MTech.- IT (Information Security Management)

First Semester

S. No.	Code	Subject	L-T-P	Credits	Category
1.	MIS-101	Advanced Programming	3-0-2	4	DCC
2.	MIS-103	Secure Coding and Security Engineering	3-0-2	4	DCC
3.	MIS-105	Fundamentals of Information Security	3-0-2	4	DCC
4.	MCS-107	Data Structures and Algorithm Analysis	3-0-2	4	DCC
5	GEC-101	Generic Open Elective	2-0-0 1-1-0 0-0-4 0-2-0	2	GEC
6.	ROC-101	Research Methodology	3-0-0	3	ROC
		Total Credits		21	

Second Semester

S. No.	Code	Subject	L-T-P	Credits	Category
1.	MIS-102	Advances in Machine Learning	3-0-2	4	DCC
2.	MIS-104	Applied Cryptography	3-1-0	4	DCC
3.	MIS-106	Cyber Security and Forensics	3-0-2	4	DCC
4.	DEC-1xx	Departmental Elective Course – 1	3-0-2 3-1-0 2-1-2	4	DEC
5.	DEC-1xx	Departmental Elective Course – 2	3-0-2 3-1-0 2-1-2	4	DEC
6	ROC-102	Research Ethics	3-0-0	3	ROC
		Total credits		23	

Third Semester

S. No.	Code	Subject	L-T-P	Credits	Category
1.	MIS-201	Ethical Hacking	3-0-2	4	DCC
2.	DEC-2xx	Departmental Elective-3	3-0-2 3-1-0 2-1-2	4	DEC
3.	DEC-2xx	Departmental Elective-4	3-0-2 3-1-0 2-1-2	4	DEC
4	GEC-201	General Open Elective	2-0-0 1-1-0 0-0-4	2	GEC
5	MIS-251	Dissertation – I/Project Work	-	8	DCC
6	MIS-253	Industrial Training/Internship	-	1	DCC
		Total credits		23	

Fourth Semester

S. No.	Code	Subject	L-T-P	Credits	Category
1.	MIS-252	Dissertation – II/Project Work	-	20	DCC
		Total credits		20	

List of Departmental Elective Courses

Category	Course Code	Subject	Credits
Departmental Elective Course-1	MIS-108	Adv. Database Management Systems	3-0-2
	MIS-110	Introduction to Biometrics	3-0-2
	MIS-112	Computer Vision	3-0-2
	MIS-114	Blockchain Fundamentals	3-0-2
Departmental Elective Course-2	MCS-106	Probability and Random Process	3-0-2
	MIS-118	Semantic Web	3-1-0
	MIS-120	Security Testing and Risk Management	3-0-2
	MIS-122	Natural Language Processing and Information Retrieval	3-0-2
Departmental Elective Course-3	MIS-203	Neural Network and Deep Learning	3-0-2
	MIS-205	Security Patterns	3-0-2
	MIS-207	Cryptographic Protocols and Algorithms	3-0-2
	MIS-209	Advanced Network Technology	3-0-2
Departmental Elective Course-4	MIS-211	Cyber Laws and Rights	3-1-0
	MIS-213	Security and Privacy in Social Networks	3-1-0
	MIS-215	Software Defined Networks	3-1-0
	MIS-217	Cloud Computing Architecture and Security	3-0-2

Advanced programming			
Course Code	: MIS-101	Credits	: 4
Contact Hours	: L-3 T-0 P-2	Semester	: 1
Course Category	: DCC		

Introduction: This course is designed to enable students to recognize the need for distributed, transactional and portable applications that leverage speed, security and reliability of server side technologies. This course shall inculcate programming capability to handle business logic and develop and deploy applications using Java Platform, Enterprise Edition.

Course Objective

- To learn Software Development and Deployment that is reliable, scalable and portable.
- To learn object oriented programming through Design Patterns.
- To learn secure integrating web applications developed on varied platform through web services technologies.

Pre-requisite:

- Basic Knowledge of Object Oriented programming, Java Programming Language and Database Management

Course Outcome: Upon successful completion of this course, students will be able to:

- Acquire Software development skills that are reliable, scalable and portable applications.
- Design and implement software development with Clean Code through use of Design patterns and MVC Framework.
- Implement web services application for transacting web applications built on varied platforms
- Implement security features available in enterprise edition of Java EE8.

