COMSATS Institute of Information Technology
Registrar Office Principal Seat, Islamabad
*****

No: CIIT-Reg/Notif-322/16/350 **January 20, 2016**

NOTIFICATION
BS(CS) ABET

The Academic Council in its 24th meeting held on December 01, 2015 approved the Scheme of Studies of Bachelor of Science in Computer Science, effective from Fall 2016 for CIIT Islamabad Campus only:

1. **Nomenclature of the Program: Bachelor of Science in Computer Science**

2. **Minimum Duration**
   i. No. of Years: 04
   ii. No. of Semesters: 08

3. **Course Work**

	Min No. of Courses	Min No. of Credit Hours
4. Core Courses (List Attached):		
4.1 Computing Core Courses	12	41
4.2 Computing Supporting Areas Courses	4	13
4.3 Computing General Education Courses	7	21
4.4 Computer Science Core Courses	7	21
4.5 Computer Science Supporting Areas Courses	3	9
5. Elective Courses (List Attached)		
5.1 Computer Science Elective Courses	5	15
5.2 Institutional Elective Courses	2	6
5.3 Sciences Elective Courses	2	8
6. Total No. of Courses of the Program:	**42**	
7. Total Credit Hours of the Program:		**134**

Note:

i. The Regulations relating to Undergraduate Degree Programs approved by the Competent Authority and amended from time to time shall be applicable.

ii. In the list of Institutional elective courses, a student has to take at least one course from Category I. The other course can be taken either from Category I or Category II.

iii. In order to specialize in a particular track, student must select at least ***three*** courses from that track. *CSC350* and *CSC483* (Category of General Computer Science Electives) can be part of any track.

Nadeem Uddin Qureshi
Additional Registrar

Distribution:
1. All Directors, CIIT Campuses
2. In-charge, CIIT Islamabad Campus
3. All Deans, CIIT System
4. All Chairpersons of the Faculty of Business Administration, CIIT
5. Controller of Examinations, CIIT
6. All HoDs/In-charges of Academics/Examinations Sections, CIIT, Campuses

CC:
1. PS to Rector
2. PA to Registrar

Computing Core Courses

Sr No.	Course Code	Course Title	Credit Hours	Pre-requisite (s)
1	CSC102	Discrete Structures	3(3, 0)	-
2	CSC103	Programming Fundamentals	4(3, 1)	-
3	CSC211	Data Structures and Algorithms	4(3, 1)	CSC103
4	CSC241	Object Oriented Programming	4(3, 1)	CSC103
5	CSC291	Software Engineering Concepts	3(3, 0)	-
6	CSC322	Operating Systems	3(2, 1)	CSC211
7	CSC339	Data Communications and Computer Networks	3(2, 1)	-
8	CSC356	Human Computer Interaction	3(2, 1)	-
9	CSC371	Database Systems I	4(3, 1)	CSC241
10	CSC498	Senior Design Project I	2(0, 2)	CSC371, CSC291, HUM102, CSC241
11	CSC499	Senior Design Project II	4(0, 4)	CSC498
12	EEE241	Digital Logic Design	4(3, 1)	-
		Total Credit Hours	**41**	

Computing Supporting Courses

Sr No.	Course Code	Course Title	Credit Hours	Pre-requisite (s)
1	MTH104	Calculus and Analytic Geometry	3(3, 0)	-
2	MTH231	Linear Algebra	3(3, 0)	-
3	MTH262	Statistics and Probability Theory	3(3, 0)	-
4	PHY121	Applied Physics for Engineers	4(3, 1)	-
		Total Credit Hours	**13**	

Computing General Education Courses

Sr No.	Course Code	Course Title	Credit Hours	Pre-requisite (s)
1	CSC101	Introduction to ICT	3(2, 1)	-
2	CSC110	Professional Practices for IT	3(3, 0)	-
3	HUM100	English Comprehension and Composition	3(3, 0)	-
4	HUM102	Report Writing Skills	3(3, 0)	HUM100
5	HUM103	Communication Skills	3(3, 0)	HUM100

6	HUM110	Islamic Studies **	3(3, 0)	-
7	HUM111	Pakistan Studies	3(3, 0)	-
		Total Credit Hours	**21**	

**Non-Muslim students can opt for HUM114 Ethics 3(3, 0) course in lieu of HUM110 Islamic Studies, if they intend to.

Computer Science Core Courses

	Course Code	Course Title	Credit Hours	Pre-requisite (s)
1	CSC301	Design and Analysis of Algorithms	3(3, 0)	CSC211
2	CSC312	Theory of Automata	3(3, 0)	CSC102
3	CSC321	Microprocessor and Assembly Language	3(2, 1)	EEE241
4	CSC336	Web Technologies	3(2, 1)	CSC241
5	CSC441	Compiler Construction	3(2, 1)	CSC312
6	CSC462	Artificial Intelligence	3(2, 1)	CSC102
7	EEE440	Computer Architecture	3(3, 0)	EEE241
		Total Credit Hours	**21**	

Computer Science Supporting Areas Courses

	Course Code	Course Title	Credit Hours	Pre-requisite (s)
1	CSC475	Numerical Computing	3(3, 0)	MTH104
2	MTH105	Multivariable Calculus	3(3, 0)	MTH104
3	MTH242	Differential Equations	3(3, 0)	MTH104
		Total Credit Hours	**9**	

Elective Courses

Computer Science Elective Courses

Track I/Specialization : Software Development

Sr No.	Course Code	Course Title	Credit Hours	Pre-requisite (s)
1	CSC303	Mobile Application Development	3(2, 1)	CSC241
2	CSC392	Software Design Methodologies	3(3, 0)	CSC291

3	CSC412	Visual Programming	3(2, 1)	CSC241
4	CSC494	Software Project Management	3(3, 0)	CSC291
5	CSC495	Game Development	3(3, 0)	CSC241

Track II/Specialization: Network and Security

Sr No.	Course Code	Course Title	Credit Hours	Pre-requisite (s)
1	CSC334	Parallel and Distributed Computing	3(2, 1)	CSC103
2	CSC430	Wireless Networks	3(2, 1)	-
3	CSC432	Information Security	3(3, 0)	CSC339
4	CSC482	Systems and Networks Administration	3(2, 1)	CSC322, CSC339

Track III/Specialization: Database Technologies

Sr No.	Course Code	Course Title	Credit Hours	Pre-requisite (s)
1	CSC347	Computational Intelligence	3(3, 0)	CSC241
2	CSC402	Database Systems II	3(3, 0)	CSC371
3	CSC461	Introduction to Data Science	3(3, 0)	MTH231, MTH262
4	CSC471	Distributed Database Systems	3(3, 0)	CSC371
5	CSC496	Data Warehousing and Data Mining	3(2, 1)	CSC371

Track IV/Specialization: Artificial Intelligence and Graphics

Sr No.	Course Code	Course Title	Credit Hours	Pre-requisite (s)
1	CSC331	Digital Image Processing	3(3, 0)	MTH231, MTH262
2	CSC353	Computer Graphics	3(2, 1)	MTH231
3	CSC354	Machine Learning	3(3, 0)	-
4	CSC421	Robotics	3(3, 0)	-
5	CSC451	Introduction to Modeling and Simulation	3(2, 1)	CSC102, CSC211
6	CSC454	Pattern Recognition	3(3, 0)	CSC354
7	CSC455	Computer Vision	3(3, 0)	MTH231& MTH262

General Computer Science Electives: **

Sr No.	Course Code	Course Title	Credit Hours	Pre-requisite (s)
1	CSC350	Topics in Computer Science I	3(3, 0)	-
2	CSC483	Topics in Computer Science II	3(3, 0)	-

Institutional Elective Courses*

Sr No.	Course Code	Course Title	Credit Hours	Pre-requisite (s)
	Category I			
1	MGT100	Introduction to Business	3(3, 0)	-
2	MGT101	Introduction to Management	3(3, 0)	-
3	MGT131	Financial Accounting	3(3, 0)	-
4	MGT210	Fundamentals of Marketing	3(3, 0)	-
5	MGT350	Human Resource Management	3(3, 0)	-
	Category II			
1	HUM220	Introduction to Psychology	3(3, 0)	-
2	HUM221	International Relations	3(3, 0)	-
3	HUM430	French	3(3, 0)	-
4	HUM431	German	3(3, 0)	-
5	HUM432	Arabic	3(3, 0)	-
6	HUM433	Persian	3(3, 0)	-
7	HUM434	Chinese	3(3, 0)	-
8	HUM435	Japanese	3(3, 0)	-

Sciences Elective Courses

Sr No.	Course Code	Course Title	Credit Hours	Pre-requisite (s)
1	BIO100	Fundamentals of Biology	3(2, 1)	-
2	BIO135	General Chemistry	4(3, 1)	-
3	BIO231	Fundamentals of Genetics	4(3, 1)	-
4	BIO233	Developmental Biology	4(3, 1)	-
5	BIO310	Introduction to Bioinformatics	4(3, 1)	-
6	BSC101	Zoology I	4(3, 1)	-
7	BSC102	Plant Sciences I	4(3, 1)	-
8	BSC206	Introduction to Microbiology	4(3, 1)	-
9	ENV230	Environmental Monitoring	3(2, 1)	-
10	MTH112	Set Theory and Logic	3(3, 0)	-
11	MTH232	Abstract Algebra	3(3, 0)	-

12	MTH251	Set Topology	3(3, 0)	-
13	PHY100	Mechanics and Thermodynamics	4(3, 1)	-
14	PHY120	Electricity, Magnetism and Optics	4(3, 1)	-
15	PHY229	Modern Physics	4(3, 1)	PHY120
16	SLS130	Introductory Soil Science I	4(3, 1)	-

Jaunder

No: CIIT-Reg/Notif-322/16/358

January 20, 2016

Tentative Study Plan for BS(CS)

This is a proposed model with reference to offering of courses in a particular semester. A campus may make logical changes depending upon their available resources and requirements.

