MATH 231/232/233 SYLLABUS (AS OF FALL 2011)

September 22, 2011

Current text: Discrete and Combinatorial Mathematics, Ralph Grimaldi.

1. Math 231 (basic techniques and language of mathematics)

Note the material in Math 231 is largely assumed in Math 232 and 233, so all this should be regarded as essential core content for the rest of the year.

1.1. **Principles of counting: sections 1.1, 1.2, 1.3, 1.4 [2 weeks].** Counting arrangements, permutations and combinations with or without replacement. Sigma notation for sums, some basic examples like arithmetic and geometric progressions. The binomial theorem and Pascal's triangle. The multinomial theorem. Compositions.

1.2. **Logic: sections 2.1, 2.2, 2.4 (brifly), 2.5 (supplemented with further examples of proofs) [2 weeks].** Logical connectives and truth tables. Logical equivalence, tautology and contradiction. Quantifiers "for all" and "there exists." Methods of proof and discussion of underlying rules of inference. Proof by contradiction, contrapositives. Plenty of further examples of proofs, e.g. ones involving even/odd numbers, rational/irrational numbers, etc.

1.3. **Set theory: brief discussion of sections 3.1, 3.2, 3.3, 3.4 (optional), 8.1, 8.2 [1 week].** Language of sets, Venn diagrams. Relationship between laws of sets and laws of logic. Power sets, cardinality. The princple of inclusion and exclusion, with examples.

1.4. **Induction and recursion: sections 4.1, 4.2, 4.3, 4.4, 4.5 [3 weeks].** The principle of mathematical induction from intuitive point of view. Examples. Recursive definitions and recursive algorithms. Examples. Divisibility. The division algorithm. Prime numbers. The Euclidean algorithm. The fundamental theorem of arithmetic.

1.5. **Functions and bijections: sections 5.1, 5.2, 5.5 and 5.6 [2 weeks].** Formal definition of a function, domain, codomain, image. 1-1 and onto. Invertible functions. Bijections as a tool for counting. Examples of bijections used in counting problems (perhaps refreshing about problems from chapter 1). The pigeonhole principle.

2. Math 232 (graph theory)

The material in Math 232 is largely independent from that of Math 233, so there is substantial flexibility in the topics covered according to instructor taste. Here is one possible course, based largely on Part 3 of the text. This is quite demanding material (which is a consequence of the current textbook.) The primary goal of the course is to consolidate the mathematical language developed in Math 231 in an interesting setting.

2.1. **Recurrence relations: sections 10.1 and 10.2 [1 week].** First and second order homogeneous linear recurrence relations. Examples and applications. If time allows, an optional digression into complex numbers to deal with complex roots case, De Moivre's theorem, links back to Pascal's triangle via multiple angle formulas in trigonometry, etc This could be used to provide further examples of Mathematical Induction.

2.2. **Graph theory: sections 11.1, 11.2, 11.3, 11.4 and 11.5 [4 weeks].** Definitions: directed versus undirected, multigraphs and adjacency matrices. Walks, trails, paths, circuits, cycles. Connected components. Counting edges and vertex degrees. Euler circuits. The hypercube Qn and the complete graph K_n. The complement of a graph. Subgraphs and graph isomorphism. The complete bipartite graph $K_{m,n}$. The Peterson graph. Planar graphs: Kuratowski's theorem and Euler's theorem. The platonic solids, the dual of a planar graph. Hamilton cycles.

2.3. **Trees: sections 12.1 and 12.2 [1.5 weeks].** Definitions, spanning trees, pendant vertices. Basic properties. Cayley's theorem on the number of labelled trees: proof by multinomial theorem and/or bijective proof via Prufer codes. Ordered rooted trees, formulae for number of leaves. Examples from "Counterfeit coin" problem, tournaments. Lexicographic order, Polish notation.