Pedagogy

Lectures will be imparted along with hands on lab sessions and development of a Web Application for case study (ies).

Contents

UNIT-I	10 Hours
Introduction to Java, Features of Java 8 and EE 8 , Variables, Arrays, Strings, Exception Handling, Multithreading, Collection Framework, Creating Interfaces, packages, JAR Files, Annotations , JDBC. Networking and Security Programming, socket Programming, Session Handling, Remote Method Invocation,	
UNIT-II	11 Hours
Working with Servlets, Event Handling and Wrappers, Working with Java Beans, bean validation, Java Server Pages, Expression Languages, JSP Application Development, Tags Extensions and Implementation JSP Tag Library, Implementing Filters	
UNIT-III	10 Hours
Working with java Server Faces, Understanding Java Mail, Java EE Design Patterns, Working with Hibernate, Struts, Spring MVC, Spring configuration. Case Study such as (Any One): Building an Online Book Store, Online Shopping cart, University Management System , simple e-commerce application -Forest case Study	
UNIT-IV	11 Hours

Implement SOA using Java Web Services, JSON Processing, Building Web services with JAX, Building SOAP , UDDI, RESTFul services, Working with Glassfish, JBOSS server, JUnit Testing Security in JAVA EE.	
Text Books	
1	Jim Koegh, "Java EE Complete Reference", Mc Graw Hill , First Edition, 2017
2	"Core and Advanced Java, Black Book", DreamTech Publications, First Edition, 2018
3	Java Platform, Enterprise Edition 8: The Java EE Tutorial, Oracle, Java Documentation, 2018
Reference Books	
1	David R. Heffelfinger, "Java EE 8 Application Development", Packt Publishing, First Edition, December 2017

Secure Coding and Security Engineering	
Course Code: MIS-103 Contact Hours: L-3 T-0 P-2 Course Category: DCC	Credits : 4 Semester: 1

Introduction: Security breaches in software are costing companies large fines and regulatory burdens. Developing software, that is reliable in its functionality, resilient against attackers, and recoverable when the expected business operations are disrupted, is a must have. The assurance of confidentiality, integrity and availability is becoming an integral part of software development. This course is being introduced to integrate security principles and secure programming with Software development to reduce effort in removing basic vulnerabilities and risk thereby. The course is effective in enabling students to learn and develop software that is reliable and resilient to software attacks.

Course Objective

- To learn Secure Software Development Guidelines and Best Practices.
- To learn secure programming practices so as to build secure software resilient to cyber attacks.
- To learn secure configuration of various tiers and layers involved in Software Development.

Pre-requisite:

- Basic Knowledge of Programming Language (s), Database Management, Network, Server

Course Outcome: Upon successful completion of this course, students will be able to:

- Acquire security requirements with respect to software development.
- Design and implement software development with minimum software vulnerabilities.
- Write and test software code with respect to security testing and remove security flaws.

Pedagogy

Lectures will be imparted along with hands on lab sessions and latest real world case studies about software vulnerabilities reported, prevention and patching techniques.

Contents

UNIT-I	10 Hours
Secure software development life-cycle: Software development life cycle (Microsoft, McAfee, OWASP etc), development team, Quality and Security, Application Guidelines, (ISC)^2 Ten best practices of secure software development, Security principles, Security Standards Three pillars of software security, Seven Touch points of software security, Security Methodologies, Security Framework, Security Models	
UNIT-II	**10 Hours**
Secure Software Requirements: Introduction, Objective, Sources, Types of Security Requirements, Requirements Engineering for Secure Software, Concepts of Misuse and Abuse, SQUARE Process Model, SQUARE Sample Outputs, Requirements Elicitation and Prioritization, Object Modeling, Threat Modeling Secure Software Design: Design Consideration, processes, Architecture, technologies,	
UNIT-III	**12 Hours**
Secure Software Implementation, : Introduction to Software Vulnerability and Preventive/ Defensive techniques , Vulnerability description, types, Vulnerability Databases, OWASP top 10, NVD, CWE, Common Software Vulnerabilities and Controls, Defensive Coding Practices—Concepts and Techniques : Buffer Overrun, Format String Problems, Integer Overflow, and Injection flaws : SQL Injection, Command	

