Math 466, Spring 2003

Project Writeup

Project papers are due on **Tuesday, June 10 by 5pm**; if you can hand yours in earlier than that, say, on the last day of class, that will give me more time to read it. If you cannot hand it in on the last day of class, then please bring it by my office, C-434 Padelford. I hope to devote the last day of class to project presentations. There also will be a chance to present preliminary results on Friday, May 30. If more people want to present results, then we could meet on Wed., June 11, during the time scheduled for our final exam (8:30–10:20).

The lengths of project papers may vary, but a substantial project will probably require about 8–10 pages of text, along with supporting figures, data, perhaps computer programs. If possible, use a word processor to type the paper; you can use any word processing system that you like, but one that is especially good for mathematical papers is `LaTeX`.

The exact format of your paper depends largely on the topic that you cover and what sorts of results you present. In general, however, papers should contain the following components:

- Abstract — this is a short overview of the paper, a miniature version of 100 words or so. Someone reading the abstract should get a good idea of what problem was tackled, what types of techniques were used to solve it, and what results were obtained. Most professional papers start with an abstract. It is very valuable for the potential reader, to help decide whether the paper is of interest and, if so, to get an overview of the whole picture before starting to read the details.

- Problem description — present the problem that you are attempting to solve or the algorithm that you are studying. Give some background for it, why it is important, etc. Explain what the questions are that you would like to be able to answer.

- Mathematical techniques — explain what mathematical or numerical techniques you use to solve the problem or to study the algorithm.

- Results — present the results of your computations or analysis. Interpret the results. Do they look reasonable? Have you checked the accuracy in some way? If so, explain how. If you want to attach code, put it at the end of the paper as an appendix. A line or two of code within the text might be appropriate, but generally a long piece of code detracts from the flow of the paper.

- Conclusions — summarize what you have done and what you have learned from it. Are there improvements that could be made in some future work?

- References — Please include a bibliography if you have used any references, e.g., books, journal articles, webpages. Put a citation in the paper if you refer to a reference. Remember to cite any references you used to learn about problems, or algorithms, or that were used to provide data. If you take a description directly from another source it should be in quotes and properly acknowledged. If you paraphrase it, you should still cite the reference.

Computer programs and computational results

If you have a computer program that generates a lot of output, explain any output. You should also be very selective in what you present. It is much more effective to have one or two pages of sample output which is selected to illustrate some key features (which are described and explained) than to have 20 pages of undigested numbers. Graphs or other visual presentations are often much better than lists of numbers.

Figuring out how best to present data is often an art in itself, and finding a good way can sometimes lead to new insights as well as making it easier for the reader to understand what you are saying.

Also, the output should be used as the basis for making conclusions, or for further analysis. The output should be presented to back up the conclusions, not as a replacement for making conclusions.

Computer programs that you write can be attached as an appendix at the end of the paper.