COS 521 Advanced Algorithms Fall 2013

Sanjeev Arora

INFORMATION ABOUT TERM PROJECT
Oct 15, 2013

The term project in this course **must** involve algorithms analysis/design in some nontrivial way, and can involve modeling and implementation. It is recommended to relate it to topics seen during the course but discuss with me if you want to do something unrelated. If you can connect it with your own background or special interests, that would be a bonus (for you and for the rest of us).

The default group size is 2, but can be increased by permission. Singleton groups are not allowed.

Please be on the lookout for project ideas from now on. Please give me a 1-page preproposal by November 12. I will give you feedback –-scheduling separate meetings if necessary---by November 21. An updated 2-page preproposal is then due on Dec 2.

Project reports should be about 10 pages long (but there's no page limit) and are due on Dean's date, Jan 14. They should describe the problem in some detail, explaining clearly and with some mathematical detail what you had set out to do, and what you actually managed to accomplish. What were the hurdles, and do they seem inherent?

There will be an extra class meeting on Jan 9 where all groups will give 15-minute presentations on their projects. (The session will last about 3 hours; I will order lunch.) Please let me know asap if this date does not work for you.

Further thoughts on how to choose a project

Note that the project substitutes for a final and the time requirement should be comparable to preparing for a final (say, couple of weeks tops). It is not meant to be publishable work.

This being a course in Advanced Algorithms, the project should be about algorithms and *their analysis*. (Just as in a systems class you would do a project about systems.) The topic could be drawn from another CS area (vision, computer music etc.) but its centerpiece should still be a close relative to some algorithmic idea that you picked up in this course. It can have an implementation part but it should not be the main component; the implementation should serve to test your algorithmic ideas rather than be a software project. Clearly, two weeks is not a lot, so aim low.

Sample project ideas

1): Start from one of the programming assignments (eg portfolio management or image compression) and do a study of more advanced algorithms for it, and maybe implement some on real-life data.

2) Take an idea that was briefly explored (hashing/data sketching, decision-making under uncertainty, linear modeling, markov chains etc.) and use it in some other context. To give an off-the-wall idea: can a simple markovian model predict random bits produced by people? (Quite possible since human brains can't store or process large sequences of bits. So the next bit may be very predictable from the last few bits.)

3) Take an idea from the course and apply it to whatever you are interested in. For example, does SVD or some other linear model lead to a compression algorithm for genomic data your lab is working with?

Try to keep it simple and exploratory. A primary goal for theory is to simplify and focus on the essentials. (Doesn't always work but when it works it is very powerful.)