Injection, Failure to Handle Errors, Cross Site Scripting, Broken Authentication and Session Management, Magic URLs, Weak Passwords, Failing to Protect Data, Weak random numbers, improper use of cryptography, Insecure Direct Object References, Insecure De-serialization, Security Mis-configuration, Information Leakage, Race Conditions, Poor Usability, Not Updating Easily, Executing with too much privilege, Failing to protect network traffic, improper use of PKI, trusting network name resolution

UNIT-IV	**10 Hours**

Secure Coding Standards: Memory Management, Exception management, Development tools, IDEs tools, Versioning tools, Networking tools, Coding in the cube: Functions, procedures and code blocks, Structuring for Validation, Structured Programming, Debugging, Coding and applying security requirements during maintenance,

Security code analysis and review: Code review with a tool (fortify, coverty etc), Code analysis

Securing Server, Database, Network and their secure configuration, Firewalls,

Case Study : Recent Software vulnerabilities due to insecure programming and how to prevent them during design and implementation

Text Books	
1	Paul, M. (2016). Official (ISC) 2 Guide to the CSSLP. CRC Press.
2	SEACORD, R. (2013). Secure Coding in C and C++ (2nd Edition). SEI Series in Software Engineering
3	Howard, Michael, David LeBlanc, and John Viega. "24 Deadly Sins of Software Security." Programming Flaws and How to Fix Them (2010). McGraw-Hill Education

Reference Books	
1	Ransome, J., & Misra, A. (2018). Core software security: Security at the source. CRC press.
2	Bishop, M. (2019). Computer Security(2nd Edition). Addison-Wesley Professional.
3	McGraw, G. (2006). Software security: building security in (Vol. 1). Addison-Wesley Professional
4	John Veiga, Gary Mc Graw, "Building Secure Software: How to Avoid Security Problems the Right Way", Addison-Wesley Professional Computing Series, 2001

Fundamentals of Information Security	
Course Code: MIS-105 Contact Hours: L-3 T-0 P-2 Course Category: DCC	Credits: 4 Semester: 1

Introduction:

This course will introduce students to fundamentals of information security, cryptography, access control mechanisms, system attacks and defences against them

 Course Objective:

- Analyze the vulnerabilities in any computing system and hence be able to design a security solution
- Identify the basic security issues in the computer network communications
- Evaluate various security mechanisms used in real world

Pre-requisite: None

Course Outcome:

- Learn information security basics
- Learn to use and apply various security mechanisms to real world problems

Pedagogy: Lecture, Power point Slides, Online Material, Assignments

Contents

UNIT-I	10 Hours
Security Overview, CIA model, Threats, Policy and Mechanisms, Security Policies, Confidentiality Policies, Integrity Policies, Hybrid Policies, Cryptography Basics, Classical Cryptosystems, Stream Ciphers and Block Ciphers, Public Key Cryptography: RSA	
UNIT-II	11 Hours
Cryptographic Checksums , Authentication Basics, Password management, Challenge Response, Biometrics, Key Exchange, Certificate Chains, X.509, Digital Signatures, Access Control Lists: Creation and Maintenance, Revocation of Rights, Ring based Access Control	
UNIT-III	11 Hours
Malicious Logic, Trojan Horses, Viruses, Worms, Logic Bombs, Defenses, Sandboxing, Intrusion Detection: Principles and Basics, Anomaly modelling, Architecture: Host and network based Information Gathering, Organization of Intrusion Detection Systems, Intrusion Response	
UNIT-IV	10 Hours
Firewalls and Proxies, DMZ server, User Security: Policy, Access, Files and Devices, Processes, Electronic Communications, Program Security: Requirements and Policy, Design, common security related programming problems, Virtual Machines Structure	
Text Books	
1	Matt Bishop, S.S. Venkatramanayya, "Introduction to Computer Security,

	Addison-Wesley Professional, 3rd Ed., 2005
2	W Stallings, "Cryptography and Network Security: Principles and Practice, Prentice Hall, 6th Ed. 2014
Reference Books	
1	B. Forouzan, D. Mukhopadhyay, "Cryptography and Network Security", McGraw Hill, 3rd Ed., 2011

RESEARCH METHODOLOGY	
Course Code: ROC-101 Contact Hours: L-2 T-0 P-2 Course Category: ROC	Credits: 3 Semester: 1

Introduction: The course introduced the students to the research world. They will understand how to read and interpret a research papers. Also this course will make then understand how to write their own research proposals.