Semester 1				
S. No	**Course Code**	**Course Name**	**Credit Hours**	**Pre-Requisite (s)**
1	CSC101	Introduction to ICT	3(2, 1)	-
2	HUM100	English Comprehension and Composition	3(3, 0)	-
3	HUM110	Islamic Studies	3(3, 0)	-
4	MTH104	Calculus and Analytic Geometry	3(3, 0)	-
5	PHY121	Applied Physics for Engineers	4(3, 1)	-
Credit Hours			**16(14, 2)**	

Semester 2				
S. No	**Course Code**	**Course Name**	**Credit Hours**	**Pre-Requisite (s)**
1	CSC103	Programming Fundamentals	4(3, 1)	-
2	HUM102	Report Writing Skills	3(3, 0)	HUM100
3	MTH105	Multivariable Calculus	3(3, 0)	MTH104
4	CSC110	Professional Practices for IT	3(3, 0)	-
5	CSC102	Discrete Structures	3(3, 0)	-
6		Science Elective I	4(3, 1)	-
Credit Hours			**20(18, 2)**	

Semester 3				
S. No	**Course Code**	**Course Name**	**Credit Hours**	**Pre-Requisite (s)**
1	CSC241	Object Oriented Programming	4(3, 1)	CSC103
2	EEE241	Digital Logic Design	4(3, 1)	-
3	MTH262	Statistics and Probability Theory	3(3, 0)	-
4	MTH231	Linear Algebra	3(3, 0)	-
5		Science Elective II	4(3, 1)	-
Credit Hours			**18(15, 3)**	

Semester 4				
S. No	**Course Code**	**Course Name**	**Credit Hours**	**Pre-Requisite (s)**
1	CSC211	Data Structures and Algorithms	4(3, 1)	CSC103
2	EEE440	Computer Architecture	3(3, 0)	EEE241
3	CSC291	Software Engineering Concepts	3(3, 0)	-
4	MTH242	Differential Equations	3(3, 0)	MTH104
5	HUM103	Communication Skills	3(3, 0)	HUM100
Credit Hours			**16(15, 1)**	

Semester 5				
S. No	Course Code	Course Name	Credit Hours	Pre-Requisite (s)
1	CSC312	Theory of Automata	3(3, 0)	CSC102
2	CSC301	Design and Analysis of Algorithms	3(3, 0)	CSC211
3	CSC322	Operating Systems	3(2, 1)	CSC211
4	CSC371	Database Systems I	4(3, 1)	CSC241
5		Computer Science Elective I	3(2, 1)	-
		Credit Hours	**16(13, 3)**	

Semester 6				
S. No	Course Code	Course Name	Credit Hours	Pre-Requisite (s)
1	CSC321	Microprocessor and Assembly Language	3(2, 1)	EEE241
2	CSC339	Data Communications and Computer Networks	3(2, 1)	-
3	CSC356	Human Computer Interaction	3(2, 1)	-
4	CSC336	Web Technologies	3(2, 1)	CSC241
5	HUM111	Pakistan Studies	3(3, 0)	-
6		Computer Science Elective II	3(2, 1)	-
		Credit Hours	**18(13, 5)**	

Semester 7				
S. No	Course Code	Course Name	Credit Hours	Pre-Requisite (s)
1	CSC441	Compiler Construction	3(2, 1)	CSC312
2	CSC462	Artificial Intelligence	3(2, 1)	CSC102
3	CSC475	Numerical Computing	3(3, 0)	MTH104
4		Computer Science Elective III	3(2, 1)	
5		Institutional Elective I	3(3, 0)	
6	CSC498	Senior Design Project I	2(0, 2)	CSC371, CSC291 HUM102, CSC241
		Credit Hours	**17(12, 5)**	

Semester 8				
S. No	Course Code	Course Name	Credit Hours	Pre-Requisite (s)
1		Computer Science Elective IV	3(3, 0)	
2		Computer Science Elective V	3(2, 1)	
3		Institutional Elective II	3(3, 0)	
4	CSC499	Senior Design Project II	4(0, 4)	CSC498
			Total 13(8, 5)	
Total Credit Hours			**134 (108, 26)**	

Computing Core Courses

Course Code: CSC103

Course Title: Programming Fundamentals

Credit Hours: 4(3, 1)

Pre-Requisite: None

Course Objectives:

- To introduce various programming language paradigms;
- To develop the skills to analyze, design, test and translate problems into computer programs;
- To present the fundamental programming concepts, including basic type system;
- To demonstrate basic coding, testing and debugging techniques;
- To provide an implementation of the concepts.

Course Contents:

Overview of Computer Programming; Principles of Structured, Modular and Functional Programming; Overview of Structured Programming Languages; Algorithms and Problem Solving; Program Development: Analyzing Problem, Designing Algorithm/Solution, Testing Designed Solution, Translating Algorithms into Programs; Basic Syntax and Semantics of a Higher-Level Language; Variables and Primitive Data Types; Expressions and Assignments; Simple I/O including File I/O; Conditional and Iterative Control Structures; Functions and Parameter Passing; Arrays; Strings and String Processing; The Concept of Recursion; The Concept of Type System: Compound Types Built from other Types (e.g. Array, Functions), Association of Types, Type Safety, Static Typing; Program Comprehension; Program Correctness: Types of Errors (Syntax, Logic, Run-Time), The Concept of a Specification, Code Reviews, Testing Fundamentals and Test-Case Generation, Unit Testing; Simple Refactoring; Modern Programming Environments: Code Search, Programming using Library Components and their APIs; Debugging Strategies; Documentation and Program Style.

Recommended Books:

1. Introduction to JAVA Programming, Liang, Y.D., Prentice Hall, 2012.
2. Java: An Introduction to Problem Solving and Programming, Savitch, W., Addison-Wesley, 2012.

Course Code: CSC241

Course Title: Object Oriented Programming

Credit Hours: 4(3, 1)

Pre-Requisite: CSC103

Course Objectives:

- To introduce the object oriented programming paradigm;
- To teach in depth the philosophy of object-oriented design and concepts of encapsulation, abstraction, inheritance and polymorphism;
- To develop understanding of sub typing and generic types;
- To explain the usage of library components;
- To develop code that responds to exception conditions raised during execution;
- To develop understanding of event handlers for use in reactive systems, such as GUIs;
- To demonstrate implementation of the concepts.

Course Contents:

Evolution of Object Oriented Programming (OOP); Object Oriented Concepts and Principles; Problem Solving in Object Oriented Paradigm; OOP Design: Decomposition into Objects, Class-Hierarchy Design for Modeling; Association; Aggregation; Composition; Definition of Classes; Subclasses; Inheritance; Method Overriding; Dynamic Dispatch: Definition of Method-Call; Sub-typing: Subtype Polymorphism, Implicit Up-casts, Notion of Behavioral Replacement (Subtypes acting like Super-types), Relationship between Sub-typing and Inheritance; Generic Types; Static and Dynamic Typing; Object-Oriented Idioms for Encapsulation: Privacy and Visibility of Class Members, Interfaces, Abstract Base Classes; Using Library Components such as Collection Classes and Iterators; Exception Handling; Events and Event Handlers; Canonical Uses such as GUIs, Mobile Devices; Using Reactive Framework; Externally-Generated Events and Program-Generated Events.

Recommended Books:

1. Absolute Java, Savitch, W. & Mock, K., Pearson, 2012.
2. Java How to Program, Deitel, P. & Deitel, H., Prentice Hall, 2012.
3. Java: The complete Reference, Schildt, H., McGraw Hill, 2011.

Course Code: CSC102

Course Title: Discrete Structures

Credit Hours: 3(3, 0)

Pre-Requisite: None

Course Objectives:

- To teach important discrete data structures such as sets, relations, functions, graph and trees;
- To introduce a formal system (propositional and predicate logic) on which mathematical reasoning is based;
- To thoroughly train in the construction and understanding of mathematical proofs;
- To exercise common mathematical arguments and proof strategies;
- To develop the ability to see a problem from a mathematical perspective.

Course Contents:

Sets: Venn Diagrams, Union, Intersection; Relations: Reflexivity, Symmetry, Transitivity, Equivalence Relations; Functions: Surjection, Injections, Bijection; Propositional Logic; Logical Connectives; Truth Tables; Normal Forms; Validity of Well-formed Formula; Propositional Inference Rules; Predicate Logic; Limitations of Propositional and Predicate Logic: Notions of Implication, Converse, Inverse, Contra-positive, Negation and Contra-diction; The Structure of Mathematical Proofs; Direct Proofs; Disproving by Counter Example; Proof by Contradiction; Induction over Natural Numbers; Structural Induction; Weak and Strong Induction; Recursive Mathematical Definitions; Well Orderings; Counting Arguments: Set Cardinality and Counting, Sum and Product Rule; The Pigeonhole Principle; Permutations and Combinations; Solving Recurrence Relations; Basic Modular Arithmetic; Trees; Undirected Graphs; Directed Graphs; Weighted Graphs; Spanning Trees/Forests; Graph Isomorphism; Finite Probability Space; Events; Axioms of Probability and Probability Measures; Conditional Probability, Baye's Theorem; Independence; Integer Random Variables; Expectation, Linearity of Expectation; Variance; Conditional Independence.

Recommended Books:

1. Discrete Mathematics and Its Applications, Rosen, K. H., McGraw Hill, 2012.
2. Discrete Mathematics with Applications, Susanna, S.E., Brooks Cole, 2010.
3. Discrete Mathematics with Proof, Gossett, E., Wiley, 2009.

