CURRICULUM DESIGN AND ORGANIZATION

Title of degree program

PhD in Computer Systems Engineering

Major areas PhD program

- Image Processing
- Computer Architecture
- Parallel Computing
- Human Computer Interaction

Course Titles, Course Objectives, Pre/Co-requisites, Approvals, Contents and other details

S No.	Course Code	Name of Subject	Credits
1.	PHDCS801	Advanced Digital Image Processing	3
2.	PHDCS802	Computational Models & Complexity	3
3.	PHDCS803	Multicore Microprocessor Architecture Design	3
4.	PHDCS804	High Performance Computing	3
5.	PHDCS805	Advance Computer Architecture Memory- hierarchy Design	3
6.	PHDCS806	Parallel Computing Techniques	3
7.	PHDCS807	Human-centered Design and Computer Interaction	3
Total			**21**

MEHRAN UNIVERSITY OF ENGINEERING AND TECHNOLOGY, JAMSHORO
INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES
DEPARTMENT OF COMPUTER SYSTEMS ENGINEERING

Title of Subject:	**Advanced Digital Image Processing (PHDCS801)**
Disciplines:	*Computer Systems Engineering*
Pre-requisites:	
Assessment:	Sessional: 10%, Mid Semester: 30%, Final Examination: 60%
Credit Hours:	3 + 0
Max Marks:	100
Contact Hours:	42 + 0
Aims:	This course is designed to provide an advanced level understanding and the concepts of digital image processing and machine vision techniques and algorithm. The course contents mainly focus on automated image processing models and analytical techniques.
Objectives:	After completion of this course, the students should be able to • Understand and analyses basic and advanced level digital image representation including color profiles and, memory representation, computer processing and enhancement transforms. • Understand and apply digital filters (low pass, median) for noise reduction, image enhancement and edge detection. • Conceive image texture types and texture models. • Understand, analyse and apply image restoration techniques. • Understand and utilize Fourier transforms (FFT, DFT, Walsh and DCT) • Understand object detection and recognition methods. • Expose to stochastic and content-based image processing functions and algorithms
Contents:	**Computer vision Basics:** Vision and visual system, Color vision and Computer Vision, Related fields of computer vision, Application areas of computer vision, Major steps involved in computer vision system. **Image Basics:** Image Acquisition, Image Representation, Image Resolution, Image Storage, Neighborhood of pixels and pixel connectivity, Imaging Geometry and Affine Transformation. **Filtering, Image representation and Texture Models:** Image representations (continuous and discrete), Filtering (low-pass and median), derivatives, and edges, Edge detection and stages of edge detection, LoG and Canny's edge Detector, Statistical and structural texture models **Image Processing and Transform domain:** Fourier Transform, Time domain versus Frequency domain, 1-D and 2-D Fourier transform, Shortcomings in Fourier Transform, Introduction to Discrete Fourier Transform, Properties of DFT, Fast Fourier Transform, FFT Algorithm, FFT Computations, Introduction and applications of Walsh transform, and discrete cosine transform. **Object recognition:** Hough transforms and other simple object recognition

	methods, Shape correspondence and shape matching, Principal component analysis, Shape priors for recognition. **Advancements of Image Processing:** Content based and semantic based image retrieval, Visual Surveillance and Activity Monitoring, Image-Based Rendering, Image registration, medical image analysis, (MRI/PET/CT/X-ray tumor detection/classification), Face, fingerprint, and other object recognition, Image and/or video compression, Image segmentation and/or de-noising; Digital image/video, watermarking/steganography and detection, High Dynamic Range (HDR) Imaging

`Recommended Books:

1. Computer Vision - A modern approach, by D. Forsyth and J. Ponce, Prentice Hall
2. Digital Image Processing - R.C.Gonzalez & P.Wintz
3. Digital Image Processing and Computer Vision, R.J. Schalkoff, Wiley
4. Robot Vision, by B. K. P. Horn, McGraw-Hill.
5. Digital Image Processing - W. K. Pratt

