Math 788, Sect 001: Computational Number Theory

The course will be an introductory course centered around various aspects of computational number theory. In particular, the course will be accessible to beginning graduate students and will serve as an introduction into some topics in elementary number theory.

Initially, we will begin with running time estimates for basic arithmetical operations on integers such as addition, multiplication and greatest common divisors. Then we will move to some more substantial topics, such as primality testing (including the polynomial time algorithm given by Manindra Agrawal, Neeraj Kayal and Nitin Saxena in their 2004 Annals paper), factoring of integers, and the corresponding problems for polynomials. We will also discuss private-key encryption, discrete Fourier transforms and lattice basis reduction (the latter being an important tool for attacking many computational questions). There will be no requirement to use the computer.

Some tidbits from this course include:

- The largest known prime is $2^{57885161} - 1$ and contains 17425170 digits. We will discuss how the prime was found.
- We will discuss how you can communicate privately with someone in the personals even though you have never met or even communicated with the person before. In other words, we will discuss how one can publish, for the whole world to see, a coded message to someone else, and how the other person who might be a complete stranger can be the only one who can understand the message.
- We will discuss what is behind some methods used to factor large integers, say with about 140 digits.
- It is not known whether it is possible to factor an integer in polynomial time. We will discuss what this means. On the other hand, it is possible to factor polynomials (i.e., to express a polynomial with integer coefficients as a product of irreducible polynomials with integer coefficients) in polynomial time. We will discuss the proof of this result. The proof is based on an important algorithmic method (called the L^3 algorithm or the lattice basis reduction algorithm) developed by Arjen Lenstra, Hendrik Lenstra, Jr., and Laszlo Lovász.
- Despite the above, Maple and other similar software use an older algorithm, due to Hans Zassenhaus, for factoring polynomials which is not a polynomial time algorithm. Why? Because it's usually faster. We will explain what this means and how the algorithm works.

Some Particulars:

Instructor: Michael Filaseta (Spring, 2014)

Textbook: No textbook is required. Class notes are currently available on-line at

`http://www.math.sc.edu/~filaseta/gradcourses/Math788Mnotes.pdf`

and these will be updated with new material.

Grading: The tentative plans are to base the grade on homework (50%), one in-class test (20%) and a final exam (30%). Students are encouraged to work together on homework, but as a rule the solutions to assignments should be written up independently. As long as the instructor is convinced that this rule is being followed, the final exam will be optional and will not lower the pre-final exam grade.