ENGINEERING STUDENTS!
SCIENCE STUDENTS!
MATHEMATICAL AND COMPUTER SCIENCE STUDENTS!

Are you looking for something $\mathcal{NEW}$ to Study?

Are you interested in $\mathcal{NEW}$ computer skills?

Then **Applied Computational Science** Is For YOU!

The "Grand Challenges" are problems which are considered to be pivotal to the economy of the United States. They are problems like global warming, pollution, the structure of the human genome. These problems are approachable only through the use of supercomputers. The challenge of "computational science" is to help develop solutions to these problems.

If you were ever interested in the sciences or mathematics and want to understand how to put tomorrow's supercomputers to work, then this course is for you.

"Applied Computational Science" is an interdisciplinary course which draws students from all over campus. You will have a chance to learn about scientific programming and contribute your skills in new and important ways. You work with others in the same way you can expect to work in industry. You can learn much about science and a whole lot about algorithms and new architectures.

The prerequisites are minimal: your calculus sequence and your programming skills. You will work in interdisciplinary teams—just as you might expect to in industry. Your own special skills are needed by the other members of your team.

Become part of this new and exciting study. Sign-up during pre-registration. Four course titles are available. Scheduling details on reverse side.

MTHSC 481	Mathematical Sciences
CPSC 481	Undergraduates
CPSC 681	General Graduate Students
CPSC 881	Special Permission Graduate Students(Permission required!)

For further information, see the reverse side. For even more information, contact Professor Stevenson, 656-5880 or e-mail to "steve@cs".

Syllabus for Applied Computational Science

D. E. Stevenson
Computer Science
656-5880
steve@cs.clemson.edu

D. D. Warner
Mathematical Sciences
656-5244
warner@math.clemson.edu

Module 1. $1, 2, 3 \ldots \infty$? *Exploring the floating point number system.* We look at various properties of the *IEEE* floating point model and measure the machine parameters for several machines. Programming includes examples which demonstrate roundoff and cancellation ideas.

Module 2. *The Railroad Joker.* This problem explores the use of approximations by taking a simple problem with an obvious analytic solution and attempting to compute a good answer. It is not as easy as it looks.

Module 3. *Hot Rod.* In order to acquaint the student with finite element methods and numeric linear algebra, we turn to the problem of temperature distribution in a rod heated at one end.

Module 4. *The Roller Coaster.* What is the fastest path between two points in a plane for a particle *under the influence of gravity* ? It's not a straight line, it is the brachistochrone. We use a simulated annealing approach to find the correct path.

Module 5. *Adam Smith's Revenge.* We study mass action models common in chemistry and other population oriented problems by looking at the predator-prey models.

For the laboratories—

Lab 1. *Networks and Remote Computing.* Introduction to networks, electronic mail, file transfer techniques, remote login, *etc.* Basic net-iquette. Basic ideas on how to do remote computing. Use of *NCSA* Crays.

Lab 2. *Parallel Compilation Procedures.* Explanation of the use of the directives. How to interpret the output from vectoring compilers. Principles of designing and developing parallel code. Lab: Vectorizing a simple code.

Lab 3. *How to Benchmark.* Lab: A simple benchmark project.

Lab 4. *Symbolic Algebra. Mathematica* and *Maple.* Lab: use of symbolic algebra in application context.

Lab 5. *Distributed Computing.* Lab: Porting a sequential code to the Computer Science hypercube.

Sign up for *CPSC 481* or *CPSC 681* or *MTHSC 481*. *800*-level credit possible. Contact instructors.

Course Time: Planned for 2:30-4:00, *MW.* Course Location: *TBA*

Text: Instructor's notes distributed in class and available from Campus Copy Shop.