Course Code: CSC211

Course Title: Data Structures and Algorithms

Credit Hours: 4(3, 1)

Pre-Requisite: CSC103

Course Objectives:

- To discuss the issues of time complexity and examine various algorithms from this perspective;
- To introduce the concept and usage of data structures through abstract data structures, including linked lists, stacks, queues, priority queue, trees, and graphs;
- To implement above data structures and their applications;
- Develop an understanding of recursion as they apply to trees and graphs;
- To introduce the concept of memory management and garbage collection.

Course Contents:

Introduction to Data Structures and Algorithms; Complexity Analysis; Sorting and Binary Search Algorithms; Linked Lists: Singly Linked Lists, Doubly Linked Lists, Circular List; Stacks; Queues and Priority Queue; Recursion: Function Call and Recursion Implementation, Tail Recursion, Non-tail Recursion, Indirect Recursion, Nested Recursion, Backtracking; Trees: Binary Trees, Binary Heap, Binary Search Tree, Traversal, Insertion, Deletion and Balancing a Tree; Heap; B-Tree; Spanning Tree; Splay Trees; Graphs: Representation, Traversal, Shortest Path, and Cycle Detection; Isomorphic Graphs; Graph Traversal Algorithms; Pattern Matching and String/Text Algorithms; Hashing; Memory Management and Garbage Collection.

Recommended Books:

1. Object Oriented Data Structures using Java, Dale, N., Jones and Bartlett Publishers, 2012.
2. Data Structures and Algorithms in Java, Goodrich & Tomassia, Wiley, 2010.

Course Code: CSC291

Course Title: Software Engineering Concepts

Credit Hours: 3(3, 0)

Pre-Requisite: None

Course Objectives:

- Introduce the different software process models by illustrating its phases;
- Develop awareness of using different tools and environment supported in software engineering;
- Develop basic understanding of requirement engineering to gather requirements for developing a system;
- Create design of a system by understanding its core concepts;
- Construct the system by understanding different coding techniques;
- Introduce the concepts of verification and validation.

Course Contents:

Introduction to Software Process Models; Programming in the Large vs. Individual Programming; Evaluation of Software Process Models; Requirements Analysis and Design Modeling Tools; Testing Tools; Programming Environments that Automate Parts of Program Construction Processes; Tool Integration Concepts and Mechanisms; Functional Requirements; Properties of Requirements; Software Requirements Elicitation; Describing System Data; Non-Functional Requirements; Requirements Specifications; System Design Principles; Design Paradigms; Structural and Behavioral Models of Software Designs; Design Patterns; Relationships between Requirements and Designs; Software Architecture; Refactoring Designs using Design Patterns; The Use of Components in Design; Coding Practices; Coding Standards; Integration Strategies; Verification and Validation; Inspections; Reviews; Audits; Testing Types; Testing Fundamentals; Defect Tracking; Limitations of Testing.

Recommended Books:

1. Software Engineering: A Practitioner's Approach, Pressman, R.S., McGraw Hill, 2015.
2. Software Engineering, Somerville, I., Pearson, 2015.
3. Object-Oriented Software Engineering Using UML, Patterns, and Java, Bruegge, B., & Dutoit, A. H., Pearson, 2010.

Course Code: CSC322

Course Title: Operating Systems

Credit Hours: 3(2, 1)

Pre-Requisite: CSC211

Course Objectives:

- To discuss the services provided by, and the design of an operating system;
- To explain the structure and organization of the file system and memory management;
- To discuss what a process is and how processes are synchronized and scheduled as well as how access to system resources is managed;
- To present the use of system calls for managing processes, memory and the file system;
- To explain the data structures and algorithms used to implement an OS;
- To explain security and protection issues in computer systems;
- To use C and UNIX commands to develop various system programs under Linux to make use of OS concepts related to process synchronization, shared memory, mailboxes, file systems, etc.

Course Contents:

Overview: Operating System Role, Purpose and Functionality; Computing Environments: Single User, Multi User, Multiple Simultaneous Computations; Goals of Parallelism (e.g., Throughput) versus Concurrency (e.g. Controlling access to Shared Resources); Parallelism; Communication and Coordination; Operating Systems Structure; Design Issues; Influences of Security; Networking; Multimedia; Windowing Systems; Operating Systems Principles: Structuring Methods (e.g. monolithic, layered, modular, micro-kernel models), Abstractions, Processes, and Resources, Concepts of APIs, Evolution of Hardware/Software Techniques and Application Needs, Device Organization, Interrupts (method, implementations), Concepts of User/System State and Protection, Transition to Kernel Mode; Concurrency: States and State Diagrams, Structures (ready list, process control block etc.), Dispatching and Context Switching, The Role of Interrupts, Managing Atomic Access to OS Objects, Implementing Synchronization Primitives, Multiprocessor Issues; Scheduling and Dispatch: Preemptive and Non- Preemptive Scheduling, Schedulers and Policies, Processes and Threads, Deadlines and Real Time Issues; Memory Management: Review of Physical Memory and Memory

Management Hardware, Memory Management Techniques, Working Sets and Thrashing, Caching; Security and Protection: Overview of System Security, Policy/Mechanism Separation, Security Methods and Devices Protection, Access Control and Authentication, Backups, File Systems; I/O Management; Disk Management, Data Race.

Recommended Books:

1. Operating System Concepts, Siblerschatz & Galvin, Addison-Wesley, 2012.
2. Modern Operating Systems, Tanenbaum, A. S., Prentice Hall, 2011.
3. Operating Systems, Stallings, W., Prentice Hall, 2012.

Course Code: CSC339

Course Title: Data Communications and Computer Networks

Credit Hours: 3(2, 1)

Pre-Requisite: None

Course Objectives:

- To discuss the network components, services and technologies;
- To describe the layered architecture of network protocols (e.g. TCP/IP) and explains core functions of each layer including addressing, routing, internetworking, switching, multiplexing, error and flow control, medium access and coding, Wireless and mobile networks;
- To discuss threats to network security and design of secure networks;
- To develop an understanding with the implementation of fundamental concepts of networking.

Course Contents:

Introduction: Organization of the Internet, Switching Techniques (Packet, Circuit), Physical Components of a Network (Hosts, Routers, Switches, ISPs, Wireless, LAN Access Points and Firewalls), Layering Principles (Encapsulation, Multiplexing, Roles of Different Layers); Networked Applications: Naming and Address Schemes (DNS, IP Addresses, Uniform Resource Identifiers), Distributed Applications, HTTP as an Application Layer Protocol, Multiplexing with TCP and UDP, Sockets APIs; Reliable Data Delivery: Error Control, Flow Control, Performance Issues (Pipelining), TCP: End to End versus Network, Assisted Approach, Fairness, Principles of Congestion Control, Approaches to Congestion; Routing and Forwarding: Routing versus Forwarding, Static Routing, Internet Protocol, Scalability (Hierarchical Addressing); Local Area Networks: Multiple Access Problem, Common Approaches to Multiple Access Problem, Local Area Networks, Ethernet, Switching; Resource Allocation: Need for Resource Allocation, Fixed Allocation, Introduction to Mobility: Wireless and Mobile Networks, 802.11 network; Security: Threats to Networks.

Recommended Books:

1. Computer Networking: A Top-Down Approach, Kurose, J.F. & Ross, K.W., Addison-Wesley, 2000.
2. Computer Networks and Internets, Comer, D.E., Addison-Wesley, 2014.
3. Networking: A Beginner's Guide, Hallberg, B., McGraw Hill, 2013.

Course Code: CSC356

Course Title: Human Computer Interaction

Credit Hours: 3(2, 1)

Pre-Requisite: None

Course Objectives:

- To explain the human factors to be considered in the design of human computer interaction;
- To define different processes for designing interfaces for different contexts;
- To define and implement user-centered approach in software development process and apply suitable techniques for collecting user requirement and analyzing tasks;
- To discuss the evaluation and comparison of user interfaces using different techniques such as laboratory experiments and expert reviews;
- To apply different techniques learned throughout the course on a practical project.

Course Contents:

Contexts for HCI; Processes for User-Centered Development; Different Measures for Evaluation (e.g., utility, efficiency, learnability, user satisfaction); Usability Heuristics and Principles of Usability Testing; Physical Capabilities that Inform Interaction Design (color, perception, ergonomics); Cognitive Models that Inform Interaction (movement and memory, attention, perception and gulfs of expectation and execution); Social Models that Inform Interaction Design (culture, communication, networks and organizations); Principles of Good Design and Good Designers; Engineering Tradeoffs; Accessibility (e.g. interfaces for visually impaired people); Interfaces for Different Age Groups (children, older people); Principles of GUIs; Elements of Visual Design; Task Analysis; Low Fidelity Prototyping; Quantitative Evaluation Techniques; Handling Human/System Failure; User interface Standards; Social Engineering (e.g., Phishing).

Recommended Books:

1. Designing Interactive Systems: A Comprehensive Guide to HCI, UX and Interaction Design, Benyon, D., Pearson, 2013.
2. Human Computer Interaction, Dix, A., Pearson, 2003.
3. Designing the User Interface: Strategies for Effective Human-Computer Interaction, Shneiderman, B., Pearson, 2009.
4. Interaction Design beyond Human Computer Interaction, Rogers, Y., Wiley, 2011.

Course Code: CSC371

Course Title: Database Systems I

Credit Hours: 4(3, 1)

Pre-Requisite: CSC241

Course Objectives:

- To explain the basic database concepts, information retrieval and relational theory;
- To develop the relational data model;
- To develop an enterprise data model that reflects the organization's fundamental business rules;
- To apply normalization techniques;
- To discuss the basics of transaction management, concurrency controls, query mechanisms, security and quality issues;
- To use and apply database programming languages and physical database design to gain experience in term project.

