Entrance Exam
Master in Computer Science (Informatics) program
2019

Instructions

- The time limit is 90 minutes.
- If possible, try to divide your attention evenly between the two pages of the exam.
- Don't expect to solve the entire exam, it's intentionally too large.
- You may choose which questions you want to work on. All questions have the same weight.
- Focus on the main idea, don't waste your time on details that don't matter.
- Submit your solutions by emailing them (or a link to a location where they are uploaded) both to `forisek@dcs.fmph.uniba.sk` and to `michal.forisek@gmail.com`

Mathematics

Algebra

A1. a) Give an example of an irrational algebraic number and its minimal polynomial.

   b) Give an example of a group that is not commutative.

   c) Give an example of a finite field.

A2. Consider the set M of all 2×2 matrices of real numbers. Let $\cdot$ denote matrix multiplication.

Is the pair $(M, \cdot)$ a group? Why / why not?

Discrete structures

D1. Use induction to prove the formula for the sum of squares: $1^2 + 2^2 + \cdots + n^2 \ = \ n(n+1)(2n+1)/6$.

D2. a) What is the cardinality of the set of all finite strings formed using the letters a and b?

   b) Consider a circle drawn in the plane. Let I be the set of all points inside the circle and B the set of all points on its boundary. Compare the cardinality of I and B.

Combinatorics and graph theory

K1. Prove or disprove: Given a tree, it is always possible to color some of its vertices white and the remaining ones black in such a way that no two adjacent vertices share the same color.

K2. There is one large round table with $n + 7$ chairs, each of a different color. Exactly n people (with $n > 0$) and seven dwarves want to sit at the table. The seven dwarves want to sit as one contiguous group (on seven adjacent chairs). In how many ways can they all sit at the table?

Logic

L1. Is it possible to construct any binary operator in boolean logic as some finite expression that only contains the operators $\wedge$ and $\vee$ (logical and, logical or)? Why / why not?

L2. In first-order (predicate) logic, suppose the domain are cows and that: $m(x)$ is the unary predicate "x is male" and $p(x, y)$ is the binary predicate "x is the parent of y".

As an example, the statement "every cow has a parent that is not male" can be formalized as $\forall x \exists y \big(p(y, x) \wedge \neg m(y) \big)$. Give formal versions of the following statements:
– A cow cannot be its own parent.
– Cows x and y have the same grandfather.
– Not every cow has a sibling.

Formal languages

F1. a) Show that deterministic finite automata recognize the same family of languages as nondeterministic finite automata.

   b) Show a regular expressions with back references that describes a non-regular language.

   (An example of such a regular expression is sufficient. No proof necessary.)

F2. Consider the following problem: "Given a deterministic Turing machine M and a symbol s that belongs to its work alphabet, decide whether M ever uses the symbol s during its computation on the empty input."

   Is this problem decidable or not? (In your argument you may use the fact that the halting problem is not decidable.)

Efficient algorithms

E1. Choose any comparison-based sorting algorithm whose worst-case time complexity is $O(n \log n)$. Give a high-level sketch of the argument why the claim about its time complexity is true.

E2. There are n people, numbered from 1 to n. Some pairs of people are friends. You want to divide the people into groups. The group sizes may be arbitrary, the only constraint is that whenever two people are friends, they should be put into the same group.

   Give pseudocode of an efficient algorithm that reads the list of pairs of friends and computes the largest possible number of groups.

Operating systems and computer networks

O1. Give a high-level description of how an operating system may implement multitasking.

O2. What range of addresses is represented by the short notation `192.168.0.0/23`? What protocol (including its version number) uses these addresses? What has been the most common practical solution for the fact that the demand for these addresses has already exceeded their supply?

Programming and databases

P1. Imagine that we are implementing a graphics library that should manipulate various 2D objects and draw them onto the screen. Give a small example of how object-oriented programming may be used in the design of such a library. Focus on inheritance and the use of a virtual method.

P2. Suppose we have the following tables in a relational database:

   - Contestant(id,name,country)
   - Level(id,name,score) where score is the number of points awarded for solving the level
   - Solved(timestamp,contestant_id,level_id) with a constraint that all pairs of ids are unique

   Write a SQL query that will produce the list of top 10 contestants. In case of a tie, the contestant who reached their current total score sooner should be ranked higher.

   (Partial score will be awarded for queries that produce something close to the desired result.)