TEMPLATE FOR COURSE SYLLABUS FOR NEP IMPLEMENTATION

Discipline: Science □ Arts, Humanities & Social Science □

Commerce □ BBA □ BCA [√]

Subject Name: BCA

Subject Code: UBCAMAJ36013 (Will be provided by the University)

Semester: Semester I □ Semester II □ Semester III [] Semester IV □

Semester V□ Semester VI[√] Semester VII □ Semester VIII

Course Name: ARTIFICIAL INTELLIGENCE

Course Code: (Will be provided by the University)

Course Credit:	Theoretical	3	Practical/Tutorial	1
Marks Allotted:	Theoretical	40	Practical/Tutorial	20
	Continuing Evaluation	10	Attendance	5

Course Type (tick the correct alternatives):

Major Core	[√]	AEC	□
Interdisciplinary/ DSE	□	SEC	□
Minor / Generic Elective	□	VAC	□
Research Project/Dissertation	□	Vocational	□

Is the course focused on employability / entrepreneurship? YES[√] NO □

Is the course focused on imparting life skill? YES[√ NO □

Is the course based on Activity? YES[√ NO □

Remarks by Chairman, UG BOS, if any

UG BOS Meeting Reference Number: Date:

Prepared by CIRM

Course Code: UBCAMAJ36013
Course Name: ARTIFICIAL INTELLIGENCE

Brief Course Description:

The Artificial Intelligence (AI) course offers a comprehensive introduction to the principles and practices of AI, encompassing various techniques and applications that enable machines to mimic human intelligence. This course covers foundational topics such as machine learning, natural language processing, robotics, and neural networks. It is designed to equip students with the theoretical knowledge and practical skills needed to develop intelligent systems capable of solving complex problems.

Prerequisite(s) and/or Note(s):

Students should have a solid foundation in mathematics, particularly in linear algebra, calculus, and probability. Prior programming experience, especially in languages like Python, is highly recommended. Knowledge of basic algorithms and data structures will be beneficial. This course is suitable for students pursuing studies in computer science, engineering, data science, and related fields.

Course Objectives:

The primary objectives of the Artificial Intelligence course are to provide students with a thorough understanding of the fundamental concepts and techniques used in AI. The course aims to teach students how to apply AI methods to real-world problems, design and implement AI algorithms, and evaluate their effectiveness. By the end of the course, students will be well-versed in the ethical considerations and future trends in AI, preparing them to contribute to advancements in this rapidly evolving field.

Knowledge acquired:

Students will acquire a deep understanding of various AI methodologies, including supervised and unsupervised learning, reinforcement learning, and deep learning. They will learn about the architecture and functioning of neural networks, the principles of natural language processing, and the basics of robotics and computer vision. The course also covers AI applications in areas such as healthcare, finance, and autonomous systems, providing a broad perspective on the impact of AI technologies.

Skills gained:

Throughout the course, students will develop practical skills in designing, implementing, and evaluating AI algorithms. They will gain hands-on experience with machine learning libraries and frameworks, such as TensorFlow, Keras, and scikit-learn. Students will learn to preprocess and analyze data, train and optimize machine learning models, and interpret the results. Additionally, they will develop skills in problem-solving, critical thinking, and programming, essential for building intelligent systems.

Competency Developed:

Upon completing the course, students will have developed the competency to design and implement AI-driven solutions to complex problems. They will be capable of applying AI techniques to various domains, from developing intelligent assistants to creating predictive models. This competency will enable students to pursue careers in AI research, data science, software development, and related fields. The course provides a strong foundation for advanced studies and specialization in AI, preparing students to innovate and lead in the development of intelligent technologies.

Detailed Syllabus

3rd Year: Semester 6

UBCAMAJ36013: ARTIFICIAL INTELLIGENCE [Credits:3, Lecture :45]

Unit 1: Introduction (5 Lectures)

Introduction to Artificial Intelligence, Background and Applications, Turing Test and Rational Agent approaches to AI, Introduction to Intelligent Agents, their structure, behavior and environment.

Unit 2: Problem Solving and Searching Techniques (15 Lectures)

Problem Characteristics, Production Systems, Control Strategies, Breadth First Search, Depth First Search, Hill climbing and its Variations, Heuristics Search Techniques: Best First Search, A* algorithm, AO* algorithm, Constraint Satisfaction Problem, Means-End Analysis, Introduction to Game Playing, Min-Max and Alpha-Beta pruning algorithms.