Approvals:		
Board of Studies	Resolution No. 09.01	Dated: 07.09.2020
ASRB	Resolution No. 168.23 (a)	Dated: 06.10.2020
Academic Council	Resolution No. 98.13	Dated: 17.11.2020

MEHRAN UNIVERSITY OF ENGINEERING AND TECHNOLOGY, JAMSHORO

INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES

DEPARTMENT OF COMPUTER SYSTEMS ENGINEERING

Title of Subject:	**Computational Models and Complexity (PHDCS802)**
Disciplines:	*Computer Systems Engineering*
Pre-requisites:	-
Assessment:	Sessional: 10%, Mid Semester: 30%, Final Examination: 60%
Credit Hours:	3 + 0
Max Marks Contact Hours:	100 42 + 0
Aims:	The prime aim of this course is to address the theoretical and practical limitations of computation. To provide a theoretical framework for modelling computation. The concepts of undecidability and intractability are introduced through a number of examples.
Objectives:	After completion of this course, the students should be able to, • Analyse the complexity of a variety of problems and algorithms, • Reduce one problem to another, • Prove that a problem is undecidable, • Find a polynomial time reduction from one problem to another, • Determine the complexity class of a decidable problem.

Contents:	**Models of Computation:** Deterministic Turing machines, Equivalent Turing machines, Register machines. **Languages:** Language recognition, Language acceptance, Recursive languages, Recursively enumerable languages. **Undecidability:** The Halting Problem, Problem reduction, Undecidability of the tiling problem, Undecidability of first-order logic, Other unsolvable problems. **Complexity theory and Non-determinism:**Complexity hierarchies**,** Non-deterministic Turing machines, polynomial-time reduction, elementary properties of polynomial time reduction, The complexity classes P, NP, NP-complete, Cook's theorem, how to prove NP-hardness of various problems. First and second order logic complexity and their relation to complexity classes. Probabilistic algorithms and their complexity theory. Complexity for other computational models, such as exact real arithmetic, higher type functionals, and quantum computations. **Probabilistic Algorithms:** Examples of probabilistic algorithms, how to make 'almost sure' your algorithm is correct, complexity analysis of probabilistic algorithms, The complexity classes PP and BPP. **Other Complexity Classes:** Space complexity, Savitch's theorem, Exponential time, Non-elementary problems.

Recommended Books:

1. Computability and Unsolvability, by Prof. Martin Davis, ISBN: 0486614719
2. A first Course in Computability, Blackwell, Scientific Publications, V.J., Rayward-Smith 1986.
3. Elements of the Theory of Computation, H. Lewis and C. Papadimitriou, Prentice Hall, 1998.

Approvals:			
	Board of Studies	Resolution No. 09.01	Dated: 07.09.2020
	ASRB	Resolution No. 168.23 (a)	Dated: 06.10.2020
	Academic Council	Resolution No. 98.13	Dated: 17.11.2020

MEHRAN UNIVERSITY OF ENGINEERING AND TECHNOLOGY, JAMSHORO

INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES

DEPARTMENT OF COMPUTER SYSTEMS ENGINEERING

Title of Subject:	**Multi-Core Microprocessor Architecture Designs (PHDCS803)**
Disciplines:	Computer Systems Engineering
Pre-requisites:	-
Assessment:	Sessional: 10%, Mid Semester: 30%, Final Examination: 60%
Credit Hours:	3 + 0