Course Contents:

Introduction to Databases and Information Systems along with their Applications; Information Storage and Retrieval; Information Capture and Representation; Approaches to and Evolution of Database Systems; Components of Database System; DBMS Functions; Database Architecture and Data Independence; Relational Algebra & Calculus; Data Modeling; Conceptual Models; Relational Data Model; E-R Models; Enhanced E-R Models; Object-Oriented Models; Spreadsheet Models; Semi-Structured Data Model (expressed using DTD or XML Schema, for example); Mapping Conceptual Schema to Relational Schema; Entity and Referential Integrity; Use of a Declarative Query Language; Structured Query Language (SQL); PL/Object-Oriented Programming; Functional Dependency & Normalization; Transaction Management; Query Mechanisms; Approaches for Managing Large Volumes of Data; Analysis and Indexing; Security and Quality Issues: Reliability, Scalability, Efficiency, and Effectiveness in Databases.

Recommended Books:

1. Database Systems: A Practical Approach to Design, Implementation, and Management, Connolly, T. M., & Begg, C. E., Addison-Wesley, 2009.
2. Fundamentals of Database Systems, Elmasri, R. & Navathe, S.B., Addison-Wesley, 2014.

Course Code: CSC498

Course Title: Senior Design Project I

Credit Hours: 2(0, 2)

Pre-Requisites: HUM102, CSC241, CSC291 & CSC371

Course Objectives:

- To employ the knowledge gained from courses throughout the program such as development of requirements, designing and documentation;
- To develop the project plan, software requirement specification document and software design document for a complex real world problem;
- To enhance communication, presentation and writing skills.

Course Contents:

This course is designed as capstone project, which requires students to demonstrate technical and presentation skills at levels which are commensurate with professional software engineering practices. It is desirable that students will apply their knowledge of computing throughout the course such as development of requirements, design, implementation and quality assurance, develop a software solution to a real-world problem from conception to completion. In this part of the project, students develop the project plan, software requirement specification document and software design document.

Recommended Books:

Books will be recommended by the faculty member supervising the capstone project

Course Code: CSC499

Course Title: Senior Design Project II

Credit Hours: 4(0, 4)

Pre-Requisite: CSC498

Course Objectives:

- To implement the design produced in CSC498 along with testing and evaluation of a complex real-world project in the area of Computer Science;
- To enhance presentation, communication and technical writing skills;
- To establish the ability to become an effective team player.

Course Contents:

The capstone project design course is a prominent element of the computing degree program and is central to the development of student professional competencies. This is the second part of a two-semester, senior-year capstone project. Student teams employ the knowledge gained from courses throughout the program such as development of requirements, design, implementation, and quality assurance to develop a software solution to a real-world problem from conception to completion. In this part of the project, students implement the design they produced in CSC498, test their code, and evaluate their final product.

Recommended Books:

Books will be recommended by the faculty member supervising the capstone project

Computing General Education Courses

Course Code: CSC101

Course Title: Introduction to ICT

Credit Hours: 3(2, 1)

Pre-Requisite: None

Course Objectives:

- To provide basic understanding of information and communication technologies (ICTs);
- Discuss the four main functions of computer hardware: input, processing, output, and storage;
- Identify and describe major hardware components;
- Develop understanding of the basics of storage devices, number systems, machine cycle, and microcomputer processor;
- Identify, describe and use communications and networking terminology further include Internet operations and its uses;
- Describe the major operating system functions and demonstrate usage of operating system services;
- Explain programming and application software;
- Discuss databases and e-commerce concepts;
- Understanding of IT security and other related issues.

Course Contents:

Basic Definitions; Concepts and History of Computers; The Parts of Computer Hardware: Computer Systems & Components, Storage Devices, Machine Level Representation of Data, Number Systems, Machine Cycle, and Microcomputer Processor; Software: Operating Systems, Programming and Application Software; Introduction to Programming; Introduction to Databases and Information Systems; Networks; Data Communication; The Internet: Browsers and Search Engines, Email, Collaborative Computing and Social Networking; E-Commerce; Foundational Concepts in IT Security and other Related Issues.

Recommended Books:

1. Introduction to Computers, Peter, N., McGraw Hill, 2013.
2. Using Information Technology: A Practical Introduction to Computer & Communications, Stacey, W.S & Sawyer, C., McGraw Hill, 2010.

Course Code: CSC110

Course Title: Professional Practices for IT

Credit Hours: 3(3, 0)

Pre-Requisite: None

Course Objectives:

- Develop an understanding of the basic cultural, social, legal, and ethical issues inherent in the discipline of Computing;
- Highlighting the use and significance of professional ethics;
- Discuss intellectual property and privacy rights;
- To professionally communicate and evaluate formal documents;
- Explain the consequences of computing on individuals, organizations, and society.

Course Contents:

Social Context: Social Implications of Computing and Networked Communication, Impact of Social Media on Individualism and Collectivism; Analytical Tools: Ethical Argumentation, Theories and Decision Making, Moral Values; Professional Ethics: Community Values, Nature of Professionalism, Self-assessment, Professional Certification (Such as ACM/IEEE-CS, SE, AITP), Accountability, Responsibility and Liability, Role of Computing Professional in Public Policy; Intellectual Property: Philosophical Foundations of Intellectual Property, Intellectual Property Rights, Intangible Digital Intellectual Rights, Digital Rights Management, Discrimination and Harassment, Forms of Professional Credentialing, Copyrights, Patents, Trade Secrets, Trademarks, Plagiarism, Open Source Movement; Privacy and Civil Liberties/Human Rights: Philosophical Foundations, Legal Foundations of Privacy Protection, Privacy Implications of Widespread Data Collection, Surveillance Systems and Cloud Computing, Technology based Solutions for Privacy Protection, Privacy Legislation in Areas of Practice, Civil Liberties/Human Rights, Cultural & Religious Differences, Freedom of Expression and its Limitations; Sustainability: How to be a Sustainable Practitioner, The global, Social and Environmental Impacts of Computer Use and Disposal; IEEE CS/ACM Code of Ethics and Professional Practice.

Faisal

Recommended Books:

1. Ethics in Information Technology, Reynolds, G., Cengage, 2014.
2. Ethical and Social Issues in Information Age, Kizza, J. M., Springer, 2013.

Computer Science Core Courses

Course Code: CSC312

Course Title: Theory of Automata

Credit Hours: 3(3, 0)

Pre-Requisite: CSC102

Course Objectives:

- To explain various models of automata and language theory;
- To present the classification of finite state machines and their equivalence;
- To discuss the generation of context free grammar for a specific language;
- To develop an understanding of computation through Turing machine;
- To discuss the theory of un-decidable problems.

Course Contents:

Regular languages: Closure Properties, Pumping Lemma; Regular Expressions; Finite State Machines: DFA, NFA, Equivalence of DFA and NFA; Concept of Halting Problem; Context-Free Grammars: Parsing and Ambiguity, Properties of CFG, Normal Forms, Applications, Designing CFG for a Specific language, Push Down Automata; Conversion among Equivalently Powerful Notations i.e. DFA, NFA and Regular Expressions and between PDA and CFG; Turing Machine: A Formal Model for Universal Computation, Nondeterministic Turing Machine, Halting Problem of Turing Machine;

Recommended Books:

1. Introduction to Automata Theory, Languages, and Computation, Hopcroft, J., Motwani, R. & Ullman, J., Addison-Wesley, 2007.
2. Introduction to Computer Theory, Cohen, D.I.A., Wiley, 2004.

Course Code: CSC301

Course Title: Design and Analysis of Algorithms

Credit Hours: 3(3, 0)

Pre-Requisite: CSC211

Course Objectives:

- To develop an ability to analyze the asymptotic performance of algorithms;
- To discuss rigorous correctness proofs for algorithms;
- To explain the major algorithms and data structures;
- To apply important algorithmic design paradigms and methods of analysis;
- To highlight the significance of NP complete problems.

Course Contents:

Problem solving: using Loop Invariants to Show Algorithms, Correctness; Asymptotic Notations: Worst, Best and Average Case Behavior of Algorithms; Big O notation; Complexity Classes i.e. Constant, Linear, Quadratic; Empirical Measurements of Performance; Time and Space Tradeoffs in Algorithms; Recurrence Algorithms; Analysis of Iterative and Recurrence Relations; Master Theorem; Divide and Conquer; Recursive Backtracking; Worst Case Quadratic Sorting Algorithms, Worst or Average Case Sorting Algorithms (Quick, Heap & Merge Sort); Representation of Graphs, Depth First and Breadth First Traversal; Brute Force Algorithms; Greedy Algorithms; Approximation Algorithms; Dynamic Programming; Branch-and-Bound Techniques; Heuristics; Reductions: Transform and Conquer; Basic Computability: The Complexity Classes P and NP; Introduction to NP Complete Problems.

Recommended Books:

1. Introduction to Algorithms, Cormen, T. H., Leiserson, C.E., Rivest, R.L., & Stein, C., MIT Press, 2009.
2. An Introduction to the Analysis of Algorithms, Sedgewick, R. & Flajolet, P., Addison-Wesley, 2012.

Course Code: CSC462

Course Title: Artificial Intelligence

Credit Hours: 3(2, 1)

Pre-Requisite: CSC102

Course Objectives:

- Describe the fundamentals of Artificial Intelligence;
- Implement an appropriate uninformed/informed search algorithm for a problem and characterize its time and space complexity;
- Translate natural language sentences (e.g. English) into logic statements;
- Convert logic statements into a clause form and apply resolution to a set of logic statements to answer a query;
- Explain the basic machine learning tasks and techniques.