Unit 3: Knowledge Representation (15 Lectures)

Introduction to First Order Predicate Logic, Resolution Principle, Unification, Semantic Nets, Conceptual Dependencies, Frames, and Scripts, Production Rules, Conceptual Graphs. Programming in Logic (PROLOG)

Unit 4: Dealing with Uncertainty and Inconsistencies (5 Lectures)

Truth Maintenance System, Default Reasoning, Probabilistic Reasoning, Bayesian Probabilistic Inference, Possible World Representations.

Unit 5: Understanding Natural Languages (5 Lectures)

Parsing Techniques, Context-Free and Transformational Grammars, Recursive and Augmented Transition Nets.

Suggested Readings:

1. DAN.W. Patterson, Introduction to A.I and Expert Systems – PHI, 2007.
2. Russell &Norvig, Artificial Intelligence-A Modern Approach**,** LPE, Pearson Prentice Hall, 2nd edition, 2005.
3. Rich & Knight, Artificial Intelligence – Tata McGraw Hill, 2nd edition, 1991.
4. W.F. Clocksin and Mellish, Programming in PROLOG, Narosa Publishing House, 3rd edition, 2001.
5. Ivan Bratko, Prolog Programming for Artificial Intelligence, Addison-Wesley, Pearson Education, 3rd edition, 2000.

UBCAMAJ36013: ARTIFICIAL INTELLIGENCE [Credits:1, Lecture Hours:30]

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. Write a prolog program to calculate the sum of two numbers.
2. Write a prolog program to find the maximum of two numbers.
3. Write a prolog program to calculate the factorial of a given number.
4. Write a prolog program to calculate the nth Fibonacci number.
5. Write a prolog program, insert_nth(item, n, into_list, result) that asserts that result is the list into_list with item inserted as the n„th element into every list at all levels.
6. Write a Prolog program to remove the Nth item from a list.
7. Write a Prolog program, remove-nth(Before, After) that asserts the After list is the Before list with the removal of every n„th item from every list at all levels.
8. Write a Prolog program to implement append for two lists.
9. Write a Prolog program to implement palindrome(List).
10. Write a Prolog program to implement max(X,Y,Max) so that Max is the greater of two numbers X and Y.
11. Write a Prolog program to implement maxlist(List,Max) so that Max is the greatest number in the list of numbers List.
12. Write a Prolog program to implement sumlist(List,Sum) so that Sum is the sum of a given list of numbers List.
13. Write a Prolog program to implement two predicates evenlength(List) and oddlength(List) so that they are true if their argument is a list of even or odd length respectively.
14. Write a Prolog program to implement reverse(List,ReversedList) that reverses lists.
15. Write a Prolog program to implement maxlist(List,Max) so that Max is the greatest number in the list of numbers List using cut predicate.
16. Write a Prolog program to implement GCD of two numbers.
17. Write a prolog program that implements Semantic Networks/Frame Structures.

TEMPLATE FOR COURSE SYLLABUS FOR NEP IMPLEMENTATION

Discipline: Science □ Arts, Humanities & Social Science □

Commerce □ BBA □ BCA [√]

Subject Name: BCA

Subject Code: UBCAMAJ36014 (Will be provided by the University)

Semester: Semester I □ Semester II □ Semester III □ Semester IV □

Semester V □ Semester VI[√] Semester VII □ Semester VIII □

Course Name: OPTIMIZATION TECHNIQUE

Course Code: (Will be provided by the University)

Course Credit: Theoretical 3 Practical/Tutorial 1

Marks Allotted: Theoretical 60 Practical/Tutorial 0

Continuing Evaluation 10 Attendance 5

Course Type (tick the correct alternatives):

Major Core	[√]	AEC	□
Interdisciplinary/ DSE	□	SEC	□
Minor / Generic Elective	□	VAC	□
Research Project/Dissertation	□	Vocational	□

Is the course focused on employability / entrepreneurship? YES [√] NO □

Is the course focused on imparting life skill? YES [√] NO □

Is the course based on Activity? YES [√] NO □

Remarks by Chairman, UG BOS, if any

UG BOS Meeting Reference Number: Date:

Prepared by CIRM

Course Code: UBCAMAJ36014

Course Name: OPTIMIZATION TECHNIQUE

Brief Course Description:

The Optimization Technique course provides an in-depth exploration of methods and algorithms used to find optimal solutions in various mathematical and real-world problems. This course covers linear, nonlinear, integer, and dynamic programming, as well as advanced optimization techniques such as metaheuristics and stochastic optimization. Students will learn to formulate optimization problems, select appropriate methods, and apply these techniques to diverse fields including engineering, economics, and operations research.

