Course Specifications

General Information

Course name	Parallel Processing
Course number	ITCS4348
Faculty	
Department	
Course type	Major Needs
Course level	4
Credit hours (theoretical)	3
Credit hours (practical)	0
Course Prerequisites	

Course Objectives

1 - To design, implement, test and debug a parallel application program using MPI

2 - To design, implement, test and debug a parallel application program using OpenMP

3 - To parallelize an existing application using an appropriate parallel programming paradigm

4 - To explain, in writing, the tradeoffs that result from using a specific programming paradigm for a given problem class

5 - To develop and analyze a parallel algorithm using the PRAM model

Intended Learning Outcomes

Knowledge and Understanding	* a1 An ability to apply knowledge of computing and mathematics appropriate to the discipline * a2 An ability to analyze a problem and identify the computing requirements appropriate for its solution; an ability to design, implement and evaluate a computer based system, process, component or program to meet desire needs * a3 An ability to apply mathematical foundations, algorithmic principles and computer science theory to the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices * a4 An ability to apply design and development principles in the construction of software systems of varying complexity * a5 An ability to function effectively as a member of a team in order to accomplish a common goal
Intellectual Skills	* b1 Design, implement, test and debug a parallel application program using MPI * b2 Design, implement, test and debug a parallel application program using OpenMP * b3 Parallelize an existing application using an appropriate parallel programming paradigm * b4 Explain, in writing, the tradeoffs that result from using a specific programming paradigm for a given problem class * b5 Develop and analyze a parallel algorithm using the PRAM model
Professional Skills	* c1 Explain how large? scale parallel systems are architectured and how massive parallelism are implemented in accelerator architectures * c2 Write parallel programs for large?scale parallel systems, shared address space platforms, and heterogeneous platforms * c3 Design efficient parallel algorithms and applications * c4 Be conversant with performance analyze and modeling of parallel programs
General Skill	* d1 An ability to analyze a problem and identify the computing requirements appropriate for its solution; an ability to design, implement and evaluate a computer? based system, process, component or program to meet desired needs

Course Contents

1 - Parallel computer architectures
2 - Principles of parallel algorithm design
3 - Programming scalable systems
4 - Programming shared-address space systems
5 - Programming heterogeneous systems
6 - Analytical modeling of parallel program
7 - Basic communication operations
8 - Graph algorithms
9 - Dense matrix algorithms
10 - Numerical algorithms
11 - Search algorithm for discrete optimization

Teaching and Learning Methods

1 - Lecture Notes
2 - Programming Assignments

Students Assessment

Assessment Method	TIME	MARKS
Midterm Examination	Week 8	30
Programming Assignments	During the Semester	20
Final Examination	Week 16	50

Books and References

Essential books	Ananth Grama, Anshul Gupta, George Karypis, and Vipin Kumar. Introduction to parallel computing, second edition, Addison-Wesley, 2003, ISBN: 0201648652

Knowledge and Skills Matrix

Main Course Contents	Study Week	Knowledge and Understanding	Intellectual Skills	Professional Skills	General Skill
Parallel computer architectures: Distributed memory systems, Shared memory systems and cache coherence, Heterogeneous system architecture, Interconnection networks and routing	Week 1	a1			
Parallel computer architectures (Cont.)	Week 2	a2	b1		
Principles of parallel algorithm design: Decomposition techniques, Characteristics of tasks and interactions, Mapping techniques for load balancing, Parallel algorithm models	Week 3	b3	b1	c1	
Principles of parallel algorithm design(Cont.)	Week 4	a4	b1	c2	
Principles of parallel algorithm design (Cont.) + Programming scalable systems: Programming using MPI paradigm, Programming using global address space language UPC	Week 5	a5	b2	c1	
Programming scalable systems (Cont.)	Week 6	a5	b1	c2	
Programming scalable systems (Cont.) + Programming shared?address space systems: OpenMP, Cilk Plus	Week 7	a3	b2	c3	
Programming shared?address space systems (Cont.) + Programming heterogeneous systems: CUDA and OpenCL, OpenACC and OpenMP (4.0)	Week 8	a3	b4	c2	
Programming heterogeneous systems (Cont.) + Analytical modeling of parallel program: Scalability of parallel systems, Sources of overhead in parallel programs, Asymptotic analysis	Week 9	a4	b5	c2	
Analytical modeling of parallel program (Cont.)	Week 10	a4	b5	c4	d1
Basic communication operations	Week 11	a3	b5	c3	d1

Graph algorithms	Week 12	a5	b3	c3	d1
Dense matrix algorithms	Week 13	a4	b2	c4	d1
Numerical algorithms	Week 14	a1,a4	b2,b5	c4	d1
Search algorithm for discrete optimization	Week 15	a1		c2	d1