CSCI 597 Introduction to Computational Sciences

INSTRUCTOR:

Derek Harter, Ph.D., Associate Professor, Department of Computer Science
Texas A&M University - Commerce
Office: Jour 208
E-mail: Derek.Harter@tamuc.edu
Office Hours: T, W, Th 1:30pm - 5:00pm or by appointment.

CLASS MEETINGS:

Online lectures and materials through eCollege.

01W	81776	Web Based Class

DESCRIPTION:

Big scientific data sets are growing exponentially both in size and complexity. Extracting meaningful information from this data requires not only programming skills, but also understanding the analysis work-flows and mathematical models and visualization tools that help to condense large amounts of information into a comprehensible story. Computational science is the scientific investigation of problems through modeling, simulation and analysis of physical processes on a computer. Computational science is now considered by most scientists to be on par with the development of scientific theory and the use of experimentation in order to understand more about our world. Computational science is *not* the same as computer science. Rather, it is an interdisciplinary blend of scientific models, applied mathematics, computational techniques, and practices. This Introduction to Computational Science course focuses upon *simple* and *intuitive* computational models and methods.

REQUIREMENTS AND OBJECTIVES:

This course forms one of the core subjects in the new master's degree program in the Computational Sciences, as well as being a necessary topic for any future Ph.D. program offerings that is being developed and proposed by the computer science department. Computational sciences differs from the traditional computer science discipline in several ways, but most importantly as being focused on applying computational methods to solving large scientific problems. Thus this type of scientific data analysis of large complex data sets is both increasingly crucial to scientific research, as well as being in great demand for practitioners who can apply computational analysis and modeling to such data sets. This course directly addresses this area, and forms a cornerstone subject for any student wishing to understand and practice computational science research.

Goals include:

- Understand the scientific process and the philosophy of science.

- Understand the purpose and value of computational science.
- Learn to use computer-aided problem solving and visualization techniques.
- Be exposed to the common tools and practices of working computational scientists.

COMPANION TEXTBOOKS/READINGS:

[PCS] Python Scripting for Computational Science by Hans Peter Langtangen (3rd Edition)

PREREQUISITES:

Successful enrollment in Computational Sciences Master's Program. This course assumes a basic proficiency with computer programming and computational concepts. A basic understanding of how a sequential computer works is assumed. Fundamental familiarity with basic CS topics such as algorithms and analysis, data types, data storage, I/O, loops, branches, subprograms and object-oriented programming are also assumed. Fundamental mathematical ability is assumed, such as familiarity with discrete and continuous mathematical models. Enough ability to use vectors, matrices, integration and differentiation. The REAL prerequisite is a desire to learn and explore new ideas.

EVALUATION:

Your grade for the course will be based on the following (approximate) percentages:

Two Exams: 50%
Labs / Programming Assignments (appx 4-6): 40%
Quizzes and eCollege participation: 10%

STUDENTS WITH DISABILITIES ACT COMPLIANCE:

The Americans with Disabilities Act (ADA) is a federal anti-discrimination statute that provides comprehensive civil rights protection for persons with disabilities. Among other things, this legislation requires that all students with disabilities be guaranteed a learning environment that provides for reasonable accommodation of their disabilities. If you have a disability requiring an accommodation, please contact: Office of Student Disability Resources and Services, Texas A&M University-Commerce, Gee Library, Room 132, Phone (903) 886-5150, StudentDisabilityServices@tamuc.edu

ACADEMIC ETHICS:

"All students enrolled at the University shall follow the tenets of common decency and acceptable behavior conducive to a positive learning environment." (See Student's Guide Handbook, Policies and Procedures, Conduct).

Ethics also includes the issue of plagiarism, and copying code for programming assignments is just as serious as any other type of plagiarism. If you are caught sharing or using other people's work in this class, you will receive a 0 grade and a warning on the first instance. A subsequent instance will result in receiving an F grade for the course, and possible disciplinary proceedings.

ATTENDANCE POLICY:

Students are expected to follow all instructions and visit eCollege regularly many times weekly to complete the materials for this online course. If a student is unable to submit assignments by the due date for the assignment, they are expected to make alternative arrangements to assure that the assignment is turned in ON TIME, before the assignment is actually due. Any student wishing to withdraw from the course must do so officially as outlined in the class schedule. THE INSTRUCTOR CANNOT DROP OR WITHDRAW ANY STUDENT.

COURSE REQUIREMENT DEADLINES:

Credit will be given for ONLY those exams, programs, and/or projects turned in no later than the deadline as announced by the instructor of this class, unless prior arrangement has been made with the instructor.

TENTATIVE SCHEDULE

Wk	Date	Topic / Activity	Notebook	Assg
1	08/25	Course Introduction		
		Set up Python	01 – Scientifc Computing w/ Python	
2	09/01	Introduction to Python	02a – Intro to Python	#1
			02b – More Python Programming	
3	09/08	NumPy – Python Numerical Computing Library	03 - NumPy	#2
4	09/15	SciPy – Python scientific library	04 - Scipy	
5	09/22	Matplotlib – Python graphing and visualization Library	05 - Matplotlib	
6	9/29	Computational Sciences Tools	06 - Tools	#3
7	10/06	Vectorizing numerical algorithms w/ NumPy		
		Exam 1		
8	10/13	Random Numbers and Monte Carlo Sims	07 – part A	
9	10/20	Random Numbers and Monte Carlo Sims	07 – part B	#4
10	10/27	Sequences and Difference Equations	08 – part A	
11	11/03	Sequences and Difference Equations	08 – part b	#5
12	11/10	Discrete Grid Space Models	09 – part A	
13	11/17	Discrete Grid Space Models Example: Ccomputing w/ Formulas	09 – part B	
14	11/24	Example: computing w/ formulas, curve fitting		
		Thanksgiving		
15	12/01	High Performance Computing		
		Reproducible Research & Literate Programming		
Fnl	12/08 – 12/13	**Exam 2**		

STUDENT LEARNING OUTCOMES

1. Learn basic scientific modeling paradigms, discrete and stochastic models.
2. Apply computational techniques to tackling scientific research questions.
3. Familiarize with standard tools of computational science: HPC, R/Python/Numpy/Scipy environment.

Learning outcomes will be measured through mapping assignment and test questions to specific outcome items, as well as through exit surveys of student experiences with the outcome familarity.