Course Contents:

Overview of AI Problems; Intelligent Behavior: Turing Test, Rationale versus Non-rationale Reasoning; Problem Characteristics: Fully versus Partially Observable, Single versus Multi agent; Nature of Agents: Autonomous versus Semi-autonomous, Reflexive, Goal-based and Utility-based; Problem Spaces; Uninformed Search: Depth First, Breadth First, Depth First with Iterative Deepening; Informed Search: Hill climbing, A*- Search and their Time and Space Complexity, Local Search, Genetic Algorithm; Propositional and Predicate Logic; Resolution and Theorem Proving; Forward and Backward Chaining; Machine Learning: Classification, Inductive Learning, Simple Statistical Based Learning, Decision Tree, and ANN.

Recommended Books:

1. Artificial Intelligence: A Modern Approach, Russell, S., and Norvig, P., Prentice Hall, 2013.
2. Prolog Programming for Artificial Intelligence, Bratko, I., Addison-Wesley, 2012.

Course Code: CSC441

Course Title: Compiler Construction

Credit Hours: 3(2, 1)

Pre-Requisite: CSC312

Course Objectives:

- Explain the theory and principles behind language translators;
- To present the overview of automata theory, context free grammar and related tools for the compiler construction;
- Identify and handle various design issues of several programming languages;
- Construct and customize a compiler for a simple language.

Course Contents:

Program Representation: Interpreters, Assemblers, Compilers, Type-checkers, Documentation, Abstract Syntax Trees (contrast with concrete syntax), Data Structures to Represent Code for Execution, Translation or Transmission; Language Translation and Execution: Interpretation vs. Compilation to Native Code vs. Compilation to Portable Intermediate Representation, Language Translation Pipeline: Scanning, Parsing, Type Checking, Translation, Linking, Execution (virtual machine, just-in-time), Run-time Representation of Core Language Constructs, Run-time Layout of Memory, Memory Management; Syntax Analysis: Scanning (lexical analysis), Using Regular Expressions, Parsing Strategies and Role of Context Free Grammars, Generating Scanners and Parsers from Declarative Specifications; Semantic Analysis: High-Level Program Representations such as Abstract Syntax Trees, Scope and Binding Resolution, Type Checking, Declarative Specifications such as Attribute Grammars; Code Generation: Procedure Calls and Method Dispatching, Separate Compilation, Instruction Selection and Scheduling, Register Allocation, Peephole Optimization.

Recommended Books:

1. Compiler Construction: Principles and Practice, Louden, K.C., Cengage Learning, 2006.
2. Compilers: Principles, Techniques and Tools, Aho, A.V., Pearson, 2010.
3. Concepts of Programming Languages, Sebesta, R., Addison-Wesley, 2012.
4. Engineering a Compiler, Cooper & Torczon, Morgan Kaufmann, 2011.

Course Code: CSC321

Course Title: Microprocessor and Assembly Language

Credit Hours: 3(2, 1)

Pre-Requisite: EEE241

Course Objectives:

- To explain the basic characteristics of a microprocessor and its applications;
- To present the basic architecture of the IA-32 processor;
- To provide a comprehensive understanding of 80X86 instruction set;
- To develop an understanding of the basic steps of assembling, linking and executing an assembly program;
- To solve a given problem by writing programs in assembly language.

Course Contents:

Machine Level Representation of Data: Bits, Bytes and Words, Numeric Data Representation and Number Bases, Fixed- and Floating-point Systems, Signed and Two's-complement Representations; Assembly Level Machine Organization: Basic Organization of the Von Neumann Machine, Basic Microcomputer Design, Control Unit, Instruction Fetch, Decode and Execution, Instruction Sets and Types (Data Manipulation, Control, I/O), Instruction Formats, Addressing Formats, Subroutine Call and Return Mechanisms, I/O and Interrupts (Interrupts introduction: Hardware/Software interrupts, Interrupt handling routine, PC interrupt handling), IA-32 Processor Architecture; Memory System Organization and Architecture: I/O Fundamentals; Handshaking, Buffering, Programmed I/O, Interrupt-driven I/O; Interfacing and Communication: Buses, Bus Protocols, Arbitration, Direct-memory Access (DMA), Memory Hierarchy, CPU Performance Calculation, Actions Inside the Processing Chip, Multimedia Support, RAID Architectures; Assembly/Machine Language Programming: Basic Elements of Assembly Language, Assembly Language Program Development Cycle, Instruction Formats, Data Declaration and Addressing Modes, Intel x86 Instruction-set Related to Arithmetic Expressions, Logical Operations, Conditional and Repetitions, I/O Handling, Procedures, Macros and Interrupt Handling.

Recommended Books:

1. Assembly Language for x86 Processors, Irvine, K.R., Prentice Hall, 2014.
2. Computer Organization and Architecture, Stallings, W., Prentice Hall, 2012.

Course Code: CSC336

Course Title: Web Technologies

Credit Hours: 3(2, 1)

Pre-Requisite: CSC241

Course Objectives:

- To explain the fundamental concepts of web architectures and its applications;
- To provide an understanding of Search Engine Optimization (SEO);
- To provide an understanding of planning, designing and publishing a multi-page website;
- To provide hands-on experience of client-side technologies such as HTML, JavaScript, CSS;
- To provide practical experience of development of dynamic clients using modern development technologies, for example PHP, AJAX, MySQL.

Course Contents:

Overview of Web Platforms; Web Architecture; Tier Architecture; Web Standards and Constraints; Client-side Technologies: HTML Documents, HTML Tags, Use of Tables, CSS and its Working, Form Creation and Uses; Scripting Languages: Client and Server Side Script, JavaScript, AJAX; Server-side Technologies: Dynamic Web Page Creation, Server-side programming using PHP, Use of Databases in Web-based Applications, Database Queries, Use of Web APIs; Programming Via Platform-Specific APIs; Programming Under Platform Constraints.

Recommended Books:

1. PHP and MySQL Web Development, Welling, L. & Thomson, L., Addison-Wesley, 2013.
2. The Modern Web: Multi-Device Web Development with HTML5, CSS3, and JavaScript, Gasston, P., No Starch Press, 2013.

Computer Science Supporting Areas Courses

Course Code: CSC475

Course Title: Numerical Computing

Credit Hours: 3(3, 0)

Pre-Requisite: MTH104

Course Objectives:

- To present the concepts of efficiency, reliability and accuracy of a method and theory of differences;
- To apply numerical methods to obtain approximate solutions to mathematical problems;
- Derive numerical methods for various mathematical operations and tasks, such as interpolation, polynomial approximation, numerical differentiation, integration, the solution of linear and nonlinear equations, and the solution of differential equations.

Course Contents:

The Concepts of Efficiency, Reliability and Accuracy of a Method; Minimizing Computational Errors; Theory of Differences; Difference Operators; Difference Tables; Forward Differences; Backward Differences; and Central Differences. Mathematical Preliminaries; Solution of Equations in One Variable; Interpolation and Polynomial Approximation; Numerical Differentiation; and Numerical Integration; Initial Value Problems for Ordinary Differential Equations; Direct Methods for Solving Linear Systems; Iterative Techniques in Matrix Algebra; Solution of Non-Linear Equations.

Recommended Books:

1. Numerical Methods in Scientific Computing, Germund, D. Åke, B., McGraw Hill, 2008.
2. Numerical Methods for Scientific Computing, J. H. Heinbockel, McGraw Hill, 2006.

Computer Science Electives

Track I: Software Development

Course Code: CSC303

Course Title: Mobile Application Development

Credit Hours: 3(2, 1)

Pre-Requisite: CSC241

Course Objectives:

- Discuss different mobile application development platforms and architectures;
- Discuss the components of mobile application development;
- Compare different mobile application development tools;
- Describe the constraints that game platforms impose on developers;
- Develop a medium sized mobile application as a team.

Course Contents:

Introduction to Mobiles Application Development Platform; HTML5 for Mobiles; Android OS: Architecture, Framework and Application Development; iOS: Architecture, Framework; Application Development with Windows Mobile; Eclipse; Fragments; Calling Built-in Applications using Intents; Displaying Notifications; Components of a Screen; Adapting to Display Orientation; Managing Changes to Screen Orientation; Utilizing the Action Bar; Creating the User Interface; Listening for UI Notifications; Views; User Preferences; Persisting Data; Sharing Data; Sending SMS Messages; Getting Feedback; Sending E-mail; Displaying Maps; Consuming Web Services Using HTTP; Web Services: Accessing and Creating; Threading; Publishing Android Applications; Deployment on App Stores; Mobile Programming Languages; Challenges with Mobility and Wireless Communication; Location-aware Applications; Performance/Power Tradeoffs; Mobile Platform Constraints; Emerging Technologies.

Recommended Books:

1. iOS Programming: The Big Nerd Ranch Guide, Conway, J., Hillegass, A., & Keur, C., Big Nerd Ranch Guides, 2014.
2. Android Programming: The Big Nerd Ranch Guides, Phillips, B. & Hardy, B., Big Nerd Ranch Guides, 2014.

Course Code: CSC412

Course Title: Visual Programming

Credit Hours: 3(2, 1)

Pre-Requisite: CSC241

Course Objectives:

- To explain the visual concepts of event-driven programming;
- To focus on graphical user interfaces in windows environment, program structure, language syntax, and implementation details for the development of visual programming based applications;
- To concentrate on the recent usage of platforms used in the development;
- To enable students through hands-on practice to develop small-scale applications.

Course Contents:

Visual Programming Basics; Introduction to Events; Fundamentals of Event-drive Programming; Introduction to C#: Basic C# Language Constructs; Object Oriented Programming in C#: Properties, Interfaces and Indexers, Delegates, C# Events, Exception Handling, Attributes, Enums, Operator Overloading; Reading and writing XML; Working with Files and Directories, Introduction to WPF and XAML; Property Elements; Type Converters; Markup Extensions; XAML and Procedural Code, Logical and Visual Trees, Dependency Properties, Attached Properties and Element Display, Transforms and Panels, Grid Panel, Content Overflow, Panel Composition, Input Events, Touch Manipulation Events, WPF Commands; Deploying & Installing; Navigation-base Apps, XAML Browser Apps, Resources, Data binding (ADO.NET and LINQ to SQL), Collection View, Hierarchical Data Template, Introduction to Threads, Tasks and Continuations, Asynchronous Functions, Parallel Programs, Concurrent Collections, Introduction to HTML, CSS, and JavaScript, DOM and the jQuery Library, jQuery Events, AJAX Programming in JavaScript.

