1 Extra Credit

This extra credit assignment will be worth up to 6% additional credit on your final grade. That 6% will be split between 20 pts on your exam 2 grade (3% of the final grade) and the class participation section of the course (the remaining 3%). This will be the last extra credit offered. To earn high amounts of extra credit for this assignment, you are expected to put forth a substantial effort. This assignment is completely voluntary.

You may choose one of the following two options for this project, which I have described below. You must submit a short paragraph to me by **Monday April 25, 2005 11:59 PM** stating the option and topic that you have chosen. I must approve every project. **I will not approve any late proposals!** If you choose to pursue a topic other than those listed below, you may do so, but it must fall within one of the two options. You should give me an extended proposal that demonstrates that you have an initial understanding of your topic, and that it is relevant to this course. You must submit your final report to me by **Monday May 16, 2005 11:59 PM IN HARD COPY**. **Any late submissions will receive zero points!** The standard homework late penalty does not apply to this assignment.

Option 1: Implementation Project

This option requires you to design an experiment, implement algorithms, and then evaluate those algorithms on a variety of test cases to verify some aspect of discrete math. Here are some suggested projects for this option:

- Implement 3 array sorting algorithms and verify the theoretical results for their best case, worst case, and average case complexity results. You should run the tests using at least 10 different array sizes and run each array size using 10 random arrays.
- Explore primality testing by implementing 2 primality testing algorithms, and comparing their performance on testing the primality of at least 50 randomly generated integers.
- Implement two cryptographic techniques, one of simple complexity (such as a transposition cipher), and one of of a much higher complexity (such as used in a modern cryptosystem).
- Implement code to solve the Tower of Hanoi.
- Verify the theoretical probability results for at least two probabilistic games by writing code to run these games. Report your results over at least 100 runs of each game. You cannot use the Monty Hall 3-door problem that we played in class.
- Implement both iterative and recursive versions of 2 algorithms. Compare the experimental run-time and memory-usage of each algorithm on at least 10 instances of each problem.
- Implement a finite-state automata as a pattern recognizer for a domain. Demonstrate that this automata correctly recognizes 10 non-trivial patterns in this domain.
- Implement two traveling salesman problem solvers and compare their performance on 5 instances of the problem.
- Implement a map-coloring system that determines the minimal number of colors needed to color a map so that no two adjacent regions use the same color. Demonstrate your system on five maps.
- Implement two methods for pseudo-random number generation and demonstrate that these methods do generate pseudo-random numbers.

For this option, you must submit a 2–3 page report written in LaTeX, your source code, and the run-time output for the results reported in the report.

Option 2: Research Project

For this option, you must write a short technical report on a topic in discrete math of interest. Your report may discuss important theorems and results, unsolved problems and techniques attempted on these problems, historical mathematicians and their contributions, or an important application of discrete math. Here are some suggested topics for this option:

- Alan Turing
- Godel's Incompleteness Theorem
- Magic Squares
- Lewis Carroll (wrote *Alice in Wonderland* in addition to being a symbolic logician)
- Huffman Codes
- Bayes' Theory
- Applications of Bayesian Networks to Medical Diagnosis or Spam Filtering
- RSA and DES cryptosystems (see Adi Shamir)
- Euclid
- Automated Theorem Proving
- NP-completeness
- Fuzzy Set Theory

For this option, you must submit a 6–8 page report written in LaTeX, with at least 6 non-web references. The report must include mathematical content (e.g. equations).

The amount of extra credit earned for this assignment will be based on the amount of effort that the student puts into it. The general guidelines for evaluation is:

- Effort, Detail, and Thoroughness 50%
- Relevance and Significance to the course 20%
- Presentation, Writing Style, Clarity, Creativity 30%

You may work on this assignment using any resources you like (including your textbook, websites, papers, etc.), but you must use proper citations (in either APA, MLA, or another established format). Be VERY careful to avoid plagiarism. If you have any questions, please come and talk to me. A good rule of thumb is that if you use more than two words in a row from another author, you should cite the source. Changing the word order or a few words from another author's sentence does not make it your own or a paraphrase of the original work. To paraphrase properly, read the section you wish to paraphrase, put the book down, and then write the ideas in your own words. You must still cite the original author when you paraphrase.

Please typeset your report using a 12 point font. Your report should be grammatically correct. I refer you to the UMBC Writing Center for help, if you need it. Even senior English majors with high GPAs use the writing center. Even computer scientists must write!

If you have any questions, please ask!