Max Marks Contact Hours:	100 42 + 0
Aims:	This course examines the techniques and underlying principles that are used to design high-performance computers and processors. Particular emphasis is placed on understanding the trade-offs involved when making design decisions at the architectural level for single-core and multi-core. A range of processor architectures are explored and contrasted. Memory management and requirement for many cores is discussed and Multi-core programming is also introduced
Objectives:	After completion of this course, the students should be able to • Understand the basics of multi-core architecture and comprehend parallel system requirement and limitation of single core processor along with quantitative principles of computer design, • Perceive and analyse vector architectures and loop level parallelism, • Understand the memory system architecture for multi-core and multi-processor systems, • Write introductory level programs for multi-core processors.
Contents:	Introduction: Overview and taxonomy of multi-core processor architectures, Parallel processing systems classification, Flynn taxonomy Classes and types of parallel computers Bit-level, instruction level, data-level and task-level parallelization, Amdahl's Law, quantitative principles of computer design-classes of parallelism, ILP, DLP, TLP and RLP, multithreading ,SMT and CMP architectures, Limitations of single core processors DLP in vector: Vector architecture, SIMD instruction set extension for multimedia-graphics processing units- detecting and enhancing loop level parallelism Memory: Memory system architecture in multiprocessor systems, memory sharing, symmetric and distributed shared memory architectures, cache coherence issues, performance issues, synchronization issues, models of memory consistency, interconnection networks, buses crossbar and multi-stage interconnection networks. Supercomputers: Current supercomputers and processing clusters, architecture of current multi-core processors, technological solutions applied in Intel, AMD, nVidia processors, parallel systems topologies and their scalability, inter-processor communication. Power, defects and redundancy: Defects and faults in multi-core processors, hardware and software redundancy, Power dissipation in multicore processors Programming: Programming of current multi-core processors, parallel programming languages; Perspective of multi-core architectures development in the future. Multi-core architecture for embedded systems: Features and requirement for embedded systems, signal processing and embedded applications, digital signal processor, embedded microprocessors

Recommended Books: 1. Computer Organization and Architecture, by William Stallings, Prentice Hall. 2. Introduction to Parallel Computing, by Ananth Grama et al, Addison-Wesley, 2003. 3. The Art of Multiprocessor Programming, by Maurice Herlihy, Nir Shavit, Morgan Kaufmann, 2008.		
Approvals:		
Board of Studies	Resolution No. 09.01	Dated: 07.09.2020
ASRB	Resolution No. 168.23 (a)	Dated: 06.10.2020
Academic Council	Resolution No. 98.13	Dated: 17.11.2020

MEHRAN UNIVERSITY OF ENGINEERING AND TECHNOLOGY, JAMSHORO

INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES

DEPARTMENT OF COMPUTER SYSTEMS ENGINEERING

Title of Subject:	**High Performance Computing (PHDCS804)**
Disciplines:	Computer Systems Engineering
Pre-requisites:	-
Assessment:	Sessional: 10%, Mid Semester: 30%, Final Examination: 60%
Credit Hours:	3 + 0
Max Marks Contact Hours:	100 42 + 0
Aims:	The course provides a solid foundation in High Performance Computing (HPC) and its role in science and engineering. The principal methods of measuring and characterising application and hardware performance are then covered, with particular reference to complex scientific- and business-based HPC codes. The final topic is the Computational Grid, where the fundamental issues of administration, scheduling, code portability and data management are explored.
Objectives:	After completion of this course, the students should be able to, • The role of HPC in science and engineering. • The most commonly used HPC platforms and parallel programming models. • The means by which to measure, analyse and assess the performance of HPC applications and their supporting hardware. • Mechanisms for evaluating the suitability of different HPC solutions to common problems found in Computational Science. • The potential benefits and pitfalls of Grid Computing.
Contents:	**Fundamental concepts in High Performance Computing:** Parallel computer and architectures, reduced instruction set computers, vector processors, multi-core processors.

