of projects and assignment of Projects to individual teams.
3. CI / CD Pipeline
4. JUnit Tests
5. Selenium Tests
6. Presentation of the current state of the projects
7. Presentation of the current state of the projects
8. Stress tests
9. Presentation of new technologies from the project
10. Presentation of new technologies from the project
11. Presentation of the final project.
12. Presentation of the final project.

Recommended literature:
1. https://www.udemy.com/course/ Git & GitHub - The Complete Git & GitHub
2. https://www.jenkins.io/doc/
3. Study literature tied to the selected project (according to the client's recommendation)
4. "What is Agile Software Development?". Agile Alliance. 8 June 2013. Retrieved 4 April 2015.

Course language:
Slovak or english

Notes:
Content prerequisities:
advanced programming skills

Course assessment
Total number of assessed students: 89

A	B	C	D	E	FX
57.3	15.73	8.99	8.99	3.37	5.62

Provides: Mgr. Alexander Szabari, PhD., RNDr. Róbert Novotný, PhD., Mgr. Patrik Pekarčík

Date of last modification: 23.11.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice

Faculty: Faculty of Science

Course ID: ÚINF/ PRM1/15	**Course name:** Project management

Course type, scope and the method:
Course type: Lecture / Practice
Recommended course-load (hours):
Per week: 2 / 1 **Per study period:** 28 / 14
Course method: present

Number of ECTS credits: 4

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities:

Conditions for course completion:
The ongoing evaluation consists of the evaluation of the sub-tasks related to the project design. The final evaluation is based on a written and oral exam. The result of the ongoing evaluation will also be included in the overall evaluation.

Learning outcomes:
Gain basic knowledge and skills related to project preparation, project mplementation and project evaluation. Acquire basic knowledge of project team management and organization.

Brief outline of the course:
1. Introduction to project management.
2. Project planning. Preparation of project documentation.
3. Project specification.
4. Estimating project Time and Costs.
5. Work organization.
6. Monitoring and project control.
7. Project closure.
8. Project management models.
9. Estimating project times and costs.
10. Project documentation.
11. Specific approaches for projects in the field computer science.
12. Prince2

Recommended literature:
1. BERKUN, S. The Art Of Project Management. O Reilly, 2005.
2. Erik Larson and Clifford Gray : Project Management:
3. PRINCE2. Avaliable on internet: <http://www.prince2.com>.

Course language:
Slovak or english

Notes:

Course assessment
Total number of assessed students: 122

A	B	C	D	E	FX
26.23	25.41	22.13	12.3	5.74	8.2

Provides: Mgr. Alexander Szabari, PhD., prof. RNDr. Gabriel Semanišin, PhD.

Date of last modification: 23.09.2021

Approved:

COURSE INFORMATION LETTER

University: P. J. Šafárik University in Košice	
Faculty: Faculty of Science	
Course ID: ÚINF/ TYS1/15	**Course name:** Typographical systems

Course type, scope and the method:
Course type: Practice
Recommended course-load (hours):
Per week: 2 **Per study period:** 28
Course method: present

Number of ECTS credits: 2

Recommended semester/trimester of the course:

Course level: I., N

Prerequisities:

Conditions for course completion:
Satisfiable ability to correct mainly mathematical typesetting.

Learning outcomes:
To provide the basic information on principles for typesetting of documents containing mathematical formulas.

Brief outline of the course:
1. Principles for typesetting of documents containing mathematical formulas.
2. Typesetting of a plain text, special text symbols, using of text fonts.3
3. TeX macros.
4. Enumerations in text and footnote command. Parameter setting determining the appearance of the pages.
5. Typesetting of mathematical formulas in text and displays, aligning formulas.
6. Making tables and pictures.
7. Definitions, theorems, and proofs in a mathematical document.
8. Contents, bibliography, sections in a document.
9. Pictures.
10.-12. Project.

Recommended literature:
1. D. E. Knuth, The TeXbook, Computers and Typesetting, Addison-Wesley, Reading, Massachusetts, 1986.
2. M. Doob, Jemný úvod do TeXu, CSTUG, 1990; èeský preklad z "A Gentle Introduction to TeX" (text vo¾ne prístupný v CTAN archíve).
3. O. Ulrych, AMS-TeX za 59 minút, (verzia 1.0), Praha, 1989.
4. J. Chlebíková, AMS-TeX (verzia 2.0), Bratislava, 1992.
5. M. Spivak, The Joy of TeX, Amer. Math. Soc., 1986.
6. L. Lamport, LaTeX: A Document Preparation System, Addison-Wesley, Massachusetts, 1986.
7. L. Lamport, MakeIndex: An index processor for LaTeX, 17 February 1987.
8. J. Rybièka, LaTeX pro začátečníky, Konvoj, Brno, 1995.
9. H. Partl, E. Schlegl, I. Hyna, P. Sýkora, LaTeX – Stručný popis.

10. T. Oetiker, H. Partl, I. Hyna, E. Schlegl, M. Kocer, P. Sýkora, Ne příliš stručný úvod do systému LaTeX2e (neboli LaTeX2e v 73 minutách).
11. M. Goossens, F. Mittelbach, and A. Samarin, The LaTeX Companion, Addison-Wesley, Reading, Massachusetts, 1994. Kapitola 8 je volne prístupná v TeX archívoch (ch8.pdf). 4
12. G. Grätzer, Math into LaTeX, 3rd edition, Birkhäuser, Boston, 2000.

Course language:
Slovak.

Notes:

Course assessment
Total number of assessed students: 254

A	B	C	D	E	FX
48.43	17.72	20.08	6.3	6.69	0.79

Provides: prof. RNDr. Stanislav Krajči, PhD.

Date of last modification: 08.01.2022

Approved: