Revised Course Structure Academic Year 2016-17

Semester-I				Total Marks	
Course Code	Course Title	No. of Credits	No. of Hrs./Week	Internal	External
CSI401	Constitution of India	2	2	10	40
CSI402	Research Methodology	2	2	10	40
CSI403	Programming Core Java	3	3	20	80
CSI404	Computer System Architecture	3	3	20	80
CSI405	Operating System	3	3	20	80
CSI406	Information Theory	3	3	20	80
CSI451	Practical based on CSI403	2	4	-	50
CSI452	Practical based on CSI404	2	4	-	50
CSI453	Practical based on CSI 405	2	4	-	50
CSI454	Practical based on CSI406	2	4	-	50
Total No. of Credits		24	-	-	-

Semester-II				Total Marks	
Course Code	Course Title	No. of Credits	No. of Hrs./Week	Interna l	External
CSI407	Research Project Review Writing	1	1	25	-
CSI408	Interactive Programming using Python	3	3	20	80
CSI409	Software Engineering and CASE tools	3	3	20	80
CSI410	Data Structure	3	3	20	80
CSI411	Computer Network	3	3	20	80
CSI412	Relational Database Management System using MySQL	3	3	20	80
CSI455	Practical based on CSI408	2	4	-	50
CSI456	Practical based on CSI409	2	4	-	50
CSI457	Practical based on CSI 410	2	4	-	50
CSI458	Practical based on CSI411	2	4	-	50
CSI459	Practical based on CSI412	2	4	-	50
Total No. of Credits		26	-	-	-

Semester-III				Total Marks	
Course Code	Course Title	No. of Credits	No. of Hrs./Week	Internal	External
CSI501	Advanced Programming Using Python	3	3	20	80
CSI502	Data Warehousing Using MySQL	3	3	20	80
-	Elective –I (Generic)	3	3	20	80
CSI551	Project/Dissertation Part –I	12	24	100	200
CSI552	Practical based on CSI501	2	4	-	50

CSI553	Practical based on CSI 502	2	4	-	50
-	Practical based on Elective-I	2	4	-	50
Total No. of Credits		27	-	-	-

Semester-IV				Total Marks	
Course Code	Course Title	No. of Credits	No. of Hrs./Week	Internal	External
CSI503	Data Mining	3	3	20	80
-	Elective –II (Discipline Centric)	3	3	20	80
CSI554	Project/Dissertation Part -II	12	24	100	200
CSI555	Seminars	1	2	-	25
-	Service Course	3	3	20	80
CSI556	Practical based on CSI503	2	4	-	50
-	Practical based on Elective-II	2	4	-	50
Total No. of Credits		26	-	-	-

Elective-I				Total Marks	
Course Code	Course Title	No. of Credits	No. of Hrs./Week	Internal	External
CSI521	Remote Sensing	3	3	20	80
CSI522	Image Processing	3	3	20	80
CSI523	Cyber Law & Cyber Crime	3	3	20	80
CSI524	Network Security	3	3	20	80
CSI525	Cloud Computing	3	3	20	80
CSI526	Mobile Computing	3	3	20	80
CSI527	ERP	3	3	20	80
CSI557	Practical Based on CSI521	2	4	-	50
CSI558	Practical Based on CSI522	2	4	-	50
CSI559	Practical Based on CSI523	2	4	-	50
CSI560	Practical Based on CSI524	2	4	-	50
CSI561	Practical Based on CSI525	2	4	-	50
CSI562	Practical Based on CSI526	2	4	-	50
CSI563	Practical Based on CSI527	2	4	-	50

Elective-II				Total Marks	
Course Code	Course Title	No. of Credits	No. of Hrs./Week	Internal	External
CSI528	Programming J2ME	3	3	20	80
CSI529	Open Source Web Programming Using PHP	3	3	20	80
CSI530	VB.Net Using My SQL	3	3	20	80
CSI531	GIS	3	3	20	80
CSI532	Biometrics	3	3	20	80
CSI533	Android Programming	3	3	20	80
CSI534	Human Computer Interaction	3	3	20	80

CSI564	Practical Based on CSI528	2	4	-	50
CSI565	Practical Based on CSI529	2	4	-	50
CSI566	Practical Based on CSI530	2	4	-	50
CSI567	Practical Based on CSI531	2	4	-	50
CSI568	Practical Based on CSI532	2	4	-	50
CSI569	Practical Based on CSI533	2	4	-	50
CSI570	Practical Based on CSI534	2	4	-	50

Total (I+II+III+IV) Semester (24+26+27+26) Credits = 103

Service Courses:

- The student should opt service course of 3 credits either from parent department or from other department.

Service Courses				Total Marks	
Course Code	Course Title	No. of Credits	No. of Hrs./Week	Internal	External
CSC541	Communication Skills	3	3	20	80
CSC542	Introduction to MATLAB	3	3	20	80
CSC543	Web Developments	3	3	20	80
CSC544	Personality Development	3	3	20	80
CSC545	Aptitude Development	3	3	20	80
CSC546	Android Programming	3	3	20	80
CSC547	Intellectual Property Rights	3	3	20	80

Detailed Syllabus

Semester - I

1. Research Methodology

Course Code	**CSI402**	**Course Title**	**Research Methodology**
Number of Credits	**2 Credits (TH)**	**Internal**	**10**
Total Contact Hours	**2 HRS (TH/Week)**	**External (Semester/Term Exam)**	**40**

Course Outcomes:

- Critically analyze research methodologies identified in existing literature.
- Propose and distinguish appropriate research designs and methodologies to apply to aspecific research project.
- Develop a comprehensive research methodology for a research question.
- Choose appropriate quantitative & qualitative method to collect data.
- Apply the understanding of feasibility and practicality of research methodology for a system

Unit-1:

Introduction: Meaning, Concept, nature steps types and characteristics of research, Identification & formulation of Research Problem, Hypothesis, Research Design & Research Ethics.**Review of literature**:Need for Reviewing Literature, what to Review and for what purpose, Literature search Procedure, Sources of Literature, Planning of Review work, Note Taking.

Unit-2:

Types and Methods of Research: Classification of Research, Pure and Applied Research, Exploring or Formulative Research Descriptive Research, Diagnostic Research / Study, Evaluation Research / studies, Action Research, Experimental Research, Analytical study of statistical Method, Historical Research, Surveys, Case Study, Field Studies.

Unit-3:

Development of research proposal: Research proposal and its elements, Formulation of research problem - criteria of sources and definition, Development of objectives and characteristics of objectives, Development hypotheses and applications.

Unit-4:

Methods & tools of data collection:

Concept of sampling and other concepts related to sampling. Probability and non - probability samples, their characteristics and implications.Tools of data collections, their types, attributes and uses.Redesigning, research tools - like questionnaire, opinnaere, observation, interviews, scales and tests etc.**FieldWork**The Nature of Field Work, Selection and Training of Investigators, Sampling Frame and Sample Selection, Field Operation, Field Administration.

Unit-5:

Methods of data analysis: Editing, Classification and Coding, Transcription, Statistical Analysis, Measures of Central Tendency Measures of Dispersion, Measures of Association / Relationship, Regression and Correlation Analysis, Hypothesis Testing (For Proportion and Means), Test of Significance.**Report writing and evaluations:**Types of Reports, Planning of Report Writing, Research Report Format, Principles of Writing, Documentation, Data and Data Analysis reporting in a Thesis, Writing of Report, Typing of Report, Briefing. Use of Anti-plagiarism software and its importance.

References:

1) Bajpai S. R., (1975), Methods of Social Survey and Research, Kitabghar, Kanpur.
2) Bhattacharya D. K., (2004), Research Methodology, New Delhi, Excel Books.
3) Brymann Alan and Carmer D., (1995), Qualitative data analysis for social / scientist, New York, Routledge publication.
4) Best J. W. and Khan J. V., (2005), Research in Education New Delhi, Prentice Hall India. Hans Raj (19gg) Theory and practice in Social Research, Surjeet publication, Kolhapur.
5) Chandera A. and Saxena T. P., (2000), Style Manual, New Delhi, Metropolitan Book Comp. Ltd.
6) Krishnaswami O. R., (1988), Methodology of Research in Social Science, Himalaya pub. House.
7) Kothari, C. R., (2005), Quantitative Technique, New Delhi, Vikas publication House.
8) Gautam N. C., (2004), Development of Research tools, New Delhi, Shree Publishers.
9) Gupta, Santosh, (2005) Research Methodology and statistical Techniques, Deep and Deep publications.
10) Shukla J. J., (1999) Theories of Knowledge, Ahmadabad, Karnavati Publication.

2. Programming in Core Java

Course Code	**CSI403**	**Course Title**	**Programming Core Java**
Number of Credits	**3 Credits (TH), 2 Credits (PR)**	**Internal**	**20**
Total Contact Hours	**3 HRS (TH/Week) 4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course Outcomes:

- Good JAVA Programmer.
- Useful for NET/SET Examination.
- Useful for JAVA Certification.
- Useful for Applets and Apps Development

Prerequisite:

Before attending this course, students must have:

- Student should know the Object oriented programming Concepts

  – Or –

- Understanding of the basics of structured programming, including concepts such as flow control, variables and parameters, and function calls

At Course Completion:

After completing this course, students will be able to:

- Create a simple Java Application based on the Java Foundations.
- Use of Java forms and controls to create a user interface.
- Create and use variables and arrays.
- Create and use Class, Interfaces, Packages and predefined utility classes.
- Implement decision structures and loops by using conditional expressions.
- Validate user input for fields, controls, and forms.
- Apply object-oriented programming techniques to create classes, add methods, and add behavior to the classes
- Resolve syntax, run-time, and logic errors by using the structured exception handling.
- Enhance the user interface by adding menus, status bars, and toolbars.

Course Outline

Unit-1:

A. Java Introduction:The Java Environment – Overview, Writing a Java Program, Obtaining The Java Environment, Setting up your Java Environment, Creating a Class That Can Run as a Program, The main() Method, Useful Stuff Necessary to go Further, System.out.println(), Using the Java Documentation, B. Java Basics:Basic Java Syntax: General Syntax Rules, Java Statements, Blocks of Code, Comments, Variables: Data types, Primitive Data Types, Object Data Types, Literal Values, Constants and the final keyword, Mathematics in Java: Expressions, Operator Precedence, Multiple Assignments, Order of Evaluation, Bitwise operators, Compound Operators, Expressions that Mix Data Types: Typecasting Creating and Using Methods, Creating Methods, Variable Scope

Unit-2:

A. Java Objects: Objects: Object-Oriented Languages, Object-Oriented Programs, Encapsulation, Creating and Using an Instance of an Object, References Defining a Class, Constructors, Method Overloading, The this Keyword, static Elements, Garbage Collection, Java Packages, Dealing with Keyboard Input, String, StringBuffer, and StringBuilder, Creating Documentation, Comments and Using javadoc, Javadoc Comments B. Comparisons And Flow Control Structures: Controlling Program Flow: Boolean-Valued Expressions, Complex Boolean Expressions, Simple Branching, Two Mutually Exclusive Branches, Nestedif...elseStatements- Comparing a Number of MutuallyExclusive Options, Comparing a Number of Mutually ExclusiveOptions- The switchStatement, Comparing Objects, Conditional Expression, while and do. . .while Loops, for Loops, Additional Loop Control: break and continue, Breaking Out of a Loop, Continuing a Loop, Classpath, Code Libraries, and JAR files, Using CLASSPATH Creating a jar File (a Library) C. Arrays and Vectors: Arrays : Defining and Declaring Arrays, Instantiating Arrays, Initializing Arrays, Working With Arrays, Array Variables, Copying Arrays, Arrays of Objects, Enhanced for Loops - the For-Each Loop, Multi-Dimensional Arrays, Multidimensional Arrays in Memory, Example - Printing a Picture, Typecasting with Arrays of Primitives, Using Vectors: Defining Vectors and using Vectors.

Unit-3:

A. Inheritance: Inheritance: Derived Class Objects, Polymorphism, Inheritance and References Dynamic Method Invocation, Creating a Derived Class, Inheritance and Access Inheritance and Constructors - the super Keyword, Derived Class Methods That Override Base Class Methods Inheritance and Default Base Class Constructors, The Instantiation Process at Runtime, Typecasting

with Object References: Typecasting, Polymorphism, and Dynamic Method Invocation, More on Overriding, Object Typecasting Example, Checking an Object's Type: Using instanceof, Typecasting with Arrays of Objects, Other Inheritance-Related Keywords: abstract, final, Methods Inherited from Object B. Packages and Interfaces: Interfaces: Creating an Interface Definition, Implementing Interfaces: Implementing Interfaces – Example, Reference Variables and Interfaces, Calling an Interface Method, Interfaces and Inheritance: Some Uses for Interfaces, Interfaces and Event-HandlingInterfaces and "Pluggable Components", Packages: Creating and using packages, Access C. Inner Classes: Inner Classes, Nested Classes, Inner Class Syntax, Instantiating an Inner Class Instance from Within the Enclosing Class, Inner Classes Referenced from Outside the Enclosing Class Working with Inner Classes D. Exceptions: Exceptions: Handling Exceptions, Exception Objects: Attempting Risky Code - try and catch, Guaranteeing Execution of Code - the finally Block, Letting an Exception be Thrown to the Method Caller, Throwing an Exception, Exceptions and Inheritance, Exception Class, Constructors and Methods, Creating and Using Your Own Exception Classes, Rethrowing Exceptions, Initializer Blocks, Static Initializer Blocks, Assertions.

Unit-4:
A. Utility Classes: Collection Interfaces, Concrete collections, Collections framework, Algorithms, Legacy Collections Streams: Output Streams, Input Streams, Filter Streams, Readers and Writers B. Threads: Thread Class and Runnable Interface, Thread SynchronizationC. I/OPackage: InputStream and OutputStreamclasses,Reader and Writer classesD. Java Networking:InetAddress, URL, URLConnection, TCP/IP Server Socket, Client Socket, User Datagram Sockets, D. Applet and Swings: Applet: Applet Life Cycle, Passing Parameters to Applet, Delegation Event Model, AWT Components, AWT Events, using listeners, Working with Graphics, Loading Image and Multimedia objects in applet

Unit-5:
Java Database Connectivity: Java Database Connectivity Architecture, JDBC-ODBC Bridge, JDBC Drivers, JDBC API, Classification of JDBC classes, Driver Interface, DriverManager Class, Connection, Statement, ResultSet, Implementing Stored Procedures.

Reference Books:

- Java 2 Complete Reference by Herbert Schieldt (Sixth Edition)
- Core Java Vol 1: Sun Press, Eighth Edition
- Core Java Vol 2: Sun Press

E-books:

- Java 2 Complete Reference by Herbert Schieldt (Fourth Edition)

Lab Exercise:
CSI451 Practical based on CSI403:There should be minimum 10 lab assignment on the topics discussed in the course.

3. Computer System Architecture

Course Code	**CSI404**	**Course Title**	**Computer System Architecture**

Number of Credits	**3 Credits (TH)** **2 Credits (PR)**	**Internal**	**20**
Total Contact Hours	**3 HRS (TH/Week)** **4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course Outcomes:

- After complication of this course students will be able to understand and able to design circuit and simplify it. Will be able to understand in detail the how logic systems are built. Students will have thorough knowledge aboutBasic structure of a digital computer Arithmetic operations of binary number system The organization of the Control unit, Arithmetic and Logical unit, Memory unit and the I/O

Prerequisite:

- Student must aware of basics of digital electronics, microprocessor architecture, ALP.

At Course Completion:

After complication of this course students can be interested to do the hardware knowledge.

Course Outline:

Unit-1:

Data Representation: Introduction, Data types, Complements, Fixed Point and Floating Point representation, Error Detecting Codes.

Unit-2:

Digital Logic Circuits: Digital Computers, Logic Gates, Boolean algebra, Map Simplification,Combinational Circuits, Flip-Flops Sequential Circuits.

Unit-3:

Digital Components: Integrated Circuits, Decoders, Multiplexers, Registers, Shift Registers, BinaryCounters, Memory Unit.

Unit-4:

Central Processing Unit: Introduction, general Register Organization, Stack Organization, InstructionFormats, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer (RISC).

Unit-5:

I/O& Memory Organization: Peripheral Devices, Input Output Interface, Asynchronous Data Transfer, Modes of Transfer, Direct Memory Access, Input-Output Processor, and Serial Communication, Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory, Memory Management Hardware.

Reference Book:

1. Computer System Architecture-M. Morris Mano, PHI Publication, ISBN-81-203-0855-7. 3rd Ed.

2. Computer Organization and Architecture- William Stallings – Fifth Edition.
3. Structured Computer Education – Andrew S Tanenbaum- (Prentice Hall)-Fourth Edition.

E-Books:
http://www.tutorialspoint.com/computer_logical_organization/

Lab Exercise:
CSI452 Practical based on CSI404: At least two experiments should be carried out on each unit.

4. Operating System

Course Code	**CSI405**	**Course Title**	**Operating System**
Number of Credits	**3 Credits (TH)** **2 Credits (PR)**	**Internal**	**20**
Total Contact Hours	**3 HRS (TH/Week)** **4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course outcomes:

- Students who complete this course successfully are expected to: Gain extensive knowledge on principles and modules of operating systems.
- Understand key mechanisms in design of operating systems modules
- Understand process management, concurrent processes and threads, memory management, virtual memory concepts, deadlocks
- Compare performance of processor scheduling algorithms
- Produce algorithmic solutions to process synchronization problems
- Use modern operating system calls such as Linux process and synchronization libraries.

Prerequisite:
Student must know Computer System Architecture in order to understand functioning of Operating system and must have knowledge of disk operating systems.

At Course Completion:
Students who complete this course successfully are expected to:

- gain extensive knowledge on principles and modules of operating systems
- understand key mechanisms in design of operating systems modules
- understand process management, concurrent processes and threads,memory management, virtual memory concepts, deadlocks
- compare performance of processor scheduling algorithms
- produce algorithmic solutions to process synchronizati on problems
- use modern operating system calls such as Linux process and synchronizati on libraries

Course Outline
Unit - 1:
Introduction:Definition of OS ,Types of OS :main Frame , Desktop, Multiprocessor, Distributed, Clustered, Real time,Multiprogramming,TimeSharing,Embedded OS. Overview of operating systems

responsibilities.Operating system components and structures.Desirable Qualities of OS.Process: Definition Processes and programs.Implicit and Explicit tasking, Independent and Co-operationprocess, Process relationship, Process States,Process State transitions,Process Control Block,Context switches,Threads: Necessity and Advantage ofThreads,Types of Threads. System Calls and System call Execution. Thread programming using OpenMP: OpenMP programming model,Specifying current task in OpenMP,SynchronizationConctructs in OpenMP,DataHandling,Libraryfunction,Environment variables. Process Scheduling:Objectives of scheduling,Types of Schedulers:Short,Long,Medium, Comparision between schedulers, Scheduling Criteria,Scheduling Algorithms: Types Preemptive and Non-Preemptive scheduling,FCFS,RR,SJF and Priority based Scheduling;Evaluation of Scheduling algorithms.

Unit - 2:

Interprocess Communication: Basic concepts,Shared Memory System, Message Passing:Direct versus Indirect Communication, Critical Sections, Race conditions, Mutual Exclusion.Semaphores:Definition,WAIT(S) and SIGNAL(S) instructions, Algorithm for WAIT and SIGNAL operations, Locks, Monitors and Condition variables.Deadlocks:Definition, Characteristics, A resource Allocation graph,Deadlockprevention,Deadlockavoidance:Banker's Algorithm, Deadlock Detection and Recovery.

Unit - 3:

Introduction:Overlays and Swapping, Logical and Physical address space ,Contiguous allocation methods, Single partition and multiple partition Systems, Relocation Memory Mangement.Paging:Principle of operation, Page allocation, Hardware support for paging, Paging address translation by direct mapping and associate mapping, Protection and sharing, Advantages and disadvantages of paging.Segmentation:Principle of operation, Address translation, Advantages and disadvantages of segmentation, Comparison between segmentation and Paging.VirtualMemory:Necessity, Hardware and control structures, Locality of reference, Page fault, Working set, Dirty page/Dirty bit, Demand paging, Thrashing, Page replacement Policies: FIFO, LRU, NRU.

Unit - 4:

I/O Management:I/O buffering ,Single and Double Buffer schemes, Disk Organization.FileManagement:FileConcepts,File descriptor, Access methods: Sequential, indexed and direct, File sharing, Protection, Access rights, File System structure, Byte Sequence, Record sequence and Tree-based, Recovery and Disk formatting.Secondary Storage Management: File allocation methods:Contiguous Allocation and Non Contiguous allocation, Chained, Indexed allocation, free space management, Disk Scheduling: FCFS, SSTF, SCAN and C- SCAN , Disk Cache.Protection and Security:System performance, protection and security, policies and methods, Access matrix

Unit - 5:

Introduction:History of Linux, Features of Linux, Drawbacks of Linux, Components of Linux, Memory Management Subsystems, Linux Process and Thread Management, File Management System, Device DriversLinux Commands and Utilities: cat, tail, cmp, diff, wc, sort, mkdir, cd, rmdir, pwd, cp, more, passwd, who, whoami, mv, chmod, kill, write, wall, merge , mail, pipes, filters and redirection utilities.ShellScripts:Creating and executing simple shell programs, variables, special

characters, comparison of expressions, iteration statements, conditional statements functions. System Administration:Installing Linux, Booting the system,Maintaining user accounts, File systems and Special Files,Backups and Restoration.

Books:

- Operating Systems Concepts, 8th edition, Abraham Silberschatz, Peter Baer Galvin, Greg Gagne; Wiley, ISBN 0-470-12872-0,2010.

