DM551/MM851 Oral exam January 27-30(31), 2020

December 11, 2019

1 Order of examination

The secretary has posted lists on the info board in the corridor where the work places are (just before the bridge). Please sign up for one of the free slots before December 20th. If you sign up for a late time and there is a hole before that, you will be moved down. If you do not sign up, you will be assigned in random order. Note that January 31st will only be used if we cannot fit everyone into the first 4 days. Hence you will not be able to sign up for that date initially. Note also that MM851 students should sign up for the 27th (according to the exam schedule).

Remember to show up and do so well ahead of the time corresponding to your number if everyone showed up! You must show up at least 45 minutes before your time slot starts (The first ones on each day should just be there 15 minutes early).

2 Exam Format

The processs starts by you drawing one of the exam questions after which you will have about 25 minutes to prepare your presentation. The exam itself is 25 minutes, including time to find your grade, that is, approximately 20 minutes of exam. You have roughly 13 minutes for your presentation and after that the censor and I will ask questions in other parts of the curriculum.

The main focus is on demonstrating understanding and usage of concepts and methods and to a lesser extend whether you can derive complicated formulas such as deriving the Chernoff bound formulas. Of course you must be able show that you understand the basic formulas and how to use them. You are welcome to choose a small example and use that to illustrate the topic you got.

The censor and I may also ask about your solution of the two projects so you must be able to explain that. This is particularly important for the first project if you worked in a group. You are welcome to use examples from the two projects to illustrate the topic you are covering in the question you got. Remember that the grade is given based on the overall impression of your performance at the oral exam and how well you answered the two exam projects.

3 Pensum

Will be updated until the end of the course. Note that the (few) MM851 students who already have MM541 still have the Rosen material as part of their pensum, but they will not get an exam question in the basic topics from those chapters (see the list of exam questions below).

- Cormen et al, Introduction to algorithms 3rd ed: Section 5.1-5.3, 5.4.3 (until page 136 line -13),page 182-184, 11.3-11.5, 26.1-26.3 and 32.1-32.3.
- Kleinberg and Tardos, Algorithm Design: 13.1-13.5, 13.6, 13.9.
- Rosen 8th ed. Chapters 6,7, 8.1-8.2, 8.5-8.6
- For MM851 students who have taken MM541: Mitzenmacher and Upfal, Probability and Computing, Cambridge UP Chapter 7.1-7.4.
- The material and exercises on all Weekly notes.

4 Exam questions

The stuff in the brackets is just to inspire you, there may be many other things to talk about. Remember that if you choose the easiest material then it is harder to get a top grade, so if you aim high (which many of you should!) then choose something where you can show your qualities. If you just want to pass/get a decent grade, you may choose some of the easier material. In any case do not choose something which you are not sure you can handle!

At the exam ifself you are not allowed to look at your notes, except for a very short list of topics you will cover. We can ask you to stop looking if we feel this is necessary.

DM551/MM851

1. Basic counting problems (pigeon hole principle, generalized permutations and combinations etc)
2. Inclusion-exclusion with applications (derivation of the general formula, number of onto-functions, the hatcheck problem)
3. Recurrence relations (definitions, solution form, (non-)homogeneous recurrence relations, applications in counting problems)
4. Discrete probability, random variables and bounds (expected value, variance, Bayes formula, Markov's inequality, Chebyshev's inequality and Chernoff bounds)
5. Randomized algorithms (Quicksort, median finding and selection, mincut in graphs, generating a random permutation, majority element and more!!!)
6. The probabilistic method (what is it? and how to use it, applied to the k-SAT problem, proving that every graph G has a spanning bipartite subgraph with at least half the edges of G).
7. Probabilistic analysis (using (indicator) random variables, coupon collector, expected running time of quicksort and selection, randomized approximation for max k-SAT).

8. Examples of applications of indicator random variables (find some yourselves, there are many!)

9. Universal hashing (universal hash functions, perfect hashing (also called 2-level hashing), count-min sketch).

10. String matching (naive algorithm, The Rabin-Karp algorithm, Finite automaton based string matching).

11. Maximum flows (Definitions, Ford Fulkerson algorithm, Max-Flow-Min-Cut theorem, Edmonds-Karp Algorithm, bipartite matching)

12. The min-cut problem (randomized algorithm, solution via flows, solution via max-back orderings).

MM851 students who made the special 1st assignment

1. Discrete probability, random variables and bounds (expected value, variance, Bayes formula, Markov's inequality, Chebyshev's inequality and Chernoff bounds)

2. Randomized algorithms (Quicksort, median finding and selection, min-cut in graphs, generating a random permutation, majority element and more!!!)

3. Markov chains (basic definitions, digraph representation, algorithm for 2-SAT, stationary distributions, random walks on graphs).

4. The probabilistic method (what is it? and how to use it, applied to the k-SAT problem, proving that every graph G has a spanning bipartite subgraph with at least half the edges of G).

5. Probabilistic analysis (using (indicator) random variables, coupon collector, expected running time of quicksort and selection, randomized approximation for max k-SAT, running times based on Markov chains).

6. Examples of applications of indicator random variables (find some yourselves, there are many!)

7. Universal hashing (universal hash functions, perfect hashing (also called 2-level hashing), count-min-sketch).

8. String matching (naive algorithm, The Rabin-Karp algorithm, Finite automaton based string matching).
9. Maximum flows (Definitions, Ford Fulkerson algorithm, Max-Flow-Min-Cut theorem, Edmonds-Karp Algorithm, bipartite matching)
10. The min-cut problem (randomized algorithm, solution via flows, solution via max-back orderings).