Prerequisite(s) and/or Note(s):

Students should have a strong foundation in calculus, linear algebra, and basic probability and statistics. Familiarity with mathematical modeling and previous experience with programming languages like Python or MATLAB is recommended. This course is particularly suitable for students in mathematics, engineering, computer science, and related fields.

Course Objectives:

The primary objectives of the Optimization Technique course are to equip students with the theoretical knowledge and practical skills needed to solve optimization problems. The course aims to teach students how to formulate optimization models, analyze different types of optimization problems, and apply appropriate algorithms to find optimal solutions. By the end of the course, students will be adept at using optimization techniques to improve decision-making processes in various applications.

Knowledge acquired:

Students will gain a comprehensive understanding of fundamental optimization concepts, including linear programming, simplex method, duality theory, and sensitivity analysis. They will also learn about nonlinear optimization, integer programming, and dynamic programming. The course covers advanced topics such as genetic algorithms, simulated annealing, and particle swarm optimization. Additionally, students will acquire knowledge about the practical implementation of these techniques using software tools.

Skills gained:

Throughout the course, students will develop practical skills in modeling and solving optimization problems. They will learn to use optimization software and programming languages to implement and solve various optimization models. Students will gain experience in analyzing and interpreting the results of optimization algorithms, ensuring their solutions are both efficient and effective. They will also develop problem-solving skills, critical thinking, and the ability to apply optimization techniques to real-world scenarios.

Competency Developed:

Upon completing the course, students will have developed the competency to design and implement optimization models for complex problems in various domains. They will be capable of selecting and applying appropriate optimization techniques to achieve optimal solutions, enhancing decision-making processes in fields such as logistics, finance, engineering, and data science. This competency will enable students to pursue careers in operations research, optimization consulting, and advanced analytics. The course provides a strong foundation for further study and specialization in optimization and related areas, preparing students to tackle complex challenges with innovative optimization strategies.

Detailed Syllabus

3^{Rd} Year: Semester 6

UBCAMAJ36014: OPTIMIZATION TECHNIQUE [Credits: 3, Lectures:45]

Unit 1: Introduction (5 Lectures)

Origin & Development of Operational Research, Definition and Meaning of Operational Research, Different Phases of an Operational Research, Mathematical Modeling of Real-Life Problems, Optimization problems.

Unit 2: Transportation Model (8 Lectures)

Transportation problem and its variants: Northwest corner method, VAM method, Assignment problem: Hungarian method.

Unit 3: Linear Programming (15 Lectures)

Introduction to Linear algebra. Solution of a system of Linear Equations, Linear independence and dependence of vectors, Concept of Basis, Basic Feasible solution, Problem Formulation, Solution by Graphical Method, Simplex Algorithm, Artificial Starting Solution, Big-M and Two-phase method, Special cases in Simplex Method: Degeneracy, Alternative optima, Unbounded Solution, Infeasible Solution.

Unit 4: Duality (5 Lectures)

Definition of Dual Problem, Primal-Dual relationship, Duality problem, Dual Simplex.

Unit 5: Integer Programming (12 Lectures)

Integer programming: Branch and bound algorithm.

<u>Suggested Readings</u>:

1. 1. G. Hadley: Linear Programming. Narosa, 2002 (reprint).
2. A. Ravindran, D. T. Phillips and James J. Solberg: Operations Research-Principles and Practice, JohnWiley & Sons, 2005.
3. Hamdy A. Taha: Operations Research-An Introduction, Prentice Hall, 8th Edition, 2008. 4. F.S. Hillier.G.J.
4. Lieberman: Introduction to Operations Research- Concepts and Cases, 9th Edition, Tata McGraw Hill.2010.

BCAMAJ36014 OPTIMIZATION TECHNIQUE TUTORIAL [Credit:1, Tutorial Hours:30]

Optimization techniques tutorial as advised by teacher.