Reference Books:

- Operating Systems: Internals and Design Principles, 6th edition, William Stallings; Prentice Hall, ISBN-10: 0136006329, Operating Systems, 3rd edition
- Modern Operating Systems, 3rd edition, Andrew S. Tanenbaum; Prentice Hall, ISBN-10: 0-13-600663-9, 2008.
- Using OpenMP,Portable Shared Memory Parallel Programming ,Barbara Chapman, Gabriele Jost and Ruud van der Pas , ISBN: 9780262533027 , 2007

Website:

- http://codex.cs.yale.edu/avi/os-book/OS8/os8c/slide-dir
- http://openmp.org/wp/resources/
- http://www.compunity.org/training/tutorials/3%20Overview_OpenMP.pdf

Lab Exercise:

CSI453 Practical based on CSI405:At least five experiments should be carried out on each unit.
By considering the sample experiment list as guidelines :

- Implementing a CPU scheduling policy in a Linux OS.
- Implementing a memory management policy in a Linux OS.
- Implementing a file system in a Linux OS.
- Process control system calls: The demonstration of fork, execve and wait system calls along with zombie and orphan states.
- Thread management using OpenMP API.: Thread execution ,Static scheduling, Dynamic scheduling , Synchronization Conctructs in OpenMP ,Data Handling,Libraryfunction,Environment variables.
- Thread synchronization using counting semaphores and mutual exclusion using mutex. Application to demonstrate: producer-consumer problem with counting semaphores and mutex.
- Deadlock Avoidance Using Semaphores
- Implement the deadlock-free solution to Dining Philosophers problem to illustrate the problem of deadlock and/or starvation that can occur when many synchronized threads are competing for limited resources.
- Demonstrate the following CPU Scheduling Algorithms
  a. FCFS b.SJFc.Priorityd.Round Robin
- Demonstrate all Page Replacement Algorithms
  a.FIFO b.LRU c.MRU
- Simulate Bankers algorithm for Deadlock Avoidance
- Simulate Bankers Algorithm for deadlock Prevention

5. Information Theory

Course Code	**CSI406**	**Course Title**	**Information Theory**
Number of Credits	**3 Credits (TH), 2 Credits (PR)**	**Internal**	**20**
Total Contact Hours	**3 HRS (TH/Week) 4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course Outcomes:

- The students will be able to understand and apply fundamental concepts in information theory such as probability, entropy, information content and their inter-relationships.
- Understand and explain the basic concepts of information theory, source coding, channel and channel capacity, channel coding and relation among them. Describe the real life applications based on the fundamental theory.
- Calculate entropy, channel capacity, bit error rate, code rate, steady-state probability and so on. Implement the encoder and decoder of one block code or convolutional code using any program language.
-

Course Outline:

Unit – 1

Information Theory: Introduction, Measure of information, Average information content of symbols in long independent sequences, Average information content of symbols in long dependent sequences. Mark-off statistical model for information source, Entropy and information rate of mark-off source.

Unit – 2

Source Coding: Encoding of the source output, Shannon's encoding algorithm. Communication Channels, Discrete communication channels, Continuous channels

Unit - 3

Fundamental Limits on Performance: Source coding theorem, Huffman coding, Discrete memory less Channels, Mutual information, Channel Capacity.

Unit - 4

Channel coding theorem, Differential entropy and mutual information for continuous ensembles, Channel capacity Theorem.

Unit - 5

Introduction to Error Control Coding: Introduction, Types of errors, examples, Types of codes Linear Block Codes: Matrix description, Error detection and correction, Standard arrays and table look up for decoding.

Unit – 6

Binary Cycle Codes, Algebraic structures of cyclic codes, Encoding using an (n-k) bit shift register, Syndrome calculation. BCH codes.

Unit - 7

RS codes, Golay codes, Shortened cyclic codes, Burst error correcting codes. Burst and Random Error correcting codes.

Unit - 8

Convolution Codes, Time domain approach. Transform domain approach.

Text Books:

1. Digital and analog communication systems, K. Sam Shanmugam, John Wiley, 1996.
2. Digital communication, Simon Haykin, John Wiley, 2003.

Reference Books:

1. ITC and Cryptography, Ranjan Bose, TMH, II edition, 2007
2. Digital Communications - Glover and Grant; Pearson Ed. 2nd Ed 2008

Semester - II

1. Research Project Review Writing

Course Code	**CSI407**	**Course Title**	**Research Project Review Writing**
No of Credits	**1 Credits** (TH)	**Internal**	**100%**
Total Contact Hrs./Week	**1 HRS** (TH/Week)	**External** (Semester end Exam.)	-

Course Outcomes:

- With this course, students will become familiar with and learn to identify the most relevant textbooks, reviews, papers and journals for their research topics. During the course the students will also learn how to critically read and assess research papers and reviews. The review should point to research gaps that can be operationalized into feasible research questions.

Course Guidelines:

1. What is a literature review?

The ability to review, and to report on relevant literature is a key academic skill. A literature review: situates your research focus within the context of the wider academic community in your field; reports your critical review of the relevant literature; and identifies a gap within that literature that your research will attempt to address.

1. Why do I need a literature review?

When readers come to your assignment, dissertation, or thesis, they will not just assume that your research or analysis is a good idea; they will want to be persuaded that it is relevant and that it was worth doing. They will ask questions such as: What research question(s) are you asking? Why are you asking it/them? Has anyone else done anything similar? Is your research relevant to research/practice/theory in your field? What is already known or understood about this topic? How might your research add to this understanding, or challenge existing theories and beliefs? These are questions that you will already probably be asking yourself.

3) A critical review

It is important that the literature review is more than just a list of references with a short description of each one. Merriam (1988:6) describes the literature review as: 'an interpretation and synthesis of published work'.

This very short statement contains some key concepts, which are examined in the table below.

A] Published work 's Concepts Explanation and its Associated critique

B]Interpretation.

C]Synthesis

1. Getting started

Reading anything on your research area is a good start.

According to Taylor and Procter (2008) of The University of Toronto have some useful suggested questions to ask yourself at the beginning of your reading:

1.What is the specific thesis, problem, or research question that my literature review helps to define?

2.What type of literature review am I conducting?

3.Am I looking at issues of theory? methodology? policy? quantitative research? qualitative research?

4.What is the scope of my literature review?

5.What types of publications am I using (e.g., journals, books, government documents, popular media)?

6.What discipline(s) am I working in (e.g., GIS,Biometrics ,Imageprocessing ,NLP)?

- **Ways of finding relevant material**

  Electronic sources.

  References of references

  Hand searching of journals
- **Collecting material**
- **Keeping a record**
- **Plagiarism Detection**
- **When to stop**

It is important to keep control of the reading process, and to keep your research focus in mind. Rudestam and Newton (1992:49) remind us that the aim is to 'Build an argument, not a library'. It is also important to see the writing stage as part of the research process, not something that happens after you have finished reading the literature.

1) Writing it up

The task of shaping a logical and effective report of a literature review is undeniably challenging. Some useful guidance on how to approach the writing up is given by Wellington et al (2005:87): "It should be framed by your research questions. It must relate to your study. It must be clear to the reader where it is going: keep signposting along the way. Wherever possible, use original source material rather than summaries or reviews by others.

1. Using tables

As well as using tables to display numerical data, tables can be useful within a literature review when you are comparing other kinds of material.

1. Reference list

Almost all academic writing will need a reference list. This is a comprehensive list of the full references of sources that you have referred to in your writing.

References

1. Ask Oxford (2006). Found at: http://www.askoxford.com/

Useful reading

1) Murray, R. 2003: How to survive your viva. Maidenhead: Open University Press.

2) Rugg, G. & Petre, M. 2004: The unwritten rules of PhD research. Maidenhead: Open University Press.

1. Tinkler, P. & Jackson, C. 2004: The doctoral examination process: a handbook for students, examiners and supervisors. The Society for Research into Higher Education. Maidenhead: Open University Press.
2. Wellington, J., Bathmaker, A., Hunt, C., McCulloch, G. & Sikes, P. 2005: Succeeding with your doctorate. London: Sage.

2 Interactive Programming using Python

Course Code	**CSI408**	**Course Title**	**Interactive Programming using Python**
Number of Credits	**3 Credits (TH), 2 Credits (PR)**	**Internal**	**20**
Total Contact Hours:	**3 HRS (TH/Week) 4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course Outcomes:

Student will be able to work with python. Students will be able to enrich their skills to deal with different python flavors for variety of applications.

Prerequisite:

Student should possess basic programming skills. Exposure with C++, Java was preferred.

At Course Completion:

Student will be able to work with python. Students will be able to enrich their skills to deal with different python flavours for variety of applications.

Course Outline

Unit-1:

Getting started with python: Python features, python environment, configuration and installation, python interpreter, interactive mode. Data types and Operations: Core data types, Numbers, Strings, Lists, Dictionaries, Tuples, files and others.

Unit-2:

Statement and Syntax: python statements, assignments, expression and prints, conditional statements if, multiway branching. Looping Controls: while, for, loop coding techniques, Iterations and Comprehension. Iterators, Lists Comprehension, Range iterators, the map, zip and filter iterators, multiple vs single iterator, generators, timing iterators. Functions: scope, arguments, types of functions, recursion, function objects, anonymous function, Modules

Unit-3:

Exception Handling: exceptions, default exception handler, catching exception, raising exception, user defined exceptions, termination action. Exception coding details: try/except/else statement, try statement, try else clause, try/finally statement, unified try/except/finally statement, raise statement, assert statement. Exception Objects: exception hierarchy, built-in exceptions, nesting exceptions, designing exceptions.

Unit-4:
Classes and OOP: class statement, constructors and expressions, methods, Inheritance, Multiple inheritance (Is-a, Has-a), static, decorators, metaclasses, Namespaces. Operator overloading: indexing and slicing, memberships, attribute reference. Delegation, Extending Built-in types, User Defined Modules.

Unit-5:
Wrappers in Python: Reflections, Isinstance, Duck typing, callable, Dir, Getattr, Regular expression: overview, matching and searching, replacing, splitting, escaping, flags, pattern objects.

Reference Books:

- Learning Python, 5th Edition, powerful Object-Oriented Programming, By Mark Lutz, and Publisher: O'Reilly Media, Final Release Date: June 2013

E-books:

1. Python Book(http://upload.wikimedia.org/wikipedia/commons/9/91/Python_Programming.pdf)
2. http://pythonbooks.revolunet.com/

Lab Exercise:
CSI455 Practical based on CSI408:There will be minimum 10 programs on the entire content of the course. Students have to undertake a mini-project based on the concept learned under this course.

3 Software Engineering and CASE Tools

Course Code	**CSI409**	**Course Title**	**Software Engineering and CASE Tools**
Number of Credits	**3 Credits (TH) 2 Credits (PR)**	**Internal**	**20**
Total Contact Hours	**3 HRS (TH/Week) 4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course Outcomes:

Student should be able to understand the basic and advance concepts of software Engineering. The students should be able to apply these to real time projects in software development.

Prerequisite:
Basic understanding of software development process.

At Course Completion:
Student should be able to understand the basic and advance concepts of software Engineering.The students should be able to apply these to real time projects in software development.

Course Outline
Unit-1:
The Nature of Software,. Defining Software, Legacy Software, Software Engineering, Software characteristics, Application software.Software myths, Software engineering-A layered technology, A process framework, waterfall model, incremental and evolutionary models. *Requirement engineering:* Requirement engineering task, initiating the requirement engineering process, eliciting requirements, developing use cases, building analysis model, Negotiating requirements, validating requirements, data modeling, functional modeling and behavioral modeling. Requirements Specification: Characteristics of an SRS, Components of an SRS, Specification Languages, Structure of a requirements document.

Unit-2:
Design within the Context of Software Engineering, The Design Process, Design Concepts, The Design Model. Architectural Design:Software Architecture, Architectural Genres, Architectural Styles Architectural Design, Assessing Alternative Architectural Designs. Component Level Design: Designing Class-Based Components Conducting Component-Level Design, Cohesion and Coupling.User Interface Design: The Golden Rules, Interface Analysis and Design, Interface Analysis Interface Design Steps.

Unit-3:
Verification and Validation, Testing Overview: Verification vs Validation, Design of test cases Black- Box testing: Equivalence Class Partitioning, Graph based testing Boundary Value analysis White- Box Testing: Statement Coverage, Branch Coverage, Condition Coverage, Path Coverage, Cycloramic Complexity Metric Data Plow- Based Testing . Integration Testing: Top down Testing, Bottom Up testing, Regression Testing, fazed vs Incremental Integration testing Systems Testing: Stress Testing Recovery Testing Security Testing. Debugging Techniques, Approaches, Tools

Unit-4:
Project Management Concepts, Process and project metrics, Process Metrics and Software Process Improvement, Project Metrics, Software Measurement, Size-Oriented Metrics, Function-Oriented Metrics, Object-Oriented Metrics, Use-Case–Oriented Metrics.
Project Planning Process: Software Scope and Feasibility, Resources, Software Project Estimation, Decomposition Techniques, Empirical Estimation Models, Scheduling, Risk analysis, monitoring and management. Software Configuration Management, Quality Management.

Unit-5:
Clean room Software Engineering, Web Engineering, Software Reengineering, Reverse Engineering, Forward Engineering, Computer Aided Software Engineering.

Books:

- Software Engineering – A Practitioners Approach Roger S. Pressman, 4th /7th Edition, Tata McGraw Hill, International Education.

- An Integrated Approach To S/w Engineering, Pankaj Jolote, 1st / 2nd Edition, Narosa.
- Software Engineering – A Programming Approach, D. Belie I. Moray, J. Rough, PHI.

Reference Books:

- James Peter, W Pedrycz, "Software Engineering", John Wiley & Sons
- K K Aggarwal & Yogesh Singh, "Software Engineering", New Age International, 2001

E-books:

- Software Engineering – A Practitioners Approach Roger S. Pressman, 5th Edition, Tata McGraw Hill, International Education.

Lab Exercise:
CSI456 Practical based on CSI409: At least two experiments should be carried out on each unit.

4 Data Structure

Course Code	**CSI410**	**Course Title**	**Data Structure**
Number of Credits	**3 Credits (TH)** **2 Credits (PR)**	**Internal**	**20**
Total Contact Hours	**3 HRS (TH/Week)** **4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course Outcomes:

- After completing this course, students will be able to: Understand structure and behavior of Algorithms, Better scope to write effective programs, the course content helpful in the preparation of UGC-SET/NET, DRDO entry level Examinations.
- This course will be the basic course for learning Algorithms and Approximation of Algorithms related research.

Prerequisites:
Before attending this course, students must have introductory concepts of like set theory, Principals of programming languages, Introductory Object oriented programming paradigm.

At Course Completion:
After completing this course, students will be able to: Understand structure and behavior of Algorithms, Better scope to write effective programs, the course content helpful in the preparation of UGC-SET/NET, DRDO entry level Examinations, This course will be the basic course for learning Algorithms and Approximation of Algorithms related research.

