Final Exam: 2 Short Essays on Algorithms
Wednesday, 11 December 2024, 1:00–3:00 pm, Chapman 206

The in-class Final Exam will be relatively short. I would like you to write two essays, each with length in the range from 250 to 400 words, on certain optimization algorithms we have covered, or on important algorithm components. See the seven topics below, which we have covered in lecture and homework. (This is an exam, not a project!) Make sure to read the indicated sections of the textbook for your topics.

During the Exam, please

choose 2 topics from the seven topics listed at the bottom,

and, on separate sheets, **write a 250–400 word essay on each topic**.

You may not bring notes to the final. However, you are strongly encouraged to practice your planned essays. Feel free to get feedback on your practice essays from other students or faculty/friends/family/pets. Please think through, as a part of your preparation, how you will remember enough detail to recreate the essay during the Final Exam itself.

Note that 250 words equals 25 lines of text with 10 words on each line, or 42 lines with 6 words each. Thus each essay should be less than a page, or about a page if you write large. Use equations when appropriate to communicate meaning, but generally only a few equations are needed. Likewise one or two figures as needed. Most of your essays should be words.

Your goal is to write something like a short Wikipedia entry on the topic. For example, if someone searches `line search` in google, what they should get on the first screen? It might be something like your essay for topic ④ below. Said a different way, if a friend asks you to explain what a "line search" is, and why you would want to use one, what would be your 5 minute answer? A good 5 minute oral answer, perhaps augmented with scribbling on a whiteboard, should be converted here into a clean, clear, short essay.

For reference, the four paragraphs above this sentence total 247 words.

Seven Topics.

① How to move around a feasible set defined by linear equality and inequality constraints, including the ratio test.   *(section 3.1, especially pages 80, 81)*

② How to put linear programming (LP) problems in standard form.   *(section 4.2)*

③ The simplex method for LP problems in standard form.   *(sections 5.2–5.4)*

④ Back-tracking line search to generate sufficient decrease.   *(section 11.5)*

⑤ Steepest descent using exact line search on quadratic functions.   *(section 12.2)*

⑥ Quasi-Newton methods, with symmetric rank-one as the example.   *(section 12.3)*

⑦ Optimality conditions for nonlinear minimization subject to *linear* equality and inequality constraints.   *(sections 14.3-14.4)*