Call out

If you are interested in any of the three projects I describe below for the summer 2013 Charlotte Research Scholars program, please contact me to discuss your interest. I can explicitly suggest your name if we agree there is a match.

Project 1:

Title:

Exploring Conformational Pathways in Myosin V Using Free Energy Driven Geometrical Simulation

Abstract:

Myosin V is a molecular motor protein that has a primary function to pull cargo and organelles along actin filaments within cells. Despite decades of experimental studies, a complete picture of the molecular mechanisms responsible for the conversion of chemical energy to mechanical energy remains elusive. Molecular simulation can play an important role for identifying molecular mechanisms that experimental methods cannot resolve. However, due to the large size of the myosin protein, standard molecular dynamics simulation, due to computational inefficiencies, is not the best approach to monitor large-scale dynamics. This work will computationally explore large-scale conformational pathways in myosin V using free energy driven geometrical simulation at the all atom level. Principal component analysis will be applied to the conformational ensemble to quantify the essential dynamics. The computational tools needed to generate and analyze the data are available that include both commercial and special methods that have been developed at UNC Charlotte. Preliminary work on conformational pathways in myosin V has been done previously, which lead to developing a better method, for which this project will employ for the first time. As such, the results of this project will have to be benchmarked, and the model parameters will need to be optimized. The expected results of the simulation, after benchmarking, will give an unprecedented view of the conformational fluctuations that myosin experiences in its monomer form in solution, for which experimental FRET data is available to compare against. This computational project is expected to yield insight into the molecular mechanisms governing energy transduction by elucidating the relationships between thermodynamic stability, mechanical stability and dynamics. This project is ideal for students interested in Biophysics, Computational Physics, Computational Chemistry or Applied Statistics.

Minimum Qualifications:

A desire to learn about molecular simulations and statistical analysis is necessary. Programing experience or performing data analysis using software packages of any type would be desirable.

Project 2:

Title:

Building Software Engineering Skills: Preparing Scientific Software for Public Domain

Abstract:

FAST software developed at UNC Charlotte provides a Flexibility And Stability Test on protein structures, which consists of a well-organized C++ library with 1,909 classes, extensive metadata involving 1,350 flat-files that define a model parameter database, and 13 XML files that control operational functionality. *FAST* supports a computational platform (computing engine) for high throughput Pharmaceutical Chemistry applications related to protein design and drug discovery. There are more than 110,000 lines of C++ code that takes advantage of the object-oriented power of C++ by extensively using the standard template library, boost library, BLAS and LAPACK++ libraries, an open source XML reader and MPI for parallel computing. The on-going *FAST* software project adheres to being platform independent. Nevertheless, *FAST* has been developed, tested and run only on LINUX and OSX systems under GNU to date. As the project has evolved starting from day 1 with 0 lines of code, to the size it is now, and still growing, the lack of documentation and lack of a convenient debugging logger and test-code schema has limited the ability for new developers and client-programmers to use the library without going mad in the process. There are several goals of this project ranked in order of greatest priority as: 1) Use Doxygen to create detailed documentation for classes and class-members; 2) develop best practices for code documentation using the most critical classes as examples; 3) implement an extensible debugging logger functionality; 4) fix library linkage problems when switching between LINUX and OSX; 5) set up configuration files to facilitate a platform independent software distribution; 6) replace the currently employed Makefile approach to compile the *FAST* library with SCons (Python based); 7) Compile *FAST* on a PC. This project is ideal for a Computer Science student interested in becoming a Software Engineer, looking for a real-life large-scale project to hone on skills.

Minimum Qualifications:

In order of importance: Good working knowledge of LINUX, GNU g++ compiler, object oriented programming and C++, and, familiarity with Makefiles, linking libraries, SCons, MPI and Doxygen is desirable.

Project 3:

Title:

Analysis of Income Tax Burden Relative to the Distribution of Wealth in Society

Abstract:

Based on IRS data from 1999 to 2009, the bottom half of income earners have paid no more than 4.00% of the total tax revenue, and no less than 2.25%, while the top 1% of income earners have paid no more than 40.42% of the total tax revenue, and no less than 33.71%. Do these numbers seem fair? In this project, an approach called the “spherical cow approximation” will be borrowed from theoretical physics to answer the question: What is the tax burden for taxpayers across the spectrum of income levels? In physics, toy models are constructed to focus on essential aspects of a problem to identify cause and effect while eliminating superficial complications that do not change key characteristics of the system. In this spirit, a tunable model has been devised to describe the distribution of wealth in society, characterized by mean income, poverty line and income gap ratio comparing top earners to mean income. A tunable model of taxation has been constructed characterized by an average tax rate, degree of tax progression, and distribution of deductions. By adjusting model variables the tax burden across income levels will be quantified for different scenarios. Furthermore, total spending by the government to subsidize low-income non-tax paying populations will be included to arrive at graphs showing expense (in addition to revenue) versus income level. The question of whether the tax burden stratefied by income is fair, will, fortunately, remain in the eyes of the beholder. This project does not answer whether a taxpayer in one income level should pay a greater or lesser tax rate than another taxpayer in a different income level. Rather, the results are expected to be important because the non-biased nature of the model will provide a quantified baseline so that informed decisions can be made about tax rates, spending and balanced budgets, all of which have recently been given much attention.

Minimum Qualifications:

Knowledge of differential equations, statistics and programing is expected. This project is ideal for someone interested in finance, economics, applied mathematics, computer science or physics.