Discrete Mathematics and Mathematical Reasoning
Course Overview and Administrative Matters

Kousha Etessami and Richard Mayr

University of Edinburgh, UK

Outline

Discrete Mathematics and Mathematical Reasoning (DMMR)

A 2nd year course taught by
Lecturers:

- Dr. Richard Mayr, first half of the term
- Dr. Kousha Etessami, second half of the term

Course TA: Daniel Raggi
Course Secretary (ITO): Kendal Reid (kr@inf.ed.ac.uk)
Course web page:

http://www.inf.ed.ac.uk/teaching/courses/dmmr/

Contains important info: lecture slides, tutorial sheet exercises, course organization, etc.

Tutorials

- You should receive email from the ITO (course secretary Kendal Reid) informing you of preliminary allocation of tutorial groups.
- Also, see link on course web page for current assignment of tutorial groups.
- If you can't make the time of your allocated group, please email Kendal suggesting some groups you can manage.
- If you change tutor groups for any reason, you **must** let Kendal and the ITO know. (Because your marked coursework is returned at the tutorial groups.)
- Tutorial attendance is mandatory. If you miss two tutorials in a row, your PT (DoS) will be notified.

Tutorials and (marked) exercises

- Weekly exercise sheets, available every Wednesday 2pm on the course web page.
- The last question on every sheet will be graded. The coursework grade contributes 15% to the total course grade, and every one of the 10 exercise sheets counts $1/10$ of the coursework grade.
- Starting in week 2, deadline for submission of each tutorial sheet is **Wednesday at 4:00pm** at the ITO (they also have a collection box).
- Solutions will be discussed in tutorials the following week. Graded sheets are returned in tutorials (or collected later from the ITO).
- Tutorial attendance is mandatory.

Exception for 1st week: The first exercise sheet is available **today** (16. Sep.) and the solution must be handed in at the ITO by Thursday, 19. September at 4:00pm.

Textbook

- Kenneth Rosen, **Discrete Mathematics and its Applications, 7th Edition, (Global Edition)** McGraw-Hill, 2012.
- Available at Blackwells.
- For additional material see the course webpage.

Grading

- Written Examination: 85%
- Assessed Assignments: 15%
  Each one of the 10 exercise sheets counts equally, i.e., $1/10$.

Class Representatives

- We need to select at least 3 (up to 6) class representatives for this course.
- Class reps can be used by students as a channel to provide feedback about the course to the course staff.
- Class reps collect and forward such feedback, and will also attend a staff-student liason meeting.
- If you wish to volunteer to be a class rep, please give us your name and email address at the end of this lecture.

Clicker System

- This year, during DMMR lectures, we (the lecturers) will occasionally use the Clicker system (which most of you will have experienced in Year 1 maths courses), in order to poll the class for (anonymous) answers to some questions.
- We would therefore like to ask all of you to borrow a Clicker from the library before classes start, if you have not already done so.
- You can borrow a Clicker from the Main Library's help desk on the ground floor, which has a supply of them. They will lend out one Clicker to you on a semester-long basis.
- Clickers should be returned to the library at the end of the first semester, and before the Christmas break at the latest. Note that students who lose a clicker will be charged a replacement fee of GBP 25.

Test of the Clicker System

What is your first language?

1. English
2. Russian
3. Spanish
4. German
5. Some other language.

Test of the Clicker System

What is your first language?

1. English
2. Russian
3. Spanish
4. German
5. Some other language.

Who is your favorite Mathematician?

1. Leonhard Euler
2. Isaac Newton
3. Gottfried Leibniz
4. Carl Friedrich Gauss

Summary of Intended Learning Outcomes

- reason mathematically about basic (discrete) structures (such as numbers, sets, graphs, and trees) used in computer science.
- use of mathematical and logical notation to define and formally reason about mathematical concepts such as sets, relations, functions, and integers, and discrete structures like trees, graphs, and partial orders;
- evaluate elementary mathematical arguments and identify fallacious reasoning
- construct inductive hypothesis and carry out simple induction proofs;
- use graph theoretic models and data structures to model and solve some basic problems in Informatics (e.g., network connectivity, etc.)
- prove elementary arithmetic and algebraic properties of the integers, and modular arithmetic, explain some of their basic applications in Informatics, e.g., to cryptography.

