Thematic areas for the Master State Examination in elective courses

Data Analysis and Processing

1. Data types, data preprocessing, data features. Feature Selection (principle, Entropy, Gini index).
2. Frequent pattern mining (basic principles, methods, variants, implementation).
3. Clustering algorithms (representative-based clustering, hierarchical clustering).
4. Density-based clustering, cluster validation), advanced clustering methods (CLARANS, BIRCH, CURE).
5. Decision Trees (principle, algorithm, metrics used for proper splitting attribute selection, pruning).
6. Probability classification (Bayes theorem, Naive Bayes theorem).
7. Support Vector Machines (principle, algorithm, kernel trick).
8. Neural networks (basic principle, learning methods, activation function).
9. Assessment of classification algorithms (error-rate, precision, recall, F-measure).
10. Regression (linear and non-linear regression, regression trees, model quality assessment).
11. Network types. Graph and adjacency matrix as network representation. Data structures for representation of different types of networks, advantages and disadvantages (adjacency matrix, adjacency list, and adjacency tree), operation complexity, hybrid representation.
12. Topological properties of networks, characteristic values and their distribution (degree, path length, radius, clustering coefficient), centrality types.
13. Global properties of networks (small world, scale-freeness, growing and preferential attachment). Power law and its interpretation in real-world network environment.
14. Network models and their properties (Edrös-Rényi, Watts-Strogatz, Barabasi-Albert).
15. Communities. Global and local approaches. Modularity.
16. Advanced network models – multilayer networks, community based models, temporal networks.
17. Networks resilience, information spreading in networks.
18. Algorithm for pattern matching on strings (one/multiple sample searching, regular expression searching, and approximate search).
19. Information retrieval (IR) (models – Boolean, vector, extended Boolean, lexical analysis, stemming and lemmatization, stop-words, index construction, query evaluation, precision, relevancy, recall, F-measure).
20. Linear algebra in IR (dimensionality reduction methods, matrix factorization, latent semantics, random projection).
21. Searching on the web (hypertext documents analysis, structural methods, PageRank and HITS, metasearch and cooperative search).
22. Parallel computing and platforms: Flynn taxonomy, SIMD, MIMD, SPMD. Data and task parallelism. Processes and threads.
23. Shared-memory systems and distributed memory systems. inter-process communication (race condition, deadlock, mutual exclusion). Communication by message passing. OpenMP, MPI.
24. Parallel reduction and parallel scan: basic principles and case-studies in the selected technology

Topics are covered by these courses: Data analysis I, Data analysis II, Data analysis III, Methods of Analysis of Textual Data, Parallel Algorithms I.

Database and Information Systems

1. Database systems modeling, conceptual modeling, data analysis, functional analysis.
2. Relational data model; function dependencies, decomposition and normal forms.
3. Transactions, recovery, log file, ACID, COMMIT and ROLLBACK operations; anomalies of concurrency, techniques and implementations: locking, multi versioning; isolation levels of transactions in SQL.
4. Procedural extensions of SQL: PL/SQL, T-SQL, triggers, cursors, bind variables, bulk operations.
5. Physical implementation of database systems; tables (the heap table, clustered table, hash table) and indices (the B-tree, bitmap index), materialized views, data partitioning, data paging, column vs row stores.
6. Query execution plan, logical and physical operations, random and sequential operations, query execution tuning, join algorithms.
7. Operators of query execution plan; statistics of values in database systems; cost optimization.
8. Physical implementation of data structures and algorithms of query execution; optimization of accesses to the main memory and disk; design and implementation of the cache buffer.
9. Object-relational data model, XML data model: principles, query languages.
10. Data layer of information systems; API, frameworks and implementations; transactions in programming languages, security, object-relational mapping.
11. Distributed database management systems, fragmentation, replication.

These topics cover the following courses: Database and information systems II, Physical implementation of database systems, Algorithms of query evaluation in database systems.

Computer graphics and image analysis

1. Color systems in computer graphics, nonlinearity of graphical output (gamma correction), composition of raster images (alpha channel), HDR.
2. Affine and projective space. Affine and projective transformations and their mathematical notation. Modeling and imaging transformations in computer graphics.
3. Methods of photorealistic images synthesis, recursive ray tracing, radiometry, rendering equation, Monte Carlo approaches in lighting calculation, acceleration methods.
4. Graphics pipeline and implementation of its individual steps, illumination models and shading algorithms, visibility (hidden surface removal), real-time global illumination techniques, brief characteristics of the OpenGL standard.
5. Image and video compression, principles of image manipulation in spatial and frequency domain.
6. Basic methods of image modification and segmentation (filtering, thresholding, edges, areas, corners).
7. Basic methods of object recognition, feature recognition. Universal feature descriptors (e.g. HOG), trainable classifiers (e.g. SVM).
8. Deep neural networks (e.g. convolutional, description of individual types of layers).
9. Reconstruction of 3D objects from 2D images (basic principles).

These topics are discussed in Computer Graphics I, Computer Graphics II, Digital Image Processing, and Image Analysis I.

