Knight Foundation School of Computing and Information Sciences

Course Title: Applied Parallel Computing **Date:** 10/16/2019

Course Number: COT 5432

Number of Credits: 3

Subject Area: Computer Applications	**Subject Area Coordinator:** **email:**
Catalog Description: This course teaches advance undergrad and graduate students to solve problems from scientific, social and financial domains using parallel computing principles and techniques.	
Textbook: None. We will distribute reading materials throughout the semester.	
References: 1) Petascale Computing: Algorithms and Applications, Edited by David A. Bader, Chapman & Hall/CRC Computational Science Series, 2007 2) Big Data: Algorithms, Analytics, and Applications; ISBN 9781482240559 3) Multicore Computing: Algorithms, Architectures, and Applications (Chapman & Hall/CRC Computer and Information Science Series) 1st Edition 4) Techniques and Environments for Big Data Analysis: Parallel, Cloud, and Grid Computing (Studies in Big Data) 1st ed. 2016 Edition 5) Networking for Big Data (Chapman & Hall/CRC Big Data Series) Hardcover August 3, 2015	
Prerequisite Courses: COP 3530 and (CDA 3102 or CDA 4101 or EEL 4709)	
Corequisite Courses: None	

Type: **Elective**

Audience: Both undergraduate and graduate student will be exposed to the course materials listed in the course outline. The graduate students will have additional work assigned to them such as reading additional materials, research papers and executing a small project in the context of the course.

Prerequisite Topics:

- Calculus, Basic Programming, and Data Structures

Course Outcomes:

Students who successfully complete this course will be able to:

1. Describe a selection of concepts, algorithms, and models used in parallel computing for solving complex real-world problems.
2. Identify a class of parallel computing techniques and algorithms that might be applied to a specific task for a given parallel architecture

3. Explain the basic understanding of sources of parallelism and locality and how to identify enough parallelism and reduction in data movement costs for given parallel architectures
4. Identify, understand, and apply the concept of programming "patterns" and how to identify the "7 basic dwarfs" of high-performance computing and how these dwarfs can be parallelized for big data problems in manufacturing, medicine, scientific and financial domains
5. Identify and explain the metrics for measuring performance and be able to identify bottlenecks in parallel algorithms and architectures and be able to describe dynamic and static load balancing techniques for a given architecture.

Outline:

Topic	***Hours (Total: 37.5 hours = 15 weeks * 2 lectures/week * 1.25 hrs/lecture)***	**Outcome**
1. Introduction: Why Parallel Computing?	**1**	***1,2***
2. Single processor machines: Memory hierarchies and processor features	**1**	***1,2,3***
3. Introduction to parallel machines and programming models	**1**	***1,2,3***
4. Sources of parallelism and locality in simulation and Algorithms: part 1	**1**	***1,2,3,4***
5. Sources of parallelism and locality in simulation and Algorithms: part 2	**1**	***1,2,3,4***
6. Shared memory machines and programming: OpenMP and Threads	**1**	***1,2,3,4,5***
7. Distributed memory machines and programming in MPI	**2**	***1,2,3,4,5***
8. Partitioned Global Address Space Programming with Unified Parallel C UPC	**1**	***1,2,3,4,5***
9. GPUs, and programming with CUDA and OpenCL	**1**	***1,2,3,4,5***
10. Performance and Debugging Tools	**1**	***3,4,5***
11. Dense Linear Algebra Algorithms: Part 1	**1**	***1,2,3,4,5***
12. Dense Linear Algebra: Part 2	**1**	***1,2,3,4,5***
13. Graph Partitioning: Part 1	**1**	***1,4,5,***
14. Graph Partitioning: Part 2	**1**	***1,4,5***
15. Sparse-Matrix-Vector-Multiply: Part 1 and 2	**2**	***1,3,4,5***
16. Particle (N-Body) methods: Efficient Data Race Detection for Distributed Memory Parallel Programs	**1**	***1,2,3,4,5***
17. Structured grids and multigrid	**1**	***4,5***

18. Cloud computing paradigms with MapReduce and Hadoop	**1**	***3,4,5***
19. Patterns of Parallel Programming	**2**	***4,5***
20. Structured grids	**1**	***4,5***
21. Exascale Computing	**1**	***1,2,3,4,5***
22. Parallel Graph Algorithms	**1**	***1,2,3,4,5***
23. Parallel Climate Modeling	**1**	***1,2,3,4,5***
24. Parallel Fast Fourier Transform (FFT)	**1**	***1,2,3,4,5***
25. Static and Dynamic Load Balancing	**1**	***1,2,3,4,5***
26. Parallel Algorithms and Architectures for Machine-Learning and Deep-Learning Solutions	**2**	***1,2,3,4,5***

Assessment Plan for the Course & how Data in the Course are used to assess Program Outcomes

Student and Instructor Course Outcome Surveys are administered at the conclusion of each offering, and are evaluated as described in the School's Assessment Plan: https://abet.cs.fiu.edu/csassessment/