	Parallel semantic: Single–instruction, single–data (SISD), Single–instruction, multiple–data (SIMD), Multiple instructions, multiple data (MIMD), Coarse–grain parallel, Medium–grain parallel, Fine–grain parallel systems **Shared memory programming (OpenMP):** Shared Memory Concepts; OpenMP Fundamentals; Parallel Regions, Exercises: Hello World; Parallel regions, Work sharing, Parallel loops, Synchronization, Orphaning, OpenMP Tasks, Sequential, optimization techniques, Multithreaded optimization. **Message passing programming (MPI):** Message-Passing Concepts, Parallel Traffic Modelling, MPI Programs, MPI on ARCHER, Point-to-Point Communication, Communicators, Tags and Modes, Non-Blocking Communication, Message Round a Ring, Collective Communication, Virtual Topologies, Derived Data Types. **Advanced topics:** Parallel decompositionHardware, compilers and performance programming, Performance measurement and estimation, High performance networking, Computation grids.

Recommended Books:
1. High Performance Computing: Programming and Applications, First Edition, by John Levesque Gene Wagenbreth, ISBN: 1420077058.
2. Introduction to High Performance Computing for Scientists and Engineers, by Georg Hager, Gerhard Wellein, ISBN: 143981192X
3. The Art of Multiprocessor Programming, Morgan Kaufmann, by Maurice Herlihy, Nir Shavit, 2008.

Approvals:		
Board of Studies	Resolution No. 09.01	Dated: 07.09.2020
ASRB	Resolution No. 168.23 (a)	Dated: 06.10.2020
Academic Council	Resolution No. 98.13	Dated: 17.11.2020

MEHRAN UNIVERSITY OF ENGINEERING AND TECHNOLOGY, JAMSHORO
INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES
DEPARTMENT OF COMPUTER SYSTEMS ENGINEERING

Title of Subject:	**Advanced Computer Architecture and Memory Hierarchy Design (PHDCS805)**
Disciplines:	Computer Systems Engineering
Pre-requisites:	-
Assessment:	Sessional: 10%, Mid Semester: 30%, Final Examination: 60%
Credit Hours:	3 + 0
Max Marks Contact Hours:	100 42 + 0
Aims:	To understand simple to advanced level microprocessor architecture and memory (cache) management systems. Further, to learn advanced techniques deployed in

	state-of-the art microprocessors from major industries. Learn how to use tools, techniques, and models to evaluate microprocessor design.
Objectives:	After completion of this course, the students should be able to • Understand and characterize the basic and advanced concepts of microprocessor architecture design including memory hierarchy, I/O and Instruction set. • Write compile, debug and execute low level languages programs. • Perform quantitative analyses of Single core, multi-core and superscalar architectures. • Understand and evaluate the cache memory systems based on performance parameters
Contents:	**Introduction:** Introduction to computer architecture, Embedded Systems, Harvard Architecture, Von Neumann Machines, Design Cycle and Metrics, Microcontrollers, Microprocessors, memory & I/O, Pipelining, Concept of Superscalar. **Single Core Architecture:** Addressing modes, reset Interrupt Priority and Management, the 16-bit CPU architecture, Memory Management and Paging, instruction set overview and product selection. **From Source to Execution:** Assemblers, Compilers, and Debuggers. ELF debug output and object files. Debugging techniques and pitfalls. Emulators, simulators, and programmers. **Quantitative Analysis:** Quantitative Analysis of Superscalar Processors, Multicores Processors, Advanced Branch Predictors, Advanced Cache Management, Front-end Design Back-end Design. **Microcontrollers and Peripherals**: Memory management and paging, Timers, ADC, IIC, Flash, CAN, SCI and SPI. **Advanced Architectures:** ColdFire Architecture, The 32-bit ColdFire Microprocessor, instruction set, addressing modes, cache and memory management, Interrupt priority levels and management, The Intel IA-32 Architecture, The ARM CPU architecture. **Symmetric Multiprocessing:** Parallel computing taxonomy, SMP and cache management, memory management of a parallel machine. **PCI:** The Peripheral Component Interconnect bus specification, bridging, posting, forwarding, caching, overview of PCI express architecture. **Cache Memory Management:** Cache system Installation and configuration, Architecture, Cache Architecture, Databases, Namespaces, Routines, Globals, Classes. **Licensing:** License Units, Processes and Configuration, Configuration, Networking Parameters, Journal Files, Creating Namespaces and Databases, Global and Routine Mapping, Configuration Devices, Configuration Files, Journaling: Write **Image Journaling:** journal files, Cache Networking: Remote Sensing Access, ECP (Enterprise Cache Protocol) **Regular Operations**: Cache Backup, Review of Alternate Backup Methods, Restore, Integrity Checks, Databases