Course Outline:
Unit-1:

Introduction: Need of Data structures and Algorithms, Writing Algorithms, Testing Algorithms using Hand runs, Building Algorithms, Data Structures types like Linear, Non Linear, and Abstract Data Types, Data Structure operations.

Unit-2:
Design and Analysis Techniques: Dynamic Programming:- Elements of Dynamic Programming, Matrix Multiplication, Longest Common Subsequence Problem, Introduction to Greedy Algorithms, Amortized Analysis Methods, time and Space Complexity of Algorithm.

Unit-3:
Elementary Data Structures: Arrays- Ordered List, Sparse Metrics, Array Representations, Stacks and Queues: -Organization, Operations, multiple stacks, types of queues, Linked Lists:- Singly Linked List, Doubly Linked List, Doubly Linked List and Dynamic Storage management, Garbage collection and Compaction.

Unit-4:
Trees: Terminology, Binary Trees, Tree representation, Tree traversal, Threaded Binary Trees, Binary Search trees, B-Tree, AVL- Tree, Graph Algorithms: Terminology and Representation, Traversal: BFS, DFS, Connected Components and Spanning trees, Dijkstra's algorithm.

Unit-5:
Searching and Sorting: Linear Vs Binary search, Sorting: - Insertion Sort, Merge Sort, Quick sort, Radix Sort.

Reference Books:
1. Fundamentals of Data Structures by Ellis Horowitz and SartajSahani.
2. Introduction to Algorithms by Thomas H Core man et.al, PHI Publication.

E-Books:
http://www.tutorialspoint.com//java/java_data_structures.htm

Lab Exercise:
CSI457 Practical based on CSI410: At least two experiments should be carried out on each unit.

5 Computer Network

Course Code	**CSI411**	**Course Title**	**Computer Network**
Number of Credits	**3 Credits (TH)** **2 Credits (PR)**	**Internal**	**20**
Total Contact Hours	**3 HRS (TH/Week)** **4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course Outcomes:

- The student will be well acquainted with how computer network works, what are the architectures and protocols required for it., as well as some special topics.

Prerequisite:
Basic knowledge of Computer.

At Course Completion:
The student will be well acquainted with how computer network works, what are the architectures and protocols required for it., as well as some special topics.

Course Outline

Unit - 1:
Introduction to Computer Networks: Types of Network, Topologies, Client Server Model, Connection oriented, connection less services, Transmission Media-Guided, Unguided, Multiplexing, Bit rate, Baud Rate, Bandwidth etc. Network Hardware Components: Hub, Switch, Bridge, Router, Repeater, Gateway, peer to peer networks etc. Network architecture – layers: Reference Models ISO-OSI Reference model, TCP/IP reference model. Asynchronous Transfer Mode (ATM): Cells, Header and Cell Formats, Layers in ATM.

Unit - 2:
Data Link Layer: Introduction, DLL Protocols, Frame Relay, X.25 protocol, MAC sub layer protocols, IEEE Standards for LAN, error detection and correction at DLL.

Unit - 3:
Network layer : Routing Algorithms, Congestion Control, IP Addressing & DNS: What are IP address?, class of IP address, Subnet Masks, MAC address, IPv4 and IPv6 IP address, The Domain Name System: concept, zones of DNS, Switching: Switching concept, Circuit Switching, Packet Switching. Transport layer: Introduction, TCP and UDP services, Multicasting, Spanning Tree.

Unit - 4:
Network Security: Cryptography Techniques, Algorithms: Secret key and Public key, DES, RSA. Digital Signature, Firewalls, Proxy server.

Unit - 5:
Special Topics: current and emerging trends: Next Generation Networks, Data Centers, Cloud Computing, GSM, GIS, Queuing models, ASN(abstract syntax Notation), Introduction to wireless and sensor networks, distributed Networks.

Books:
1. Computer Networks – A. Tanenbaum, (PHI pub.)
2. Data and Computer Communication – Willam Stallings, PHI pub.

Reference Books:
1. Data Communication & Network – Forouzan (TMH)
2. Internetworking with TCP/IP Vol-I – Comer (PHI pub.)
3. Data Communications and distributed Networks-V.B, Black, (Prentice Hall pub.)

E-books:
http://newwayofengineering.blogspot.in/2014/12/computer-networks-tanenbaum-5th-edition.html

Lab Exercise:
CSI458 Practical based on CSI411: At least two experiments should be carried out on each unit.

6 Relational Database Management System

Course Code	**CSI412**	**Course Title**	**Relational Database Management System using SQL**
Number of Credits	**3 Credits (TH) 2 Credits (PR)**	**Internal**	**20**
Total Contact Hours	**3 HRS (TH/Week) 4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course Outcomes:

- To make the students aware of the basic knowledge of the relational database management which will include the aspects of database design, query languages and database system implementation.

Prerequisite:
It is just assumed to have a familiarity with basic data structures, computer organization and any high level programming language.

At Course Completion:
The student will be able to understand the basic concepts of database as well as will be familiar with its design ,development and security issues.

Course Outline
Unit-1:
Introduction: An overview to DBMS,Data models, levels of abstraction,dataindependence,instances and schema,structure of DBMS,database users, functions of database administrators. Entity relationship model: Entities, Attributes and Entity sets,Relation and relationship sets,features of E-R model.

Unit–2:
Relational model: Introduction, Integrity constraints over relations,Enforcing data integrity,relational data, logical database design,introduction to views,

Unit-3:
Relational algebra and relational calculus: operations on relational algebra,operations on relational calculus, tuple relational calculus,domain relational calculus

Unit-4:
Relational Database Design: Functional dependencies, schema refinement,Normal forms-first,second,third,BCNF,fourth and fifth normal forms,multivalued dependencies.Structured Query language(SQL): Basic SQL queries,nestedqueries,aggregateoperators,nullvalues,Embedded SQL. Query Processing: Concept, Need of query processing, procedure and its importance , query optimization,estimation of query processing cost,structure of query optimizer and Join strategies.

Unit-5:
Internals of RDBMS:Transaction management, ACID property, Schedules and its various types,serializability, deadlock handling,high performance transaction systems.Concurrency Control: Lock based protocols, Timestamp based protocols, Validation techniques, Multiple Granularity.Crash Recovery:-Failure classification,Log based recovery,Checkpoints,Shadow paging.Distributed Databases:-Structure and design of distributed databases, Design of distributed databases,Transparency and autonomy,distributed query processing, recovery in distributed databases,Commitprotocols.Security and Integrity: Authorization and views, security specification in SQL,Encryption.

Books:
Database System Concepts by Henry Korth and Abraham Silberschatz

Reference Books:
Relational database systems by Desai, Relational database concepts by Aho Ullman.

Lab Exercise:
CSI459 Practical based on CSI412: At least two practical's should be carried on each unit.

Semester - III

1. Advanced Programming using Python

Course Code	**CSI501**	**Course Title**	**Advance Programming using python**
Number of Credits	**3 Credits (TH)** **2 Credits (PR)**	**Internal**	**20**
Total Contact Hours	**3 HRS (TH/Week)** **4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course Outcomes:

- This course will provide an opportunity for student to use python threading, GUI building, use of databases, Web Framework, Web Services etc. FLASK framework, exposure to compatible framework will be provided to the student so that they may utilize python for their target web applications.

Prerequisite:
Student should possess basic programming skills using python.

At Course Completion:
Student will be able apply python web application developments.

