Computer Science BSc (2017) – Final Exam topics

Mathematical and computer sciences

1. Discrete and continuous probability distributions. Binomial, Poisson, uniform, exponential, and normal distribution.

   Basic notions concerning data structures: levels of abstraction (logical and implementaton level), abstract data types (homogeneous and heterogeneous, static and dynamic, structure, operations). Elementary data structures: lists, stacks, queues. Sets, multisets, arrays. The representation of trees, tree traversal, deletion and insertion.

2. Basic concepts of probability and combinatorics (permutation, variation, combination). Conditional probability, independence, Bayes theorem.

   Time complexity of algorithms: insertion sort, merge sort, searching in linear and logarithmic time. Quick sort, the minimal number of necessary comparisons. Sorting in linear time: radix sort, bucket sort.

3. Extreme values of functions, function analysis. Least squares method.

   First-order logic languages. Inductive definition of terms and formulas. Free and bound occurrence of variables. Interpretations, variable assignment. Term and formula valuation. Satisfiable and valid formulas, consequences, equivalent formulas. Conjunctive and disjunctive normal forms, prenex formulas. Logical calculus (e.g. sequent calculus).

4. Functions, curves, and surfaces, and their computer visualization.

   State-space representation. Backtracking strategies. Graph-search procedures: depth-first, breadth-first, and the A algorithms. Two-player games, representation of the game. Winning strategy. Min-max procedure.

5. Concept of matrix, matrix operations, determinant, rank. Matrix classes, inverse matrix. Linear transformation. Eigenvalues and eigenvectors.

   Problem-reduction representations and AND/OR graphs. Knowledge representation techniques and uncertainty management (fuzzy logic). The resolution principle. Logic programming and SLD-resolution. Basic techniques in logic programming.

6. Concept and descriptions of graphs. Simple, directed and undirected graphs. Walk, path, connected graphs. Paired graph, complete graph, tree, circle, weighted graph.

   Generative grammars, language classes, Chomsky hierarchy, finite automata, accepting words in linear time, pushdown automata.

7. System of linear equations and its solution by Gaussian elimination.

   Deterministic Turing machines, linear bounded automata, unsolvable problems, time and space complexity. Nondeterministic Turing machines, famous languages classes: P and NP.

8. Statistical sample and basic descriptive statistics, mean and standard deviation. Confidence intervals. The u-test.

   Concept of computer security, main security objectives. Physical, human, technical threats and their countermeasures. Algorithmic control: encryptions, digital signatures, hash functions. AES and RSA algorithms.

Informatical sciences

1. Database, database system, database management system (DBMS): concepts and properties. Entity, attribute, and relationship: concepts and properties. Relational, object-relational and NoSQL databases. Functional dependency. Conceptional database design, ER model, ER-to Relational mapping. SQL: DDL, DML, DCL, simple queries and joining tables.
2. Lexical units. Data types. Named constant. Variable. Expressions. Statements. Program units. Parameter evaluation, parameter passing methods. Block. Scoping, accessibility. Abstract data type. Exception handling.
3. Basic concepts of object-oriented paradigm. Class, object, instantiation. Inheritance, class hierarchy. Polymorphism, method overloading. Scoping, information hiding, accessibility levels. Abstract classes and interfaces. Type members.
4. The concept of operating system. Structure and classification of operating systems. Files and file systems. Special files on Unix systems. Redirection, pipes. Process management. Signals. Task scheduling.
5. Version control, version control systems. Basic concepts of software testing (test levels, test types, test design techniques). Principles of object-oriented programming (GoF, SOLID). Dependency injection. Architectural patterns (MVC). Design patterns. Free and non-free software. Software licences, free and open source licenses.
6. Traditional software development methodologies: waterfall model, V-model, spiral model, prototyping, iterative and incremental models and rapid application development methodology. Agile software development: fundamentals of agile software development methodologies, agile manifesto, and presentation of a freely chosen agile methodology.
7. Basics of the operation of the Web. Web standards and standards organizations. URIs and their syntax. HTTP: the structure of requests and responses, methods, status codes, content negotiation, cookies. Markup languages of the Web: the structure of XML and HTML documents. Stylesheet languages. JSON.
8. Classification categories of the computer networks. Layered network models. Addressing system and control (ICMP) of the IP technology. Basic aspects of the routing and characterization of the routing categories. TCP and UDP mechanisms.