CPS 5401

Introduction to Computational Science

Fall 2014

The University of Texas at El Paso

Shirley Moore

http://www.cs.utep.edu/svmoore/

http://svmoore.pbworks.com/

CPS 5401 is an introduction to basic computational science skills including UNIX, scientific programming using high level languages, parallel computer architectures, parallel programming paradigms, and numerical libraries.

Course #:	CPS 5401
Course title:	Introduction to Computational Science
Credit hours:	4
Term:	Fall 2014
Time & location:	Mondays and Wednesdays, 4:30-6:30pm, CCSB 1.0510
Prerequisites:	Instructor approval
Course fee:	None
Instructor:	Shirley Moore
Office location:	Chemistry and Computer Science Building 3.0422
Contact information:	Office phone: 915-747-5883 Email: svmoore@utep.edu Fax: 915-747-5030 WWW: http://www.cs.utep.edu/svmoore/
Office hours:	3:00-4:00pm Monday and Wednesday or by appointment
Teaching assistant:	Henry Moncada
TA office location:	CCSB 3.1202H
TA email:	hrmoncadalopez@miners.utep.edu
TA office hours:	
Online textbook:	Victor Eijkhout, *Introduction to High-performance Scientific Computing,* http://tacc-web.austin.utexas.edu/veijkhout/public_html/istc/istc.html

Course Objectives

The course will cover three major aspects of computational science in three parts:

- Part I will consist of a practical introduction to UNIX, scientific programming using high level languages, parallel computer architectures, parallel programming models, and current trends in high performance computing.
- Part II will cover performance and scalability issues, performance evaluation and optimization, and a more detailed treatment of parallelization issues, including hybrid programming models and locality optimization.
- Part III will cover broadly useful numerical algorithms and their implementation in parallel libraries including dense and sparse linear algebra, FFTs, and differential equations.

Grading

Your grade for the course will be based on the following:

- 20% homework
- 30% labs
- 40% midterm and final exams
- 10% class preparation and participation

Attendance Policy

This is a challenging course and attendance is essential for success. Although attendance will not be taken, please try not to be absent unless absolutely necessary.

Accommodations for Students with Disabilities

If you have a disability and need classroom accommodations, please contact The Center for Accommodations and Support Services (CASS) at 747-5148, or by email to cass@utep.edu, or visit their office located in UTEP Union East, Room 106. For additional information, please visit the CASS website at www.sa.utep.edu/cass.

Academic Honesty Policy

Make sure you understand the UTEP academic honesty policy. Students are encouraged to share ideas, but you must do your own homework and you must write your own code for the projects (you may copy code that is on the course website). If homework or program code is suspected of being duplicated or copied, you will receive an incomplete for the assignment, and your case will be referred to the Dean of Students for adjudication. If the instructor has reason to believe that you have cheated on a quiz or exam, your case will be referred to the Dean of Students for adjudication.

Course Format

Each class will consist of an hour of lecture and an hour of practical hands-on laboratory exercises. The lab exercises will make use of the Linux lab machines and the UTEP Research Cloud. Students should be able to login to these resources remotely from a home or office computer.

Course Topics

- UNIX
  - Logging in and account setup
  - Shell and environment variables
  - File system
  - Job control
  - Shell programming
  - Makefiles
  - Source code control
- Scientific programming languages
  - General concepts
  - Fortran 90/95
  - C and C++

    - Python
- Computer architecture
    - Cache-based microprocessors
    - Memory hierarchies
    - Multicore processors
    - Shared memory computers
    - Distributed memory computers
    - Hierarchical (hybrid) systems
    - GPGPUs
    - Networks
- Parallel programming paradigms
    - Data and task parallelism
    - Shared memory parallel programming with OpenMP
    - Distributed memory parallel programming with MPI
    - Hybrid parallelization with MPI and OpenMP
    - PGAS languages
    - GPU programming models
- Performance optimization
    - Single node profiling and optimization
    - Parallel profiling and tracing
    - Data access optimization
    - Efficient OpenMP programming
    - Locality optimizations on ccNUMA architectures
    - Efficient MPI programming
    - Topology and affinity in multicore environments
- Scalability
    - Scalability and efficiency metrics
    - Strong vs. weak scalability
    - Communication models
- Dense and sparse linear algebra libraries
    - Vector and matrix products
    - LU factorization
    - Cholesky factorization
    - Direct sparse methods
    - Iterative methods
- Fast Fourier transform
- Numerical optimization
- Ordinary and partial differential equations