	Expansion, Mirroring and Shadowing, Task Scheduler, Automation and Notification

Recommended Books:

1. Computer Architecture: A Quantitative Approach, Fourth Edition. John Hennessy and David Patterson. Morgan Kauffman (Elsevier), 2006, ISBN-13: 978-0-12-370490-0
2. A Course in In-Memory Data Management, by Hasso Plattner
3. Digital Signal Processing – A Practical Approach,by Ifeachor M., Jarvis B.W., Pearson Education.
4. Vaseghi S.V. Advanced Digital Signal Processing and Noise Reduction, Wiley.

Approvals:			
	Board of Studies	Resolution No. 09.01	Dated: 07.09.2020
	ASRB	Resolution No. 168.23 (a)	Dated: 06.10.2020
	Academic Council	Resolution No. 98.13	Dated: 17.10.2020

MEHRAN UNIVERSITY OF ENGINEERING AND TECHNOLOGY, JAMSHORO
INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES
DEPARTMENT OF COMPUTER SYSTEMS ENGINEERING

Title of Subject:	**Parallel Computing Techniques (PHDCS806)**
Disciplines:	Computer Systems Engineering
Pre-requisites:	-
Assessment:	Sessional: 10%, Mid Semester: 30%, Final Examination: 60%
Credit Hours:	3 + 0
Max Marks Contact Hours:	100 42 + 0
Aims:	This is an advanced interdisciplinary introduction to applied parallel computing on modern supercomputers. It has an emphasis on understanding the realities and myths of what is possible on multi-core machines. This course aims to make student acquainted with free, open-sourceand high-performance dynamic programming languages for technical computing.
Objectives:	After completion of this course, the students should be able to, • Fundamental design philosophies that multicore architectures address, • Parallel programming philosophies and emerging best practices, • Use and evaluate parallel computing techniques to sole real life problems.
Contents:	**Introduction:** Why Parallelism, modern multi-core processor, parallel programming models, parallel programming basics. **Parallel computer architectures,** shared memory systems and cache coherence distributed-memory systems, interconnection networks and routing. **Performance optimization:** Work distribution and scheduling, locality, communication, and contention, performance monitoring tools, GPU architecture and cuda programming, parallel application case studies, performance measurement and analysis of parallel programs

	Principles of parallel algorithm design: Decomposition techniques, mapping & scheduling computation, templates. **Programming shared-address space systems:** Cilk Plus, OpenMP, Pthreads. **Programming scalable systems:** message passing, MPIs, global address space languages **Analytical modeling of program performance:** speedup, efficiency, scalability, cost optimality, iso-efficiency, collective communication, synchronization **Non-numerical algorithms:** Sorting, graphs, dynamic programming. **Numerical algorithms:** Dense matrix algorithms, sparse matrix algorithms **GPU Programming:** Problem solving on clusters using MapReduce, Warehouse-scale computing. **Cell processor Programming:** Introduction to Cell processor, getting to know Cell, Introduction to parallel architectures, Introduction to concurrent programming, Design patterns for parallel programming. Cell programming hands- on, stream it language, cell debugging tools, debugging parallel, Programs Performance monitoring and optimizations. **Advanced topics**: Analyzing program performance on a multi-core CPU, Simple CUDA Renderer, Parallel VLSI Wire Routing via OpenMP, parallel VLSI wire routing via MPI.

