Literature review and project suggestions 2013
Jonathan Deane

I am willing to supervise general computing projects based on your own ideas: come and talk to me if you have an idea that you would like to discuss.

1 Literature Reviews

1.1 Chebyshev's Theorem and Chebyshev polynomials

Chebyshev's theorem states
Let $p(x)$ be a polynomial of degree $n \geq 1$ with leading coefficient (i.e., coefficient of x^n) $= 1$. Then the maximum value of $|p(x)|$ over the range $x = [-1, 1] \geq 1/2^{(n-1)}$.
You will first need to understand a proof of this theorem. There is a set of orthogonal polynomials, known as the Chebyshev polynomials, for which the maximum value exactly equals $1/2^{(n-1)}$, and these polynomials have many other interesting properties as well, a selection of which you will investigate.

1.2 Irrationality Proofs

Aim: to understand a variety of proofs of irrationality. One possible objective might be presenting a proof of the irrationality of π The literature review might also contain proofs of irrationality of other numbers, e.g. e, square roots of rational numbers that are not squares.

1.3 Probabilistic Primality Testing

It is possible to test whether a number N is prime or not by testing whether it is divisible by any of the primes less than square root of N. For large numbers, this obviously takes an inordinate amount of time. There are many other tests that can be used, and one at least involves probabilistic methods that gives an answer with a probability close to, but not equal to 1. To find out how this is done, the method's limitations, and why primality testing is important in practice (e.g. in relation to 'uncrackable' codes) is the subject of this literature review.

1.4 The workings of RSA cryptography

The aim of this literature review is to understand, and be able to present, the details of the working of this important cryptography algorithm. RSA Cryptography is based on various theorems from Number Theory, and these need to be proved before the algorithm can be properly understood.

2 Projects

2.1 Teaching software for the WWW (ONE project only)

If you know about at least one of html, Java, cgi you might consider writing some teaching software for the Web. There is a lot of scope for good Web page design and there is a free

choice about the area of the course you choose to cover.
Standard: Anything from easy to difficult. The perfect example of an 'elastic' project. Students choosing this need to consider carefully how to make it of a sufficiently high technical level.

2.2 WWW-accessible computation software

It is desired to make available via the WWW a program, written in C, that does a numerical evaluation of an integral, for given sets of input parameters. The C program is already written; the project is to interface this with the WWW, possibly by machine translating it into Java — or in any other practicable way. Knowledge of WWW technologies required.

2.3 An analysis of 'Aces up'

Aces up is a very simple game of patience, using a single standard pack of cards. The rules can be found at `http://www.kmgassociates.com/games/acesup/`. Experience (i.e. playing it a lot) gives the impression that the probability of succeeding at the game is something of the order of 1/50. The objective of this project is to estimate this probability by Monte Carlo simulation, and determine whether there is any strategy that can increase this probability.
Standard: Quite challenging. Needs good programming skills (preferably in C) and a little elementary probability theory.

2.4 Finding poles in the complex time domain solution of Duffing's equation

Duffing's equation $y'' + ky' + By(y^2 - 1) = A \sin \omega z$, where primes denote derivatives with respect to z, is a well-known nonlinear differential equation whose solutions (for real time, $z = t$) display chaotic behaviour for some values of the parameters. The objective is to write a C program to solve this equation for complex time, z. There are poles off the real axis in the solution of the equation, which are (a) movable (their position depends on the initial conditions) and (b) conjectured to be simple, i.e. of the form $a/(z - z_0)$. The project will be to write software to locate some of these these poles and find their orders and residues using a ratio test. The challenging part of the project is to carry this out as automatically as possible — that is, without human intervention.
Standard: A very challenging project. Needs C programming skills and understanding of some topics in the year 3 complex analysis course (the Cauchy-Riemann conditions and Cauchy-Goursat theorem for instance).