2.4. **Generating functions: parts of sections 10.4 and 10.5 [1.5 weeks].** The idea at this point is to build up to a discussion/proof of the formula for number of binary rooted trees in terms of Catalan numbers and give the proof via generating functions in section 10,5. In order to make sense of that, discussion of power series and generating functions, motivated by the binomial theorem for powers that are not positive integers (without proof), is appropriate. Practise of solving simple recurrence relations using power series as in section 10.4 is helpful, linking back to the material from as appropriate, e.g. power series can be used to reprove the basic results about second order linear recurrence relations from 10.2. More examples of recurrence relations from problems involving graphs can be incorporated as needed.

2.5. **Algorithms: sections 5.7, 12.3, 13.1 and 13.2 [2 weeks].** Sorting algorithms: bubble sort and merge sort. Brief discussion of "Big-Oh" notation for measuring computational complexity, illustrated by complexity of bubble sort and merge sort algorithms. Dijkstra's shortest path algorithm with proof/complexity discussion. Prim's minimal spanning tree algorithm with proof/complexity discussion. Further discussion of algorithms as appropriate.

2.6. **Graph Coloring and Chromatic Polynomials (optional): section 11.6 [1 week].** Map coloring, Graph coloring. Decomposition Theorem for Chromatic polynomials.

3. Math 233 (algebra/cryptography)

Again there is substantial flexibility according to instructor taste. Here is one suggested course based largely on Part 4 of the text proposed by an instructor who has never taught this course. Note this is challenging material in a 200 level class and needs to be taught largely by examples Again the real goal of the course is to develop student's familiarity with abstract mathematical language.

3.1. **Relations: sections 7.1, 7.2, 7.3, 7.4 [1-2 weeks].** Relations and presentations of relations (using matrices and graphs): Topics covered include: Properties of reflexive, symmetric, antisymmetric and transitive. Partial orders and equivalence rela- tions. Some counting problems involving relations. In chapter 14, students will need to understand the construction of $\mathbf{Z}_n = \mathbf{Z}/n\mathbf{Z}$ using equivalence relations and their connections to partitions of sets.

3.2. **Rings and modular arithmetic: sections 14.1, 14.2, 14.3, 14.4 [3-4 weeks].** This chapter is taught mainly by examples (the main examples are $\mathbf{Z}_n$, fields $\mathbf{Z}$, $\mathbf{Q}$, $\mathbf{R}$ and sets with operations Δ, $\cap$). Formal definition of a ring, zero divisors, units and subrings are covered. The primary techniques taught are finding inverses of units in $\mathbf{Z}_n$ using the Euclidean algorithm and computing exponentials in $\mathbf{Z}_n$. Affine cipher shifts are covered in this chapter. Ring homomorphisms and the Chinese Reminder Theorem are also covered.

3.3. **Groups and coding theory: sections 16.1, 16.2, 16.3, 16.4 [3 weeks].** As with chapter 14, this chapter is mainly taught using examples. It may seem strange to cover groups after rings, but the most important group introduced in this chapter is the group of units in $\mathbf{Z}_n$ under multiplication (hence rings are needed first). Cyclic groups, generators, permutation groups (the main example of a non-abelian group), subgroups, cosets and Largrange's thm are also covered. The main application is RSA encryption which uses topics from chapter 14 as well.

3.4. **Polynomial rings, irreducible polynomials and construction of finite fields: sections 17.1, 17.2 [2 weeks].** The goal of this chapter is to show how integer structures can generalized to polynomial structures. The main examples are polynomial rings over $\mathbf{Z}_n$, $\mathbf{Z}$, $\mathbf{Q}$, $\mathbf{R}$, $\mathbf{C}$. Division algorithm, factoring and the Euclidean algorithm for polynomials are covered. Constructing rings $R[x]/(p(x))$ as an analogue to constructing $\mathbf{Z}_n = \mathbf{Z}/n\mathbf{Z}$ using equivalence relations and partitions is also covered.

(Last edited by Boris Botvinnik; Created by Jon Brundan)