Course Outline

Unit-1:
Threads: Introduction to Threads, thread organization, thread architectures, starting new thread, thread modules, Creating thread using Threading module, Synchronising threads. Controlling access to resources,

Unit-2:
GUI Design using TkInter or jython: Introduction, Layout Management, Widgets, Menus and Toolbars, Dialog boxes, Drawings, Nibbles. Database Programming using python (MySql): Python Database Interfaces and APIs, Database Connections, Creating Table, Insert Operation, Read operation, Update and Delete Operation, Performing Transactions, Commit & Rollback Operations, Handling Errors.

Unit-3:
Web Framework using Web.py: Introduction, URL Handling, GET and POST method, difference between GET and POST, Configuring server, starting server, Templating, Forms, Databashing, development

Unit-4:
Web socket programmingandWeb services: Introduction, About Sockets, socket module, types of sockets, Server Socket Methods, Client Socket Method, Design of simple server and client, python internet modules, HTTP web services: features of HTTP, using web services, debugging web services, setting user agents, handling redirects, handling compressed data.

Unit-5:

FLASK framework: Introduction, Exploring FLASK, Coding Conventions, Environment, configuration, organizing project, handling templates, files, handling forms, deployments.

Reference Books:

1. Python 3 Web Development Guide, Michel Anders, Beginners guide, PACKT Publishing, open source.

E-books: -

- Python Threading:
  - http://www.tutorialspoint.com/python/python_multithreading.htm
  - http://pymotw.com/2/threading/
  - http://www.python-course.eu/threads.php
- GUI:
  - https://wiki.python.org/moin/TkInter
  - https://wiki.python.org/jython/LearningJython
  - http://www.tutorialspoint.com/python/python_gui_programming.htm
- Database:
  - Python MySQL API https://wiki.python.org/moin/DatabaseInterfaces
  - http://www.tutorialspoint.com/python/python_database_access.htm
- Web Framework: http://webpy.org/docs/0.3/tutorial
- Python webSocket:
  - http://www.tutorialspoint.com/python/python_networking.htm
  - https://docs.python.org/2/howto/sockets.html
  - https://docs.python.org/3.0/library/socket.html
  - http://www.diveintopython.net/http_web_services/index.html
- FLASK framework,
  - http://www.fullstackpython.com/flask.html
  - https://exploreflask.com/

Lab Exercise:

CSI552 Practical based on CSI501:Students have to complete minimum 10 practical assignments based on the course contents.

2. Network Security (Elective-I)

Course Code	**CSI524**	**Course Title**	**Network Security**
Number of Credits	**3 Credits (TH)** **2 Credits (PR)**	**Internal**	**20**
Total Contact Hours	**3 HRS (TH/Week)** **4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course Outcomes:

- After completing this course, students will be able to:

- Explain concepts related to applied cryptography, including plaintext, cipher text, symmetric cryptography, asymmetric cryptography, and digital signatures.
- Explain the theory behind the security of different cryptographic algorithms.
- Explain common network vulnerabilities and attacks, defense mechanisms against network attacks, and cryptographic protection mechanisms.
- Outline the requirements and mechanisms for identification and authentication.
- Identify the possible threats to each mechanism and ways protect against these threats.
- Explain the requirements of real-time communication security and issues related to the security of web services.
- Explain the requirements of non-real-time security (email security) and ways to provide privacy, source authentication, message integrity, non-repudiation, p.

Prerequisite:
Before attending this course, students must have:
Programming experiences in C/C++ or JAVA.

At Course Completion:
After completing this course, students will be able to:

- Explain concepts related to applied cryptography, including plaintext, cipher text, symmetric cryptography, asymmetric cryptography, and digital signatures.
- Explain the theory behind the security of different cryptographic algorithms.
- Explain common network vulnerabilities and attacks, defense mechanisms againstnetwork attacks, and cryptographic protection mechanisms.
- Outline the requirements and mechanisms for identification and authentication.
- Identify the possible threats to each mechanism and ways to protect against these threats.
- Explain the requirements of real-time communication security and issues related to the security of web services.
- Explain the requirements of non-real-time security (email security) and ways toprovide privacy, source authentication, message integrity, non-repudiation, p

Course Outline
Unit - 1:
Introduction, Security Concepts, Threats and Risks, Attacks – Passive and Active, Security Services, Confidentiality, Authentication, Non-Repudiation, Integrity, Access Control, Availability. Security attacks, Unauthorized Access, Impersonation, Denial of Service Malicious Software, Viruses, Worms, Trojan, spyware

Unit - 2:
Access Control Models, Bell-LaPadula, Biba Integrity Model, Role Base Model. Cryptography: Secret Key and Public Key Cryptosystems: Cryptanalysis and attacks Symmetric Ciphers, Block Ciphers and Stream Ciphers: DES, Triple DES, RC4 and RC5, Cryptographic Modes, RSA., Deffie Hellman key exchange Message Authentication: MD5 and SHA 512 Public Key Infrastructure (PKI): Digital Certificates , Certificate Authorities

Unit - 3:
Network Attacks: Buffer Overflow, IP Spoofing, TCP Session Hijacking, Sequence Guessing, Network Scanning,ICMP, TCP sweeps, Basic Port Scans Network Security: Objectives and Architectures, Internet Security Protocols, IP encapsulating ,Security Protocol Virtual Private Network: concepts, PPTP, L2TP

Unit - 4:
Web security Consideration: Secured Socket Layer and Transport layer security, Secured Electronic Transaction (SET) and Secured Mail: Pretty Good Privacy (PGP), S/MIME Network Security Authentication Mechanisms: a) Passwords, b) Cryptographic authentication protocol, c) Smart Card, d) Biometrics) Digital Signatures and seals, f) Kerberos, g) X.509 LDAP Directory

Unit - 5:
Intruders, Intrusion Detection and Prevention, Firewall: Firewall Design Principles, Firewall Characteristics, Types of Firewalls: Packet Filtering Router, Stateful Inspection Firewall, Application Level Gateway or Proxy, Circuit level gateway, Bastion Host Firewall Configuration: Screened Host Firewall System, Screened Subnet Firewall System. Cybercrimes: Crimes against the computer, Crimes using a computer, Indian IT Act 2000: Objectives, Provisions, And Offenses.

Reference Books:

1. Network Security Essentials, William Stallings, Prentice-Hall.
2. Fundamentals of Computer Security Technology, Edward Amoroso, Prentice-Hall.
3. Cryptography and Data Security, Dorothy E. Denning, Addison-Wesley.
4. Computers under Attack, Peter J. Denning, Addison-Wesley.
5. Cryptography: Theory and Practice, Douglas R. Stinson, CRC Press.
6. Computer Crime and Computer Forensics, Dr. R.K.Tiwari,P.K.Sastri,K.V.Ravikumar, First Edition, 2002,
7. Select Publishers
8. Computer Security Gollmann, Dieter, First Edition, 1999, John Wiley & Sons Ltd.

Lab Exercise:
CSI560 Practical based on CSI524: Lab exercise will cover the program related to each unit.

Semester - IV

1. Data Mining (Elective –II)

Course Code	**CSI503**	**Course Title**	**Data Mining**
Number of Credits	**3 Credits (TH) 2 Credits (PR)**	**Internal**	**20**
Total Contact Hours	**3 HRS (TH/Week) 4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course Outcomes:

- Understand the basic concepts and principles in data mining and visualization.
- Learn commonly used algorithms for mining both structured and unstructured (text) data.
- Understand how to handle a large amount of text data with search engines.

Prerequisite:
Student should have knowledge about the Relational Database Management System and Basic knowledge of probability and statistics.