- compare the asymptotic growth growth rates of basic functions; derive asymptotic bounds, and limits, for simple series and recurrence relations.
- Use these to derive bounds on the resource consumption (e.g., running time) of simple iterative and recursive algorithms.
- calculate the number of possible outcomes of elementary combinatorial processes such as permutations and combinations.
- be able to construct discrete probability distributions based on simple combinatorial processes, and to calculate the probabilities and expectations of simple events under such discrete distributions;

Foundations (Chapters 1, 2 and 9)

- Review of propositional logic (see INF1A). Including: basic laws for boolean logic (double negation, commutativity, associativity, distributivity, De Morgan's laws, law of excluded middle,...), normal forms: disjunctive and conjunctive normal form
- Rudimentary predicate (first-order) logic: existential and universal quantification, basic algebraic laws of quantified logic (duality of existential and universal quantification).
- The structure of a well-reasoned mathematical proof; Proof strategies: proofs by contradiction, proof by cases; examples of incorrect proofs (to build intuition about correct mathematical reasoning);
- Sets (naive): operations on sets: union, intersection, set difference, laws of set operations (boolean algebra: same as boolean logic) Venn diagrams the power set operation examples of finite and infinite sets (the natural numbers), examples of set-builder notation (set comprehension notation). Ordered pairs, n-tuples, and Cartesian products of sets

Foundations (cont.)

- Relations: (unary, binary, and n-ary) properties of binary relations (symmetry, reflexivity, transitivity). partial orders, total orders functions viewed as relations.
- Functions: injective, surjective, and bijective functions, inverse functions, composition of functions.
- Rudimentary counting: size of the Cartesian product of two finite sets, number of subsets of a finite set, (number of n-bit sequences), number of functions from one finite set to another

Basic algorithms and their complexity (Chapter 3)

- pseudo-code notation for algorithms
- Basics of growth of function, and complexity of algorithms: asymptotic, Big-O-notation, (and little-o, Big-Omega, Big-Theta, etc.), comparison of growth rate of some common functions:

Basic number theory and cryptography (Chapter 4)

- integers and elementary number theory (divisibility, GCDs and the Euclidean algorithm, prime decomposition and the fundamental theorem of arithmetic).
- modular arithmetic, (congruences, Fermat's little theorem, the Chinese remainder theorem)
- Application: RSA public-key cryptography (explain that primality testing is "easy" (polynomial), while factoring is thought to be "hard" (exponential))

Induction and Recursion (Chapter 5)

- Principle of Mathematical Induction (for positive integers)
- Examples of proofs by (weak and strong) induction
- Recursive definitions, and Structural Induction
- Examples of recursive definitions: well-formed propositional logic syntax, well-formed regular expressions (covered in Inf1-C&L), Fibonnaci numbers, and other, recursively defined functions,
- Examples of structural induction proofs for: propositional logic, regular expressions, Fibonnaci, etc.

Counting (Chapter 6)

- basics of counting
- pigeon-hole principle
- permutations and combinations
- binomial coefficients, binomial theorem, and basic identities on binomial coefficients.
- generalizations of permutations and combinations (e.g., combinations with repetition/replacement)
- Stirling's approximation of the factorial function.

Graphs (Chapter 9, and parts of Chapter 8)

- directed and undirected graph: definitions and examples in informatics.
- adjacency matrix representation
- terminology: degree (indegree, outdegree), and special graphs: bipartite, complete, acyclic, ...
- isomorphism of graphs; subgraphs
- paths, cycles, and (strong) connectivity
- Euler paths/circuits, Hamiltonian paths (brief)
- weighted graphs, and shortest paths (Dijkstra's algorithm).
- bipartite matching: Hall's marriage theorem
- planar graphs (brief), graph coloring (brief)

Trees (Chapter 10)

- rooted and unrooted trees,
- ordered and unordered trees,
- (complete) binary (k-ary) tree,
- subtrees
- examples in informatics
- spanning trees (Kruskal's algorithm, Prim's algorithm.)

Discrete Probability (Chapter 7)

- discrete (finite or countable) probability spaces
- events
- basic axioms of discrete probability
- independence and conditional probability
- Bayes' theorem
- random variables
- expectation; linearity of expectation
- basic examples of discrete probability distributions
- birthday paradox, and other subtle examples in probability.