DM551/MM851 Oral exam January 10-12, 2023. PRELIMINARY VERSION!!!!

November 16, 2022

1 Order of examination

By mid December it will be possible to choose your examination day in itslearning on a first come first serve basis. More info about that later.

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 formulas and concepts as well as 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 solutions of the two projects so you must be able to explain these. 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. That means that your exam projects can help you get a better grade but also that you will fail the exam if you show up at the oral exam and can only answer very little.

3 Pensum

Will be updated until the end of the course!

- 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
- 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, 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 reasonably well!

At the exam ifself you are not allowed to look at your notes, except for a very short list of topics you will cover. There can be no proofs or definitions on the sheet. 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).