TEMPLATE FOR COURSE SYLLABUS FOR NEP IMPLEMENTATION

Discipline: Science □ Arts, Humanities & Social Science □

Commerce □ BBA □ BCA [√]

Subject Name: BCA

Subject Code: UBCAMAJ36015 (Will be provided by the University)

Semester: Semester I □ Semester II □ Semester III □ Semester IV □

Semester V□ Semester VI [√] Semester VII □ Semester VIII □

Course Name: MOBILE COMPUTING

Course Code: (Will be provided by the

University) Course Credit: 3 Theoretical 1

Practical/Tutorial

Marks Allotted: Theoretical 40 Practical/Tutorial 20

Continuing Evaluation 10 Attendance 5

Course Type (tick the correct alternatives):

Major Core [√] AEC □

Interdisciplinary/ DSE □ SEC □

Minor / Generic Elective □ VAC □

Research Project/Dissertation □ Vocational □

Is the course focused on employability / entrepreneurship? YES [√] NO □

Is the course focused on imparting life skill? YES [√] NO □

Is the course based on Activity? YES [√] NO □

Remarks by Chairman, UG BOS, if any

UG BOS Meeting Reference Number: Date:

Prepared by CIRM

Course Code: UBCAMAJ36015
Course Name: MOBILE COMPUTING

Brief Course Description:

The Mobile Computing course offers a comprehensive introduction to the principles, technologies, and applications of mobile computing. This course covers the fundamental concepts of mobile networks, mobile operating systems, and application development for mobile devices. Students will explore the challenges and solutions related to designing and implementing mobile applications, focusing on both hardware and software aspects. The course prepares students to develop innovative mobile applications and understand the underlying technologies that enable mobile communication.

Prerequisite(s) and/or Note(s):

Students should have a basic understanding of computer science principles, including programming, data structures, and algorithms. Familiarity with operating systems and networks is also beneficial. Prior experience with programming languages such as Java, Swift, or Kotlin will be advantageous but is not mandatory. This course is ideal for students pursuing degrees in computer science, information technology, or related fields.

Course Objectives:

The primary objectives of the Mobile Computing course are to provide students with a solid foundation in mobile computing technologies and application development. The course aims to teach students how to design, develop, and deploy mobile applications, understand mobile network architectures, and address security and privacy issues in mobile computing. By the end of the course, students will be proficient in creating mobile solutions that are efficient, user-friendly, and secure.

Knowledge acquired:
Students will acquire detailed knowledge of mobile computing concepts, including mobile network architectures, protocols, and standards. They will learn about mobile operating systems, such as Android and iOS, and gain insight into the development environments and tools used for mobile application development. The course covers topics like mobile application lifecycle, user interface design, data storage, and synchronization. Additionally, students will understand the challenges of mobile computing, including power management, connectivity, and security.

Skills gained:

Throughout the course, students will develop practical skills in designing and developing mobile applications. They will learn to use development environments such as Android Studio and Xcode, and programming languages like Java, Kotlin, and Swift. Students will gain experience in creating responsive and intuitive user interfaces, integrating APIs and services, and ensuring application performance and security. They will also develop skills in testing and debugging mobile applications, ensuring they function seamlessly across different devices and operating systems.

Competency Developed:

Upon completing the course, students will have developed the competency to design, develop, and deploy professional-quality mobile applications. They will be capable of addressing the unique challenges of mobile computing, including optimizing applications for performance and ensuring data security. This competency will enable students to pursue careers in mobile application development, mobile computing research, and related fields. The course provides a strong foundation for advanced study and specialization in mobile technologies, preparing students to innovate and lead in the dynamic and rapidly evolving field of mobile computing**.**

Detailed Syllabus

3rd Year: Semester 6

UBCAMAJ36015: MOBILE COMPUTING **[Credits: 3, Lectures:45]**

Unit – 1: Introduction to Mobile Communications and Computing (5 Lectures)
Mobile Computing (MC): Introduction to MC, Novel Applications, Limitations, and Architecture. GSM: Mobile Services, System Architecture, Radio Interface, Protocols, Localization and Calling, Handover, Security, and New Data Services.

Unit – 2: (Wireless) Medium Access Control (4 Lectures)
Motivation for a Specialized MAC (Hidden and Exposed Terminals, Near and Far Terminals), SDMA, FDMA, TDMA, CDMA.

Unit – 3: Mobile Network Layer (6 Lectures)
Mobile IP (Goals, Assumptions, Entities and Terminology, Ip Packet Delivery, Agent Advertisement and Discovery, Registration, Tunneling And Encapsulation, Optimizations), Dynamic Host Configuration Protocol (DHCP).