Course Objectives:

➢ The purpose is to align the M. Tech. students with the research process of the rest of the world so that when they go for higher studies either in university or industry , they should be aware of how to communicate their research to the rest of the world and how to read and interpret the research work of others.

➢ The students will also become aware of metrics of how to evaluate the quality of different research works.

➢ The practical statistical tools will enhance their skills to apply mathematical tools to their research to interpret results and communicate in a common jargon.

Course Outcomes: Having successfully completed this course

➢ The students will be able to write their own research proposal.

➢ They will also have familiarity with research journals, how do the whole process works right from draft of a research paper to reviewer comments and publishing.

Pedagogy: Classroom teaching which focuses upon relating the textbook concepts with real world phenomena, along with regular tutorial classes to enhance the problem-solving ability.

Contents

UNIT-I	10 Hours
Objectives and Motivation of Research, Types of Research, Research Approaches, Significance of Research, Research Methods verses Methodology, Research and Scientific Method, Important of Research Methodology, Research Process, Criteria of Good Research, Problems Encountered by Researchers in India, Benefits to the society in general. Defining the Research Problem: Definition of Research Problem, Problem Formulation, Necessity of Defining the Problem, Technique involved in Defining a Problem.	
UNIT-II	**10 Hours**
Literature Survey: Importance of Literature Survey, Sources of Information, Assessment of Quality of Journals and Articles, Information through Internet, Significance of ISSN, DOI, Impact Factor, Citations. Rules of Citation.	
Research Design: Meaning of Research Design, Need of Research Design, Feature of a Good Design, Important Concepts Related to Research Design, Different Research Designs, Basic	

Principles of Experimental Design, Developing a Research Plan, Design of Experimental Set-up, Use of Standards.

UNIT-III	11 Hours

Descriptive Statistics, Points of Central tendency, Measures of Variability, Measures of relationship, Inferential Statistics-Estimation, Hypothesis Testing.

How to read data from multiple files, Interpretation of Data: Univariate Analysis, Tests for significance: Chi square, t-test, Regression modeling, Direct and Interaction effects, ANOVA, F-test, Time Series analysis, Autocorrelation and Autoregressive modeling.

Inferential statistics: Normal Curve, Confidence Interval, Type1 and Type 2 errors.

UNIT-IV	11 Hours

Research Report Writing: Format of the Research report, Style of writing report, References/Bibliography/Webliography, Technical paper writing/Journal report writing.

Survey Methods: Questionnaire method; Types of Questionnaires; Process of Questionnaire Designing; Advantages and Disadvantages of Questionnaire Method.

Text Books

1.	C.R Kothari, "Research Methodology, Methods & Technique"; New Age International Publishers, 2004
2.	R. Ganesan, "Research Methodology for Engineers", MJP Publishers, 2011
3.	Y.P. Agarwal, "Statistical Methods: Concepts, Application and Computation", Sterling Publs., Pvt., Ltd., New Delhi, 2004
4.	Vijay Upagade and Aravind Shende, "Research Methodology", S. Chand & Company Ltd., New Delhi, 2009

DATA STRUCTURES AND ALGORITHM ANALYSIS	
Course Code: MCS- 107 Contact Hours: L-3 T-0 P-2 Course Category: DCC	Credits: 4 Semester: 1

Introduction: This course is about teaching of various data structure designs & its implementations, analyzing the various algorithm strategies and designing of new algorithms for various classes of problems. It is intended to be a gentle introduction to how we specify data structure, algorithms, some of the design strategies, and many of the fundamental ideas used in algorithm analysis throughout the syllabus.

Course Objective:

- To build an understanding on the basics of core and advance data structure.
- To introduce the various strategies used in the algorithm design and their analysis.
- TO teach the selection of data structure for a particular problem
- To teach students, how to write complex program using dynamic data strcutres

Pre-requisite: Students should have some programming experience. In particular, they should understand recursive procedures and simple data structures such as arrays and linked lists. Students should have some facility with proofs by mathematical induction.