At Course Completion:

- Understand the basic concepts and principles in data mining and visualization.
- Learn commonly used algorithms for mining both structured and unstructured (text) data.
- Understand how to handle a large amount of text data with search engines.

Course Outline:

Unit-1:
Introduction to Data Mining: Why Mine Data? Commercial Viewpoint, Scientific Viewpoint Motivation, Definitions, Origins of Data Mining, Data Mining Tasks, Classification, Clustering, Association Rule Discovery, Sequential Pattern Discovery, Regression, Challenges of Data Mining.
Data Mining: DataWhat is Data? Attribute Values, Measurement of Length, Types and Properties of Attributes, Discrete and Continuous Attributes, Types of data sets, Data Quality, Data Preprocessing, Aggregation, Sampling, Dimensionality Reduction, Feature subset selection, Feature creation, Discretization and Binarization, Attribute Transformation, Density.Data Mining: Exploring Data:Data Exploration Techniques, Summary Statistics, Frequency and Mode, Percentiles, Measures of Location: Mean and Median, Measures of Spread: Range and Variance, Visualization, Representation, Arrangement, Selection, Visualization Techniques: Histograms, Box Plots, Scatter Plots, Contour Plots, Matrix Plots, Parallel Coordinates, Other Visualization Techniques, OLAP : OLAP Operations

Unit-2:
Data Mining Classification: Basic Concepts, Decision Trees, and Model EvaluationClassification: Definition, Classification Techniques, Tree Induction, Measures of Node Impurity, Practical Issues of Classification, ROC curve, Confidence Interval for Accuracy, Comparing Performance of Two Models, Comparing Performance of Two Algorithms.Data Mining Classification: Alternative

TechniquesRule-Based Classifier, Rule Ordering Schemes, Building Classification Rules, Instance-Based Classifiers, Nearest Neighbor Classifiers, Bayes Classifier, Naive Bayes Classifier, Artificial Neural Networks (ANN), Support Vector Machines.

Unit-3:

Data Mining Association Analysis: Basic Concepts and AlgorithmsAssociation Rule Mining, Frequent Itemset Generation, Association Rule Discovery : Hash tree, Factors Affecting Complexity, Maximal Frequent Horible Closed Item set, Alternative Methods for Frequent Item set Generation, FP-growth Algorithm, Tree Projection, Rule Generation, Pattern Evaluation, Statistical Independence, Properties of A Good Measure, Support-based Pruning, Subjective Interestingness Measure.

Unit-4:

Data Mining Cluster Analysis: Basic Concepts and AlgorithmsApplications of Cluster Analysis, Types of Clusters, Clustering Algorithms:K-means and its variants, Hierarchical clustering, Density-based clustering. Graph-Based Clustering, Limitations of Current Merging Schemes, Characteristics of Spatial Data Sets, Shared Near Neighbor Approach, ROCK (Robust Clustering using links), Jarvis-Patrick Clustering, SNN Clustering Algorithm.Data Mining Anomaly DetectionAnomaly/Outlier Detection, Importance, Anomaly Detection Schemes, Density-based: LOF approach.

Unit-5:

WEKA (Waikato Environment for Knowledge Analysis):is a well-known suite of machine learning software that supports several typical data mining tasks, particularly data preprocessing, clustering, classification, regression, visualization, and feature selection.Orange: is a component-based data mining and machine learning software suite that features friendly yet powerful, fast and versatile visual programming front-end for explorative data analysis and visualization, and Python bindings and libraries for scripting. It contains complete set of components for data preprocessing, feature scoring and filtering, modeling, model evaluation, and exploration techniques.RapidMiner:Formerly called YALE (Yet another Learning Environment), is an environment for machine learning and data mining experiments that is utilized for both research and real-world data mining tasks. HepWork:Designed for scientists, engineers and students, intelligible and comprehensive open-source data integration, processing, analysis, and exploration platform. It gives users the ability to visually create data flows or pipelines, selectively execute some or all analysis steps, and later study the results, models, and interactive views.

REFERENCES:

1. Introduction to Data Mining by Tan, Steinbach, Kumar.
2. Data Mining: Concepts and Techniques by Jiawei Han, Micheline Kamber, Morgan Kaufmann Publishers.
3. Data Mining: Practical Machine Learning Tools and Techniques by Ian H. Witten and Eibe Frank, Morgan Kaufmann, 2nd Edition (2005).
4. Principles of Data Mining: David Hand, HeikkiMannila&Padhraic Smyth, PHP Publication.
5.

Lab Exercise:

CSI556, Practical based on CSI503:There should be minimum 10 lab assignment on the topics discussed in the course.

Semester - IV (Elective-II)

VB.NET using MySQL

Course Code	**CSI530**	**Course Title**	**VB.NET using MySQL**
Number of Credits	**3 Credits (TH)** **2 Credits (PR)**	**Internal**	**20**
Total Contact Hours	**3 HRS (TH/Week)** **4 HRS (PR/Week)**	**External (Semester/Term Exam)**	**80**

Course Outcomes:

- After completing this course, students will be able to; Create a simple Visual Basic.NET–based application based on the Windows Application template. Create a simple Visual Basic.NET–based Web Forms application that uses an XML Web Service. Access and manipulate data in a MYSQL database by using ADO.NET, and Build, package, and deploy an application.

Prerequisite:
This course is intended for both novice and experienced programmers who have a minimum of three months programming experience and have basic Microsoft Windows navigation skills.
At Course Completion:
After completing this course, students will be able to; Create a simple Visual Basic.NET–based application based on the Windows Application template.Create a simple Visual Basic.NET–based Web Forms application that uses an XML Web Service. Access and manipulate data in a MYSQL database by using ADO.NET, and Build, package, and deploy an application.

Course Outline:

Unit-1:
Getting Started: Microsoft Visual Studio .NET Architecture, Application in Visual Basic .NET, Basic .NET Concepts, Exploring the Development Environment, Creating a Visual Basic .NET Projects.
Using Variables and Arrays: Introduction to Data Types, Using Variables, Variable Scope, Converting Data Types, Creating and Using Structures, Storing Data in Arrays

Unit-2:
Working with Procedures: Creating Procedures, Using Procedures, Using Predefined Functions
Decision Structures and Loops: Using Conditional Expressions, Using Decision Structures, Using Conditional Loop Structures. Validating User Input: Restricting User Input, Validating Field Data, Validating Form Data

Unit-3:
Object-oriented Programming in Visual Basic .NET: Understanding Classes, Working with Classes, Using Shared Members, Inheritance, Polymorphism, and Namespaces Handling Errors and Exceptions: Types of Errors, Using the Debugger and Handling Exceptions.

Unit-4:
Working with Forms and Controls: Understanding Programming Concepts, Working with Windows FormsWorking with Controls, Styling Your Code.Enhancing the User Interface: Creating Menus, Creating Status Bars, Creating Toolbars.

Unit-5:
Using ADO.NET: Database Concepts, Overview of ADO.NET, Overview of MySQL, Working with Database using MySQL, Create Insert, delete Table commit etc. Deploying Applications: Introduction to Deployment, Deploying a Windows-based Application

Reference Books:

1. Steven holzner, Visual basic .net programming black book, Wiley publishing.
2. Heinrich Gantenbein, Microsoft Visual Basic .NET 2003 Unleashed

E-Books:
http://www.tutorialspoint.com/mysql/mysql_tutorial.pdf

Lab Exercise:
CSI566 Practical based on CSI530: At least two experiments should be carried out on each unit.