Unit – 4: Mobile Transport Layer (6 Lectures)
Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast Retransmit/Fast Recovery, Transmission /Time-Out Freezing, Selective Retransmission, Transaction Oriented TCP.

Unit – 5: Database Issues (6 Lectures)
Hoarding Techniques, Caching Invalidation Mechanisms, Client Server Computing With Adaptation, Power-Aware And Context-Aware Computing, Transactional Models, Query Processing, Recovery, And Quality Of Service Issues.

Unit – 6: Data Dissemination (5 Lectures)
Communications Asymmetry, Classification of New Data Delivery Mechanisms, Pushbased Mechanisms, Pull-Based Mechanisms, Hybrid Mechanisms, Selective Tuning (Indexing) Techniques.

Unit – 7: Mobile Ad Hoc Networks (MANETS) (5 Lectures)
Overview, Properties of A MANET, Spectrum Of MANET Applications, Routing And Various Routing Algorithms, Security In MANET.

Unit – 8: Protocols and Tools (8 Lectures)
Wireless Application Protocol-WAP. (Introduction, Protocol Architecture, and Treatment Of Protocols Of All Layers), Bluetooth (User Scenarios, Physical Layer, Mac Layer, Networking, Security, Link Management) And J2me

Suggested Readings:

1. Jochen Schiller, “Mobile Communications”, Addison-Wesley. second edition, 2004.
2. Stojmenovic and Cacute, “Handbook of Wireless Networks and Mobile Computing”, Wiley, 2002, ISBN 0471419028.
3. Reza Behravanfar, “Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML”, ISBN: 0521817331, Cambridge University Press, October 2004,
4. Adelstein, Frank, Gupta, Sandeep KS, Richard III, Golden , Schwiebert, Loren, “Fundamentals of Mobile and Pervasive Computing”, ISBN: 0071412379, McGraw-Hill Professional, 2005.
5. Hansmann, Merk, Nicklous, Stober, “Principles of Mobile Computing”, Springer, second edition, 2003.
6. Martyn Mallick, “Mobile and Wireless Design Essentials”, Wiley DreamTech, 2003.

UBCAMAJ36015 MOBILE COMPUTING [Credit:1, Lab Hours: 30]

Students are advised to do laboratory/practical practice not limited to, but including the following types ofproblems:

1. Write a WML program to create a card.

2. Write a WML program to create a deck that contain two cards and provide the Functionality of calling two cards from one another.

3. Write a WML program to implement the functionality of Login by username.

4. Write a WML program to display special characters on the screen.

5. Write a WML program to display the image on the screen after 5 seconds.

6. Write a WML program to develop the calculator.

7. Write a WML program to develop a salary calculator.

8. Write a WML program to develop a loan calculator.

9. WAP to display an image in WML.

10. Write a WML program to display the text on the screen after 5 seconds.

11. Write a WML program to create a card.

12. Write a WML program to display username and password.

13. Write a WML program to display special characters on the screen.

14. Write a Program in WML to create a table and insert data into the table.

15. Write a WML script to use the different formatting tags

16. Write a WML program to display a string message

17. Write a WML program to create a link between two cards.

18. Write a program to create a table having rows and columns in WML

TEMPLATE FOR COURSE SYLLABUS FOR NEP IMPLEMENTATION

Discipline: Science □ Arts, Humanities & Social Science □

Commerce □ BBA □ BCA [√]

Subject Name: BCA

Subject Code: UBCAMAJ36016 (Will be provided by the University)

Semester: Semester I □ Semester II □ Semester III □ Semester IV □

Semester V□ Semester VI [√] Semester VII □ Semester VIII □

Course Name: **INFORMATION SECURITY**

Course Code: UBCAMAJ36016 (Will be provided by the University)

Course Credit: Theoretical 3 Practical/Tutorial 1

Marks Allotted: Theoretical 60 Practical/Tutorial 0

Continuing Evaluation 10 Attendance 5

Course Type (tick the correct alternatives):

Major Core [√] AEC □

Interdisciplinary/ DSE □ SEC □

Minor / Generic Elective □ VAC □

Research Project/Dissertation □ Vocational □

Is the course focused on employability / entrepreneurship? YES [√] NO □

Is the course focused on imparting life skill? YES [√] NO □

Is the course based on Activity? YES [√] NO □

Remarks by Chairman, UG BOS, if any

UG BOS Meeting Reference Number: Date:

TEMPLATE FOR COURSE SYLLABUS FOR NEP IMPLEMENTATION

Course Code: UBCAMAJ36016
Course Name: INFORMATION SECURITY

Brief Course Description

The Information Security course provides a comprehensive overview of the principles and practices essential for protecting information systems from cyber threats. This course covers topics such as cryptography, network security, risk management, and ethical hacking. Students will learn about the various types of security vulnerabilities and the strategies to mitigate them. The course prepares students to design, implement, and manage secure information systems, ensuring the confidentiality, integrity, and availability of data.