Recommended Books:

1. Microsoft Visual C# 2013 Step by Step (Step by Step Developer), Sharp, J., Microsoft Press, 2013.
2. Beginning C# Object-Oriented Programming (Expert's Voice in .NET), Clark, D., Apress Publisher, 2013.

Course Code: CSC495

Course Title: Game Development

Credit Hours: 3(3, 0)

Pre-Requisite: CSC241

Course Objectives:

- To discuss the leading trends in game development;
- To describe the concepts related to game play, game flow, interactive narratives, storytelling and apply them in practice to develop an addictive game;
- To practice main tools (game editors) available to develop games and to identify the best one for a specific purpose.

Course Contents:

History of Computer and Video Games; Types of Game Platforms (e.g., XBox, Wii, PlayStation); Game Design Principles; Game Platform Languages (e.g., C++, Java, Lua, Python); Game Platform Constraints; Pygame; Storytelling; Sprites; Animation; Game Development Methodologies; Physics; Loose Ends; Audio; Sound; Music; 2D Game Group Project Check-In; Game Testing; Ethics; MMORPGs; Securing Online Games; Game Engines; iOS Development; Cocos2D; Leading Trends in Games Development.

Recommended Books:

1. Agile Game Development with Scrum, Keith, C., Addison-Wesley, 2010.
2. AI for Game Developers, Bourg, D. & Seemann, G., O'Reilly Media, 2004.
3. The Art of Game Design: A Book of Lenses, Schell, J., Morgan Kaufmann, 2008.
4. Fundamentals of Game Design, Adams, E., New Riders, 2010.

Course Code: CSC494

Course Title: Software Project Management

Credit Hours: 3(3, 0)

Pre-Requisite: CSC291

Course Objectives:

- To discuss different software project management phases;
- To prepare a project plan for a software project that includes estimates of size and effort, a schedule, resource allocation, configuration control, change management, and project risk identification and management;
- To compare different methods and techniques used to assure the quality of a software product;
- To explain an approach to risk that will help to secure the on-time delivery of software;
- To demonstrate the use of the MS-project as tool for software project management.

Course Contents:

Overview of Software Project Management; Classic Mistakes; PMI Process Groups, Software Project Phases; Project Charter, Statement of Work (SOW); Planning Phase: Development Life Cycle Models; Matching Life Cycles of Projects; Project Plans; Work Breakdown Structures (WBS); Estimation of Effort and Cost (Expert Judgment, FP and Use Case point methods); Scheduling: Project network diagram fundamentals, CPM, PERT, Gantt charts, Critical chain scheduling, Using MS-Project, Assigning Resources; Resource leveling; Team Models: Team Participation, Responsibilities for Tasks, Meeting Structure, Work Schedule, Roles and responsibilities, Team Conflict resolution, Risks Associated with Virtual Teams (communication, perception, structure), Managing Conflict and Motivating; Project Monitoring and Control: Status Reporting, Project Metrics, EVM, Communications Techniques; Risk Management: Risk Categories including Security, Safety, Market, Financial, Technology, People, Quality, Structure and Process; Change Control; Project Recovery; Cutover/Migration; Post Project Reviews; Closing.

Recommended Books:

1. Information Technology Project Management, Schwalbe, K., Cengage Learning, 2013.
2. Software Project Management, Hughes, B. & Cotterell, M., McGraw Hill, 2009.

Course Code: CSC392

Course Title: Software Design Methodologies

Credit Hours: 3(3, 0)

Pre-Requisite: CSC291

Course Objectives:

- To articulate the principles and elements of software design;
- To analyze software design from the perspective of significant design quality attributes;
- To discuss and select appropriate software architecture for a medium-sized system suitable for a given scenario;
- To explain the role of objects in middleware systems and the relationship with components;
- To apply component-oriented approaches to the design of a range of software systems;
- To apply the principles of secure design.

Course Contents:

Design Principles; Internal Design Qualities & Models; External Design Qualities & Models; Design Modeling using UML; Software Architecture; Contexts of Software Architecture; Architectural Patterns and Tactics; Software Architectural Styles; Component Level Design; Component Based Development; Pattern Based Software Design; Component Level Design Patterns; User Interface Design Patterns; Web App Design Patterns; Application Frameworks; Middleware; Principles of Secure Design & Coding.

Recommended Books:

1. Software Engineering, A Practitioner's Approach, 8th Edition, Roger S Pressman, McGraw-Hill 2015.
2. Software Architecture in Practice, 3rd Edition, Len Bass, Paul Clements, Rick Kazman, Addison-Wesley, 2013.
3. Software Design Methodology, Hong Zhu, Elsevier, 2005.

Track II: Network and Security

Course Code: CSC334

Course Title: Parallel and Distributed Computing

Credit Hours: 3(2, 1)

Pre-Requisite: CSC103

Course Objectives:

- To explain the fundamental concepts of parallel and distributed computing along with its benefits and limitations;
- To provide an understanding of basic concepts of parallel and distributed systems paradigms: Grid Computing, Cloud Computing, cluster and Peer-to-Peer Computing;
- To develop an understanding of the application of parallel and distributed algorithms in problem solving;
- To provide hands-on experience of distributed and parallel programming paradigms using open MPI.

Course Contents:

Parallel and Distributed Architectures and Systems; Parallel and Distributed Programming Paradigms; Parallel and Distributed Algorithms; and other Applications of Parallel and Distributed Computing; Multi-core; Client-server; Clouds; Grids; Peer-to-peer Systems; Parallel and Distributed Architectures; Open MP; MPI; Pthreads; Distributed Objects; States and Causality in Distributed Systems; Coordination and Agreement; Distributed Transactions; Fault Tolerance; Data Parallel Languages; Map Reduce; Parallel Debugging, and Applications of Parallel and Distributed Computing.

Recommended Books:

1. Principles of Parallel Programming, Lin, C. & Snyder, C., Addison-Wesley, 2008.
2. Distributed Systems: Concepts and Design, Coulouris, G., Dollimore, J. & Kindberg, T., Addison-Wesley, 2011.
3. Java: How to Program, Deitel, H. M., & Deitel, P. J., Prentice Hall, 2005.
4. Java in Distributed Systems, Concurrency, Distribution and Persistence, Boger, M., Wiley, 2001.

Course Code: CSC432

Course Title: Information Security

Credit Hours: 3(3, 0)

Pre-Requisite: CSC339

Course Objectives:

- Provide familiarity with prevalent network and distributed system attacks, defenses against them, and forensics to investigate the aftermath;
- Develop an understanding of cryptography, how it has evolved, and some key encryption techniques used today;
- Develop an understanding of security policies (such as authentication, integrity and confidentiality), as well as protocols to implement such policies in the form of message exchanges.

Course Contents:

Basic Notions of Confidentiality; Integrity, Availability, Authentication Models; Threats and Attacks: Attacker Goals, Capabilities, and Motivations (Cyberwarfare, Insider threats, Hacktivisim, Advanced Persistent Threats), Examples of Malware (e.g. Viruses, worms, spyware, botnets, Trojan horses or rootkits), Denial of Service and Distributed Denial of Service; Cryptography: Basics, Cipher Types together with Typical Attacks, Public Key Infrastructure, Support for Digital Signatures, Encryption and its Challenges, Mathematical Preliminaries Essential for Cryptography, Cryptographic Primitives, Pseudo-Random Generators & Stream Ciphers, Block Ciphers, Pseudo-Random Functions, Hash Functions, Audit, Intrusion Detection & Response, Database Security, Host Based and Network-Based Security Issues; Operational Security: Physical Security Issues, Personnel Security, Policy Formation and Enforcement, Access Controls, Information Flow, Legal and Social Issues; Identification and Authentication in Local and Distributed Systems; Classification and Trust Modeling, Risk Assessment.

Recommended Books:

1. Principles of Information Security, Whitman, M., & Mattord, H., Course Technology, 2012.
2. Introduction to Computer Security, Goodrich, M., & Tamassia, R., Pearson, 2013.
3. Security in Computing, Pfleeger, C.P., Pfleeger, S.L. & Margulies, J., Prentice Hall, 2015.

Course Code: CSC482

Course Title: Systems and Networks Administration

Credit Hours: 3(2, 1)

Pre-Requisites: CSC322 & CSC339

Course Objectives:

- To explain the role and responsibilities of a system administrator;
- To provide an understanding about configuration of network protocols such as DNS, DHCP, FTP, LDAP;
- To provide a comprehensive understanding of administration tasks including installing, configuring, securing, and troubleshooting the devices and services;
- To provide hands-on experience of the use of tools and techniques for system/network administration.

Course Contents:

Overview: Installation and Up-gradation of Popular Operating Systems such as Linux, Basic Linux Commands; File Management: Managing File Systems, Permissions, Managing User Accounts, Setting up X-Windows system; Configuration: Configuring Printing Services, Backing-up Data, Tuning Kernel Parameters; Configuring and Managing Various Protocols: DNS, DHCP, Routing, Electronic Mail, and Network File System; Managing and Troubleshooting Computer Systems and Networks; Network and System Security.

Recommended Books:

1. Red Hat Certified System Administrator & Engineer: Training Guide, Ghori, A., Endeavor Technologies Inc, 2012.
2. UNIX and Linux System Administration Handbook, Nemeth, E., Snyder, G., Hein, T.R. & Whaley, B., Prentice Hall, 2010.
3. UNIX System Administration Handbook, Nemeth, Snyder, Seebass & Hein., Prentice Hall, 2001.
4. TCP/IP Network Administration, Hunt, C., O' Reilly Media, 2002.

