**ABET Course Syllabus – CS4075**

|  |  |  |  |
| --- | --- | --- | --- |
| **Code** | CS4075 | **Credits** | 3 |
| **Title** | Concurrent and Distributed Programming | **Coordinator** | Yuqing Zhu |

**Course Information**

1. **Catalog Description:** Parallel Programming techniques; abstract models of hardware and operating systems to support parallel programs; multiple models of concurrency; their advantages and disadvantages.
2. **Prerequisites:** CS 3112, CS3035
3. **Contact Hours:** 3 hours/week
4. **Required/Elective:** This course is an elective course in the BS program.

**Textbook**

An Introduction to Parallel Computing. By: Peter S. Pacheco. Elsevier Press.

**Course Goals**

The Student Learning Outcomes that are addressed by the course are:

*SLO #3. Students will have a strong foundation in the design, analysis, and application of many types of algorithms.*

*SLO #5. Students will have the training to analyze problems and identify and define the computing requirements appropriate to their solutions.*

*SLO #6. Students will have the training to design, implement, and evaluate large software systems working both individually and collaboratively.*

Other outcomes of instruction: At the end of the course students are able to:

* Know the basic parallel hardware and the parallel software design methodology.
* Implement efficient parallel programs for some sample problems..
* Be very familiar with using MPI, Pthreads, and OpenMP - three of the most widely used APIs for parallel programming to write parallel programs.

**Topics Covered**

* Neuman architecture
* Parallel hardware
  + SIMD
  + MIMD
* Parellel software
  + Caveats
  + shared-memory
  + distributed memory
* Parallel program design with Foster's methodology.
* Distributed-memory programming with MPI.
* The Trapezoidal rule in MPI
* I/O with MPI
* Collective Communication in MPI
* Performance evaluation of MPI programs
* Parallel sorting example using MPI.
* Shared-memory programming with Pthreads
* Matrix-Vector Multiplication
* Producer-consumer synchronization
* Caches and Cache Coherence
* False Sharing
* Thread-Safety.
* Shared-memory programming with OpenMP
* Scope of Variables
* Reduction Clause
* The parallel for Directive
* Loops in OpenMP example: Sorting, Scheduling Loops.
* Parallel program design samples
  + Two n-Body Solves using OpenMP and MPI respectively
  + Tree Search: recursive version and parallelized version, implementation with OpenMP and MPI respectively.