Recommended Books:

1. Introduction to Parallel Computing, Second Edition, Ananth Grama, George Karypis, Vipin Kumar, Anshul Gupta, Addison-Wesley, 2003, ISBN: 0201648652.
2. The Art of Multiprocessor Programming, by Maurice Herlihy, Nir Shavit, Morgan Kaufmann, 2008.

Approvals:		
Board of Studies	Resolution No. 09.01	Dated: 07.09.2020
ASRB	Resolution No. 168.23 (a)	Dated: 06.10.2020
Academic Council	Resolution No. 98.13	Dated: 17.11.2020

MEHRAN UNIVERSITY OF ENGINEERING AND TECHNOLOGY, JAMSHORO

INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES

DEPARTMENT OF COMPUTER SYSTEMS ENGINEERING

Title of Subject:	**Human-centered Design and Computer Interaction (PHDCS807)**
Disciplines:	Computer Systems Engineering
Pre-requisites:	
Assessment:	Sessional (Project): 30%, Mid Semester: 30%, Final Examination: 40%
Credit Hours:	3 + 0

<table>
<tr><td>Max Marks:</td><td>100</td></tr>
<tr><td>Contact Hours:</td><td>42 + 0</td></tr>
<tr><td></td><td></td></tr>
<tr><td>Aims:</td><td>This course serves as interdisciplinary Ph.D. level research in Human Computer Interaction, design process and theory. It will provide background material in key HCI related areas, discuss the nature of research typically performed in an HCI context, and explore the processes and methodologies typical of interdisciplinary work in HCI. The course will be project-oriented, giving students the experience with a small interdisciplinary research project related to research/ project theme.</td></tr>
<tr><td>Objectives:</td><td>After completion of this course, the students should be able to
Understand various aspects of Human-Computer Interaction in different engineering and non-engineering disciplines.
to carry out the design process involved in interaction design, navigation design, and systems interface design
How cognition and perception, which encompass attention, memory, thought, the “senses” play a role in affecting the experience of interactive design
Effectively utilize and experience user’s behavior and response to the research
Design and analyze HCI systems, applications, architectures, and frameworks</td></tr>
<tr><td>Contents:</td><td>HCI and Design: Learning curves of HCI, Changes and trends, Stages in interaction design process (IDP), Prioritizing the IDP

Human-centered Interaction: User focus (personae, probes), Navigation design, 4 golden rules (between screens, within Application and beyond), design and layout, whitespace physical devices, User action and control, appropriate appearance, analyzing website navigation and screen design, prototyping, animated evaluation.

Cognition and perception: Human perception and cognition, Senses and more, Optical illusion, Man-machine nightmares, human-error and healthcare, Memory and types of memory, memory education, Thinking (Wason, and Fitt’s law),
Design emotions: Cracker’s case study, design exercises for peak and episodic experiences, The drift table, Kursaal Flyers and relevant research-based scenarios

Advanced model and frameworks of implementation: Screen implementation and resource management, Architecture of implementation, Event models, General framework, MVC in nuclear control room, SEEHEIM scenario, UI design, Distributed Cognition, Situated Action, and Activity Theory, Evaluation technique (Mini project/ discussion)</td></tr>
<tr><td colspan="2">Recommended Books:
Strategies for Effective Human-Computer Interaction - Designing the User Interface,(5th ed.). Addison-Wesley by Shneiderman, B. and Plaisant, C.
Interaction Design: Beyond Human-Computer Interaction (5th Edition), Wiley, by Jennifer Preece, Yvonne Rogers, and Helen Sharp
Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Rules, Morgan Kaufmann, by Jeff Johnson
The Design of Everyday Things: Revised and Expanded Edition, Basic Book publishers by Donald A. Norman</td></tr>
</table>

Approvals:			
	Board of Studies	Resolution No. 09.01	Dated: 07.09.2020
	ASRB	Resolution No. 168.23 (a)	Dated: 06.10.2020
	Academic Council	Resolution No. 98.13	Dated: 17.11.2020