Course Code: CSC430

Course Title: Wireless Networks

Credit Hours: 3(2, 1)

Pre-Requisite: None

Course Objectives:

- To introduce the fundamentals of the wireless communications and networks;
- To explain the techniques underlying modern physical layer wireless communications;
- To discuss the various medium access and resource allocation techniques such Aloha and CSMA based randomized medium access, scheduling for TDMA/FDMA/CDMA-based wireless networks;
- To present the network layer routing protocols;
- To discuss the transport layer protocols, with an emphasis on congestion control, including TCP over wireless, congestion sharing mechanisms, explicit and precise rate control;
- To discuss the evaluation of MAC and network protocols using network simulation software tools such as NS-2 or Qualnet.

Course Contents:

Introduction to Wireless Network Architectures: Cellular Networks, Wireless Local Area Networks, Multi-hop Networks, Radio Propagation Models, Narrowband Digital Modulation; Basics of CDMA and OFDM, Diversity and MIMO, Equalization; Randomized Medium Access: Un-slotted and Slotted Aloha, System Throughput Analysis, CSMA; Window Adaptation Mechanisms, Graph Coloring and its Application to Channel Allocation in TDMA/FDMA/CDMA-based Wireless Networks under the Protocol Model, Introduction to Wireless Network Simulator (NS-2/QualNet), Introduction to Multi-hop Wireless Network Routing, The AODV and OLSR Protocols for Mobile Ad-hoc Networks, Link Estimation and Neighbor Management; Geographic Routing: Greedy Routing, Avoiding Routing Holes, Opportunistic Routing and Cooperative Routing; TCP: TCP over Wireless Networks, UDP, Congestion Sharing (IFRC, WCAP).

Recommended Books:

1. Wireless Networks, Smith, C. & Collins, D., McGraw Hill, 2014.
2. Wireless Communication Networks and Systems, Beard, C. & Stallings, W., Pearson, 2015.
3. Wireless Communications, Molisch, E.R., Wiley, 2010.

Track III: Database Technologies

Course Code: CSC402

Course Title: Database Systems II

Credit Hours: 3(3, 0)

Pre-Requisite: CSC371

Course Objectives:

- Describe fundamental theories and requirements that influence the design of modern database systems;
- To explain the background processes involved in query processing & optimization, concurrent transactions, recovery and security of databases;
- To present the architecture and implementation of distributed and replicated database systems;
- To discuss the concepts related to emerging database technologies.

Course Contents:

Object-Oriented Model; Object-Relational Model; Transaction Processing: Transactions Failure & Recovery and Concurrency Control Techniques; Database Backup &Recovery; Query Processing and Optimization; Database Security; Distributed Database Design: Distributed Data Storage, Distributed Query Processing, Distributed Transaction Processing and Concurrency control; Emerging Database Technologies.

Recommended Books:

1. Fundamentals of Database Systems, Elmasri, R., & Navathe, S.B., Addison-Wesley, 2010.
2. Principles of Distributed Database Systems, Ozsu, M.T., & Valduriez, P., Springer, 2011.
3. Database Systems: A Practical Approach to Design, Implementation, and Management, Connolly, T.M., & Begg, B., Addison-Wesley, 2009.

Faiddi

Course Code: CSC496

Course Title: Data Warehousing and Data Mining

Credit Hours: 3(2, 1)

Pre-Requisite: CSC371

Course Objectives:

- To explain the concept of collecting and integrating data arising from different sources to build a data warehouse;
- To present the knowledge of architectures and implementations of a data warehouse, as well as the guidelines to be followed in its maintenance;
- To introduce the dimensional modeling techniques for the design of the data warehouse;
- Application of data warehouse design in a warehouse building tool and performing analysis using OLAP tool;
- To explain the aspects of different data mining tasks: classification, association rules and clustering;
- To analyze data mining tasks using UCL datasets in a data mining tool such as WEKA.

Course Contents:

Requirements Gathering for Data Warehousing; Data Warehouse Architecture; Dimensional Model Design for Data Warehousing; Physical Database Design for Data Warehousing; Extracting; Transforming;& Loading Strategies; and Introduction to Business Intelligence & OLAP Tool; Introduction to Data Mining Concepts; Data Mining Tools; Data Mining Tasks: Classification, Association Rule Mining and Clustering.

Recommended Books:

1. Data Mining: Concepts and Techniques, Han, J., Kamber, M. & Pei, J., Morgan Kaufmann Publishers, 2012.
2. Data Warehousing Fundamentals for IT Professional, Ponniah, P., Wiley, 2010.
3. Introduction to Data Mining, Tan, P. N., Steinbach, M. & Kumar, V., Pearson, 2013.

Course Code: CSC471

Course Title: Distributed Database Systems

Credit Hours: 3(3, 0)

Pre-Requisite: CSC371

Course Objectives:

- Understanding the need of Distributed Databases (DDBS);
- Understanding fragmentation and its different types along with replication in DDBS;
- Understanding query processing and optimization along with administration issues in DDBS, failure recovery, transaction management and concurrency control;
- Development of concepts in parallel databases, object distributed databases, and multi-databases.

Course Contents:

Introduction to Distributed Database Systems (DDBSs); Database and Networking Concepts; Distributed DBMS Architecture; Distributed Database Design; Semantic Data Control; Distributed Query Processing; Query Decomposition and Data Localization; Optimization of Distributed Queries; Transaction Management and Distributed Concurrency Control; Distributed DBMS Reliability; Parallel Database Systems; Distributed Object Database Management Systems; Database Interoperability; Current Issues in DDBSs; Use of the MapReduce Processing Model, Data.

Recommended Books:

1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, Addison-Wesley, 2010.
2. Principles of Distributed Database Systems, M. Tamer Özsu and Ptrick Valduriez, Pearson, 2011.

Course Code: CSC461

Course Title: Introduction to Data Science

Credit Hours: 3(3, 0)

Pre-Requisite: MTH231 & MTH262

Course Objectives:

- Identify and classify data science problems;
- Understanding of the skills required for data analytics at massive levels – scalable data management on and off the cloud;
- Understanding of the basic techniques of data science, including both SQL and NoSQL solutions for massive data management (e.g., Hadoop, MapReduce and contemporaries);
- Understanding of the algorithms for data mining (e.g., clustering and association rule mining), and basic statistical modeling (e.g., linear and non-linear regression) in the context of big data.

Course Contents:

Data Science Articulated: History, Context, and Technology Landscape; Data Manipulation at Scale; Parallel Databases; Parallel Query Processing; Data Engineering: Sharding, Hadoop, MapReduce and proto buffers; Algorithms and Languages; Key-Value Stores and NoSQL, Tradeoffs of SQL and NoSQL; Topics in Statistical Modeling: Basic Concepts, Experiment Design, Pitfalls; Machine Learning: Supervised Learning (Rules, Decision Trees, Forests, Nearest Neighbor, and Regression), Optimization (Gradient Descent and Variants); Unsupervised Learning; Dat Visualization & Data Journalism,; Data Products; Visual Data Analytics; Provenance; Privacy; Ethics; Governance; Graph Analytics: Structure, Traversals, Analytics; PageRank; Community Detection; Recursive Queries; Semantic Web; Social Network Analysis.

Recommended Books:

1. Mining of Massive Datasets, Anand Rajaraman and Jeffrey D. Ullman, Cambridge University Press, 2011.
2. Big Data Analytics Beyond Hadoop: Real-Time Applications with Storm, Spark, and More Hadoop Alternatives, Vijay Srinivas Agneeswaran, Pearson, 2014.

Faisalabad

Course Code: CSC347

Course Title: Computational Intelligence

Credit Hours: 3(3, 0)

Pre-Requisite: CSC241

Course Objectives:

- To introduce the fundamental theory and concepts of computational intelligence methods;
- Discuss in particular fuzzy systems, genetic algorithms, particle swarm intelligence and their applications in the area of machine intelligence;
- Describe the basics of an evolutionary computing paradigm.

Course Contents:

Introduction to Computational Intelligence; Applicability and History; Fundamentals of Genetic Algorithms: Encoding, Fitness Function, Tournament Selection, Truncation Selection, Elitist Selection, Crossover, Mutation, Control Parameters Estimation, Parallel Genetic Algorithms; Fundamentals and Background of Particle Swarm Optimization Techniques; Discrete PSO; Hybrid PSO (HPSO); Adaptive PSO (APSO); Fundamentals of Ant Colony Search Algorithms; Behavior of Real Ants; The Max-Min Ant System; Use of Greedy Search and Constructive Heuristic Information; Fundamentals of Tabu Search; Neighborhood Structure; Characterization of the Neighborhood; The Use of Long-Term Memory in Tabu Search; Fundamentals of Simulated Annealing; Cooling Schedule; Determination of Cooling Rate; Stopping Criterion; Fuzzy Systems; Creation of the Fuzzy Control; Evolutionary Algorithms; Differential Evolution; Key Operators for Differential Evolution.

Recommended Books:

1. Computational Intelligence: An Introduction, Engelbrecht, A.P., Wiley, 2007.
2. Modern Heuristic Optimization Techniques: Theory and Applications to Power Systems, Lee, K.Y., &El-Sharkawi, M.A., IEEE Press Series on Power Engineering Publication, 2008.

Track IV: Artificial Intelligence and Graphics

Course Code: CSC354

Course Title: Machine Learning

Credit Hours: 3(3, 0)

Pre-Requisite: None

Course Objectives:

- To present the basic machine learning concepts;
- To present a range of machine learning algorithms along with their strengths and weaknesses;
- To apply machine learning algorithms to solve problems of moderate complexity.