Prerequisite(s) and/or Note(s):

Students should have a basic understanding of computer networks, operating systems, and programming. Familiarity with concepts in computer science, particularly related to data structures and algorithms, is beneficial. This course is suitable for students pursuing degrees in computer science, information technology, or related fields, as well as professionals seeking to enhance their knowledge in information security.

Course Objectives:

The primary objectives of the Information Security course are to provide students with a deep understanding of the key concepts and techniques used in securing information systems. The course aims to teach students how to identify security threats and vulnerabilities, implement appropriate security measures, and develop policies for risk management. By the end of the course, students will be equipped to protect information assets and respond effectively to security incidents.

Knowledge Acquired:

Students will gain comprehensive knowledge of information security principles, including the CIA triad (confidentiality, integrity, availability) and security models. They will learn about cryptographic techniques, such as symmetric and asymmetric encryption, hashing, and digital signatures. The course covers network security concepts, including firewalls, intrusion detection systems, and VPNs. Students will also learn about security policies, risk assessment, and compliance standards, as well as the ethical and legal aspects of information security.

Skills Gained:

Throughout the course, students will develop practical skills in implementing security measures and protecting information systems. They will learn to conduct vulnerability assessments and penetration testing to identify and mitigate security risks. Students will gain experience in configuring and managing security tools, such as firewalls, antivirus software, and encryption solutions. Additionally, they will develop skills in incident response, including detecting, analyzing, and responding to security breaches.

Competency Developed:

Upon completing the course, students will have developed the competency to design and implement robust security strategies for information systems. They will be capable of assessing security risks, deploying appropriate countermeasures, and ensuring compliance with security standards and regulations. This competency will enable students to pursue careers in cybersecurity, information assurance, and IT management. The course provides a strong foundation for advanced study and specialization in information security, preparing students to tackle the evolving challenges of cybersecurity in various industries.

Detailed Syllabus

3rd Year: Semester 6

UBCAMAJ36016: INFORMATION SECURITY [Credits: 3, Lectures:45]

Unit 1: Introduction (5 Lectures)
Security, Attacks, Computer Criminals, Security Services, Security Mechanisms, User Authentication

Unit 2: Cryptography (10 Lectures)
Substitution ciphers, Transpositions Cipher, Confusion, diffusion, Symmetric, Asymmetric Encryption. DES Modes of DES, Uses of Encryption, Hash function, key exchange, Digital Signatures, Digital Certificates.

Unit 3: Program Security (5 Lectures)
Secure programs, Non malicious Program errors, Malicious codes virus, Trap doors, Salami attacks.

Unit 4: Database Security (5 Lectures)
Requirements, Reliability, Integrity, Sensitive data, Inference, Multilevel Security.

Unit 5: Security in Networks (10 Lectures)
Threats in Networks, Security Controls, firewalls, Intrusion detection systems, Secure e-mails

Unit 6: IT Act 2000 and Cyber Crimes (10 Lectures)
Definitions, Digital signature, Electronic governance, Attribution, acknowledgement and dispatch of electronic records, Regulation of certifying authorities, Duties of subscribers, Penalties and adjudication, Appellate Tribunal, Offences and Cyber-crimes

Suggested Readings

1. C. P. Pfleeger, S. L. Pfleeger; Security in Computing, Prentice Hall of India, 2006
2. W. Stallings; Network Security Essentials: Applications and Standards, 4/E, 2010
3. PT Joseph, *E-Commerce: An Indian Perspective*, PHI Learning
4. David Whiteley, *E-commerce: Strategy, Technology and Applications*, McGraw Hill Education

UBCAMAJ36016 INFORMATION SECURITY [Credit:1, Lecture Hours: 15]

Information Security tutorial as assigned and advised by teacher(s).