COMP/CS 605 Introduction to Parallel Computing

1. **Course number and name**: COMP/CS 605 Introduction to Parallel Computing

2. **Credits and contact hours**: 3 credits, 6 contact hours / week.

3. **Instructor’s or course coordinator’s name, office hours, and contact information**: Mary Thomas, 12-1, and 1-2 M/W, Room GMCS 562, mthomas@mail.sdsu.edu, phone: 4-1694.

4. **Required materials to include**:
      Amazon link: Pacheco (2011)
   c. Designing and Building Parallel Programs, by Ian Foster. Online version: http://www.mcs.anl.gov/~itf/dbpp/
   d. GPU/CUDA Programming: CUDA By Example, by Sanders and Kandrot.
   e. Relevant online tutorials, papers, code examples and videos.

5. **Specific course information**
   a. **Brief description**: This class will provide an overview of CPU, Pthreads, OpenMP, and GPU parallel computing approaches, architectures, and programming models used to solve scientific and engineering problems. Modules for the course will include: the basic concepts of parallel programming; hardware and software architectures; optimization methods; and several different parallel models. These models include the Message Passing Interface (MPI), Pthreads, OpenMP, and CUDA/GPU libraries. Time permitting, this course will also survey advanced topics as time permits grid computing, mobile computing, cloud computing, hybrid programming, etc.

   b. **Prerequisites or co-requisites**: Knowledge of the FORTRAN or C programming languages. CS 501, 520, 525, 520, COMP 526, or equivalent Unix OS experience, Graduate standing or senior standing in Computational Science, Computer Science, or Engineering or instructor's approval.

6. **Specific goals for the course**
   a. (1) The student will be able to articulate an understanding of the fundamentals of parallel and distributed computing including parallel/distributed architectures and paradigms. (2) The students will develop an ability to design, code, and execute basic parallel applications and algorithms. (3) Though hands-on experience, students will develop the ability to work with a diverse set of computer software and systems (CPU, GPU) including laptops, workstations, small and high-end parallel systems.
b. Mapping to Program Course Outcomes: Explicitly indicate which of the CS program student outcomes (listed on the next page) are addressed by the course.

This course addresses the following student outcomes:

- An ability to apply knowledge of computing and mathematics appropriate to the program’s student outcomes and to the discipline
- An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
- An ability to function effectively on teams to accomplish a common goal
- An ability to use current techniques, skills, and tools necessary for computing practice.

7. Brief list of topics to be covered: The course consists of the following modules:
   
   I. Introduction to Parallel Computing/Unix/SciComp Basics
   II. Distributed Computing with Message Passing Interface.
   III. SciComp Basics: Performance, benchmarking, and data analysis.
   IV. Shared Memory Programming with Pthreads and OpenMP.
   V. Cuda/GPU Programming.