Course Contents:

Machine Learning Tasks: Classification; Inductive Learning; Simple Statistical based Learning: Naïve Bayesian Classifier, Decision Trees, Parameter Estimation; Measuring Classifier Accuracy; Inductive Logic Programming; Supervised Learning: Decision Trees, Neural Networks, Support Vector Machines; Unsupervised Learning and Clustering: EM, K-means, Self-Organizing Maps (SOM); Semi-Supervised Learning; Reinforcement Learning: Exploration vs. Exploitation Trade-off, Markov Decision Processes, Value and Policy Iteration; Learning Theory; The problem of Over Fitting; Applications of Machine Learning Algorithms for Data Mining.

Recommended Books:

1. Machine Learning, Tom, M., McGraw Hill, 1997.
2. Pattern classification, Duda, R.O., Hart, P.E., &Stork, D.G., Wiley, 2001.
3. Pattern Recognition and Machine Learning, Bishop, C., Springer-Verlag, 2006.
4. Bioinformatics: A Machine Learning Approach, Baldi, P., & Brunak, S., Cambridge, MA: MIT Press. 2002.

Course Code: CSC451

Course Title: Introduction to Modeling and Simulation

Credit Hours: 3(2, 1)

Pre-Requisites: CSC102 & CSC211

Course Objectives:

- To explain the basic principles of Modeling and Simulation (M&S) methodologies considering both theoretical and practical aspects;
- To introduce simulation methods and tools for discrete event systems;
- To apply verification and validation on simple simulation problems.

Course Contents:

Introduction to Simulation; General Principles; Simulation Software; Statistical Models; Queuing Models; Random Number Generation; Random Variate Generation; Input Modeling; Verification and Validation; Output Analysis of a Single Model; Comparison and Evaluation; Applications: Manufacturing Systems, Computer Systems, Computer Networks; Develop a Small-Scale M&S Project.

Recommended Books:

1. Discrete-Event System Simulation, Banks J., Carson II, J. S., Nelson, B. L. and Nicol, D. N., Prentice Hall, 2009.
2. Discrete-Event Simulation: Modeling, Programming and Analysis, Fishman, G. S., Springer-Verlag, 2010.
3. Applied Simulation Modeling, Seila, A.F., Ceric, V. & Tadikamalla, P., Thomson Learning Inc, 2013.

Course Code: CSC331

Course Title: Digital Image Processing

Credit Hours: 3(3, 0)

Pre-Requisite: MTH231, MTH262

Course Objectives:

- To explain the image formation process;
- To discuss the basic image acquisition process, and its representation;
- To explain the basic image processing techniques in both spatial and frequency domains;
- To present the usage of these techniques on small (workable) image segments;
- To explain various segmentation approaches, along with their characteristics, differences, strengths, and weaknesses;
- To explain the need for different image morphological algorithms and transformations.

Course Contents:

Introduction: Elements of Digital Image Processing, Image Model, Sampling and Quantization, Relationships between Pixels; Image Enhancement: Enhancement by Point Processing, Spatial Filtering, Enhancement in the Frequency Domain, Color Image Processing, Image Segmentation: Discontinuity Detection, Edge Linking and Boundary Detection, Thresh Holding, Region Oriented Segmentation, Use of Motion for Segmentation; Image Registration: Introduction to Image Registration, Techniques of Image Registration; Representation and Description: Boundary Description, Regional Description; Morphological Image Processing: Dilation and Erosion, Opening and Closing, Some basic Morphological Algorithms, Extensions to Gray Level Images; Image Transforms: Discrete Fourier Transform, Discrete Cosine Transform, Haar Transform, Hadamard Transform.

Recommended Books:

1. Digital Image Processing, Gonzalez, R.C. & Woods, R.E., Pearson 2008.
2. Computer Vision: A Modern Approach, Forsythand, D.A., & Ponce, J., Pearson, 2011.

Course Code: CSC353

Course Title: Computer Graphics

Credit Hours: 3(2, 1)

Pre-Requisite: MTH231

Course Objectives:

- To introduce the fundamental concepts of computer graphics;
- To explain the graphics pipeline approach;
- To provide an understanding of visualization of 2D & 3D objects using geometric primitives;
- To provide an understanding of basic concepts of projection of 3D objects and camera calibration;
- To explain the fundamental aspects of geometric modeling (e.g. curves & surfaces);
- To develop an understanding of animation techniques related to computer graphics.

Course Contents:

Overview of Graphics Systems: Graphics Primitives, Digitization of Analog Data, Resolution, and the Limits of Human Perception, e.g., Pixels for Visual Display, Dots for Laser Printers, and Samples for Audio, Standard Media Formats, including Lossless and Lossy Formats, Additive and Subtractive Color Models (CMYK and RGB) and Why these Provide a Range of Colors, Interactive Input Methods, Animation as a Sequence of Still Images; Basic Rendering: Rendering in Nature, e.g., the Emission and Scattering of Light and its Relation to Numerical Integration, Forward and Backward Rendering, Polygonal Representation, Basic Radiometry, Application of Spatial Data Structures to Rendering, Sampling and Anti-aliasing; Scene Graphs and the Graphics Pipeline; Visualization: Two-Dimensional Geometric, Transformations & Viewing, Three-Dimensional Transformations, Projections, Camera Concepts, Modeling, Graphical User Interfaces, Lighting, Shading & Textures, Hidden Surfaces, Programming Raster Display Systems, Differential Line Algorithm, Panning and Zooming, Raster Algorithms and Software — Scan-Tradeoffs Between Storing Data and Re-computing Data as Embodied by Vector and Raster Representations of Images, Converting Lines, Characters and Circles, Scaling, Rotation, Translation, Region Filling and Clipping; Small-Scale Project.

Recommended Books:

1. Computer Graphics with OpenGL, Hearn, D.D, M., Baker, M.P., & Carithers, W., Prentice Hall, 2010.
2. Computer Graphics: Principles and Practice, Hughes, J. F., Dam, A.V., Foley, J. D., Feiner, S.K., and Akeley, K., Addison-Wesley, 2010.

Course Code: CSC455

Course Title: Computer Vision

Credit Hours: 3(3, 0)

Pre-Requisite: MTH231& MTH262

Course Objectives:

- To provide an introduction to basic concepts and methodologies for recognition and extraction of features from raster images;
- To provide a foundation for developing applications and for further study in the field;
- To gain practical experience in the design and implementation of image processing algorithms.

Course Contents:

Concepts Behind Computer-Based Recognition and Extraction of Features from Raster Images; Applications of Vision Systems and their Limitations; Overview of Early, Intermediate and High Level Vision; Motion Analysis; Segmentation: Region Splitting and Merging, Quadtree Structures for Segmentation, Mean and Variance Pyramids, Computing the First and Second Derivatives of Images using the Sobel and Laplacian Operators, Grouping Edge Points into Straight Lines by Means of the Hough Transform, Limitations of the Hough Transform, Parameterization of Conic Section; Perceptual Grouping: Failure of the Hough Transform, Perceptual Criteria, Improved Hough Transform with Perceptual Features, Grouping line Segments into Curves; 3D Vision; Triangulation Principle; Stereoscopy; Audio and Speech Recognition.

Recommended Books:

1. Computer Vision: A Modern Approach 2nd ed, David Forsyth, Jean Ponce, Prentice Hall, 2011.
2. Computer Vision, Linda G. Shapiro, George C. Stockman, Prentice Hall, 2001.

Faisalabad

Course Code: CSC454

Course Title: Pattern Recognition

Credit Hours: 3(3, 0)

Pre-Requisite: CSC354

Course Objectives:

- To introduce the fundamentals of pattern recognition;
- To present the generative methods such as Bayes decision theory, parameter and density estimation;
- To explain the discriminative methods such as support vector machines and nearest neighbor classification;
- To discuss pattern recognition applications;
- To focus on explaining computer vision as a major application of pattern recognition.

Course Contents:

Introduction to Pattern Recognition; Bayesian Decision Theory; Bayesian Networks; Maximum Likelihood Estimation; Dimensionality Reduction; Feature Selection; Bayesian Estimation; Linear Discriminant Functions; Support Vector Machines (SVMs); Expectation-Maximization (EM) Algorithm; Non-parametric Estimation.

Recommended Books:

1. Pattern Classification, R. O. Duda, P. E. Hart, and D. G. Stork, Wiley-Interscience, 2001.
2. Machine Learning: A probabilistic Perspective, K. Murphy, MIT Press, 2012.

Course Code: CSC421

Course Title: Robotics

Credit Hours: 3(3, 0)

Pre-Requisite: None

Course Objectives:

- To present the fundamentals of robotic systems;
- To explain the capabilities and limitations of various state-of-the-art robots ;
- To provide an understanding of robot control architectures;
- To explain the theory of robotics navigation and control, path and motion planning;
- To discuss multiple robot coordination and feedback control strategies;
- To introduce the probabilistic robotics.

Course Contents:

Overview: Defining Robotics, Types, Uses, Brief History, Key Components, Applications, State-of-the-Art Robot Systems, Robot Control Architectures: Deliberative and Reactive, Inherent Uncertainty in Sensing & Control; Interpreting Uncertain Sensor Data; Basic Imaging for Robotics: Coordinate Transformations; Sensing: Sensors; Effectors and Actuators; Localizing and Mapping; Navigation and Control; Path Planning; Motion Planning; Kinematics: Direct and Inverse; Feed forward and feedback control; Multiple Robot Coordination; Introduction to Probabilistic Robotics.

Recommended Books:

1. Introduction to Robotics, 3rd Edition, John. J. Craig, Pearson, 2013.
2. Probabilistic Robotics, Sebastian Thrun, MIT Press, 2005.
3. Principles of Robot Motion: Theory, Algorithms, and Implementations, Howie Choset, MIT Press, 2004.