Computer Systems and Networks

1. Routing in large networks; optimization and load balancing.

2. Switched local area networks with redundancy.
3. Ensuring quality of in service computer networks. Multimedia data transfers.
4. Multicasts in LAN and WAN.
5. Virtual private networks on the 2nd and 3rd layer of the ISO OSI RM – their use for remote LANs, access interconnection and distributed data centers. Dynamic VPN.
6. Applications of MPLS technology – MPLS VPN, pseudo-wires, VPLS, traffic engineering, BGP-free core, 6PE.
7. Protocols and applications for managing, monitoring and diagnostics of computer networks: SNMP MIB, RMON. Netfow. SPAN/RSPAN/VSPAN.
8. Attacks on computer networks, their detection and prevention.
9. Basic concepts and principles of computer operation. Communication with peripherals. Software and hardware communication management. DMA.
10. Characteristics of RISC and CISC processors. Instruction pipelining, branch prediction, hazards.
11. Memory subsystems of computers, memory types. Paging mechanisms and principles of virtual memory, especially in relation to the OS.
12. Video subsystems, displays, principles of computer image generation.
13. External storage media, partitioning of data on physical medium.
14. Processes, scheduling. Process separation and interprocess communication. Device drivers, interrupt handling.
15. Memory organization and memory allocation.
16. Synchronization. Deadlocks, their detection and prevention.
17. File systems – files, directories, implementation, binding to the OS core. Safety.

Topics are covered in following subjects: Routed and Switched Networks, Advanced Computer Networks Technologies, Operating Systems of Mobile Devices, Computer Systems, Programming in Operating Systems, Computer Architecture and Parallel Systems, and Operating Systems.

Software engineering

18. Requirements engineering discipline. Discipline activities, artifact, models. Classification, prioritization, management, traceability, requirements dependency. Characteristics of "well-done" requirements. Analytical mechanism, analytical patterns.
19. Use cases -suggested structure, guidelines for a scenario writing level of use case, extend, extensions, relations among use-cases. Usage during software development.
20. Design of architecture and detail design. Views on architecture. Sources of design advice (Similar system, Reference model, Architectural styles, Design patterns, Design Conventions, Design principles). Key issues in Software Design. Object oriented design – inheritance and composition; dynamic and static dispatch; Liskov substitution principle; Demeter Law; Dependency Inversion.
21. UML – properties of language, description of diagrams, usage of diagrams when create model of requirements and analysis and design.
22. Design patterns – GoF, Patterns of Enterprise Application Architecture (M. Fowler). Enterprise Integration Patterns (G. Hohpe).
23. Web Services, Service Oriented Architecture. Architectonic Style REST. Microservice Architecture.
24. OCL – features, usage – invariants, pre/postcondition, derived attributes, query functions, initial values.

25. The importance of testing. Testing terminology. Test process. Test planning. Verification vs. Validation. Expected Results. Incident Management.
26. Testing through the software lifecycle. Levels of Testing (V-model). Testing in each level. Testing techniques.
27. Software process. RUP, SCRUM, XP – description and comparison.
28. Business process modeling languages – IDEF0, business use case, EPC, BPMN.
29. Formal methods for the specification of business processes – Petri-Nets (formal definition, firing rule, reachability graph, liveness, boundness, safety and soundness. WF-nets.
30. Declarative programming language, substance of functional and logic programming.
31. Configuration management – functions, goals, principles, core concepts, suggested practices and used tools.

Themes are covered by lectures: Software specification methods, Software design and construction, Requirement engineering, Testing and Software Quality, Enterprise Application Development, Standards and process modeling, Software process, Software maintenance and configuration management.

Theoretical Computer Science

1. Finite automata, their modular construction, the use of nondeterminism. Minimization of automata. The use of finite automata for text searching, algorithm „Knuth-Morris-Pratt“ and its complexity.
2. Specification of regular languages by regular expressions and their relation to finite automata. Characterizations of regular languages allowing proofs of non-regularity, and concrete examples.
3. Context-free grammars as a specification tools for example for programming languages. Pushdown automata as a foundation for syntactic analysis of context-free languages. Nondeterministic and deterministic version of pushdown automata and their relations with respect to itself and to context-free grammars.
4. Closure properties of the class of context-free languages. Pumping lemma for context-free languages. Non-context-free languages, Chomsky hierarchy.
5. Mathematical models of algorithms (Turing machines, RAM machines) and the complexity of algorithms based on them. General methods of design of efficient algorithms (divide and conquer, dynamic programming, greedy algorithms for optimization problems, ...)
6. Algorithmically undecidable problems, halting problem. Semi-decidable problems, Post's theorem. Rice's theorem and its consequences for automated verification of properties of programs.
7. Complexity classes, in particular, classes PTIME, NPTIME, PSPACE, NPSPACE, EXPTIME, EXPSPACE, and their relations. Examples of practical problems from the above mentioned classes.
8. Polynomial reducibility between problems. NP-hard and NP-complete problems. PSPACE-complete problems. The question of relations of classes PTIME, NPTIME, and PSPACE.
9. Problems of discrete optimization. Approximation algorithms for NP-hard optimization problems. Approximation ratio, the class of (well) approximable problems. Examples, in particular, the travelling salesman problem (TSP), general and metric.
10. Randomized algorithms, e.g., for primality testing. Properties necessary for a practical use. Applications in cryptography.
11. Deductive reasoning, a definition of a valid inference.
12. Propositional logic, syntax and semantics of the language, proving in propositional logic.

13. First-order predicate logic, syntax and semantics of the language (interpretations, models, satisfiability).
14. Semantic methods of proofs in first-order predicate logic and Aristotle's logic.
15. General resolution method in first-order predicate logic.
16. Proof calculi: definitions and properties of calculi (soundness, completeness, undecidability).
17. Natural deduction in first-order predicate logic.
18. Theories of arithmetic, incompleteness of arithmetic, Gödel's incompleteness theorems.
19. Algebraic theories, theories of relations and lattices.

The topics are covered by courses: Mathematical Logic, Theoretical Computer Science, Advanced Logic, Selected Topics of Theoretical Computer Science.

Last actualization 17.2.2020