

Design and Analysis of Algorithms Exam

- This is a closed book exam; only 1 sheet of handwritten notes is permitted.
- No cell phones, calculators, nor any other electronics.
- You will have 135 minutes for this exam.
- It consists of 5 problems worth 50 points each, plus a problem 6 for an extra 20 points credit.
- Make sure you have answered all problems from the first 5 problems before moving on to problem 6.
- Your score will be taken out of 225 points.

Instruction

Work as many problems as possible. All problems have the same value, but subparts of a problem may have different values (depending on their difficulties, importance, etc). Provide a short preliminary explanation of how an algorithm works before running an algorithm or presenting a formal algorithm description, and use examples or diagrams if they are needed to make your presentation clear. Please be concise and give well-organized explanations. Long, rambling, or poorly written/organized explanations, which are difficult to follow, will receive less credit.

Grading

Expect questions of the following form (note that a problem may consist of a few questions):

- A series of short-answer questions. Although the answers are short, some of the questions may require some thought. Don't worry about skipping a question, or guessing an answer, if you don't see the answer right away.
- Work through some algorithm on a given input, showing intermediate results. I'll pick the algorithm at random because I want you to learn them all. For some more complex algorithms (e.g. Red-Black tree insertion), I may ask you to do only one operation.
- Design and analyze simple algorithms. These are much more unpredictable. I try to ask at least a couple of questions that involve a very simple modification to a homework problem, so be sure you understand the solutions to all the homework problems. Other questions usually involve a slight modification to a problem that we have worked on in class.
- One challenging problem. Don't be upset if you cannot solve this problem. (But at least be sure to read it, since sometimes the problem is not as challenging as I think it is, and there may be a very simple solution.) If you cannot see how to solve the problem, feel free to write down your wild ideas or observations. You might get some partial credit if your idea is good.

When it comes to grading, I tend to give more credit for insight than for memorization. For example, if a problem involves inorder traversal of a binary tree, you will get little credit for just writing down the algorithm. (You could just copy that out of your cheat sheet.) However, if you can say something intelligent about the mathematical structure of the particular problem, or about why some approach is not useful, you may get some partial credit (depending on how much your observation indicates about your understanding of the problem).