Course Outcome: After studying this course, Students will be able to :

- Successfully design and implements the core and advance data structures
- Successfully analyses the complexity associated with the various data structures
- Analyse, design and implements the various proposed algorithm based on different algorithmic strategies.
- Choose data structures for various complex problems

Pedagogy : Classroom teaching which focuses on developing understanding of students to digest the concepts of subject with large number of examples.

Contents

UNIT-I	10 Hours
Algorithms performance analysis: Time and space complexity, Asymptotic Notations, Complexity Analysis Examples. **Linear Data Structures:** Arrays, Stacks, Queues, Linked lists, **Recursion:** Solving recurrences.	

UNIT-II	10 Hours
Non-linear Data Structure: Trees, Traversals, Binary Search Trees, AVL tree, B-trees, B+ Tree, Red Black Tree. **Graph Algorithms:** DFS, BFS, Minimum Spanning Tree Algorithms, Shortest path Algorithms.	

UNIT-III	12 Hours

Sorting and Searching Algorithms: Quick Sort, Merge Sort, Heap sort; Linear Search and Binary Search. **Hashing:** Hashing Functions, Collision Resolution Techniques

UNIT-IV	10 Hours

Algorithm Strategies: Greedy paradigm with examples. Divide and conquer paradigm with examples. Dynamic-programming paradigm with examples. **NP Completeness:** P, NP, NP-complete, NP-Hard categories of problems, Cook's theorem.

Text Books

1	Y. Langsam et. al., "Data Structures using C and C++", Second Edition, PHI, 2015.
2	E. Horowitz, S. Sahani, Anderson-Freed "Fundamentals of Data Structures in C", Second Edition, University Press, 2008
3	T. H. Cormen, C. E. Leiserson, R. L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rd Ed., PHI, 2011.

Reference Books

1	R. L. Kruse, B. P. Leung, C. L. Tondo, "Data Structures and program design in C", PHI, 2010.
2	Ellis Horowitz and Sartaz Sahani, "Fundamental of Computer Algorithms", Galgotia Publications, 2009.
3	A. V. Aho, J. E. Hopcroft, J. D. Ullman, "The Design and Analysis of Computer Algorithms", Addition Wesley, 2009.
4	D. E. Knuth, "The Art of Computer Programming", 2nd Ed., Addison Wesley, 2011.

Suggestive List of experiments   (More programs can be added to this list)

1. Write a program that Implement Single Link List with following operations: i) Insertion of a node at first node, at any position and at end of list. ii) Deletion of a node at start, at middle and at end of list. iii) Display the link list. iv) Count the number of nodes in the link list. v) Search a node in the link list. vi) Sort the link list. vii) Reverse the link list.
2. Write a program that Implement Stack with all primitive operations by using Array. Implement Queue with all primitive operations by using Array.
3. Write a program that Implement Stack and Queue with all primitive operations by using link list.
4. Write a program that Implement doubly link list with primitive operations: (i) Create a doubly linked list (ii Insert a new node to the left of the node. (iii) Delete the node of a given data. (iv) Display the contents of the list.

5. Write a program that Implement Circular link list with primitive operations. (i) Creation of the Circular list (ii) Insertion of the node (iii) Deletion an element (iv) Display the list
6. Write a program that Implement Binary Search Technique.
7. Write a program that Implement AVL tree and Red Black Tree.
8. Write a program that Implement Binary Tree and its Traversal.
9. Write a program that Implement BFS & DFS over a graph.
10. Write a program that Implement shortest path algorithms.
11. Write a program that Implement Quick Sort, Merge Sort and Heap Sort.
12. Write a program that implement Disjoint Set Data Structure
13. Write a program that implements Knapsack Problem.
14. Write a program that implements Huffman Coding
15. Write a program to implement Prim's and Kruskal's Algorithm
16. Write a program to implement Matrix Chain Multiplication Algorithm
17. Write a program to implement Longest Common Subsequence Algorithm.
18. Write a program to implement All Pair Shortest Path Algorithm.