DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY,

UTTAR PRADESH, LUCKNOW

EVALUATION SCHEME & SYLLABUS
Second Year

FOR

MASTER OF COMPUTER APPLICATIONS (MCA)

(Two Year Course)

Based On

NEP 2020

[Effective from the Session: 2025-26]

MASTER OF COMPUTER APPLICATION (MCA)

MCA SECOND YEAR, 2025-26

SEMESTER-III

S. No.	Subject Code	Subject Name	Periods			Sessional			ESE	Total	Credit
			L	T	P	CT	TA	Total			
1.	BMC301	Python Programming	3	0	0	20	10	30	70	100	3
2.	BMC302	Software Engineering	4	0	0	20	10	30	70	100	4
3.	BMC303	Computer Network	3	1	0	20	10	30	70	100	4
4.		Elective – 1	3	0	0	20	10	30	70	100	3
5.		Elective – 2	3	1	0	20	10	30	70	100	3
6.	BMC351	Python Programming Lab	0	0	3	30	20	50	50	100	2
7.	BMC352	Software Engineering Lab	0	0	3	30	20	50	50	100	2
8.	BMC353	Mini Project**	0	0	4	30	20	50	50	100	2
		Total								800	23

CT: Class Test **TA**: Teacher Assessment **L/T/P**: Lecture/ Tutorial/ Practical

SEMESTER-IV

S. No.	Subject Code	Subject Name	Periods			Sessional			ESE	Total	Credit
			L	T	P	CT	TA	Total			
1.		Elective – 3	3	0	0	20	10	30	70	100	3
2.		Elective – 4	3	0	0	20	10	30	70	100	3
3.		Elective – 5	3	0	0	20	10	30	70	100	3
4.	BMC451	Startup and Entrepreneurial Activity Assessment ##	0	0	4	-	100	100	-	100	2
5.	BMC452	Project	-	-	-	-	200	200	400	600	12
		Total								1000	23

CT: Class Test **TA**: Teacher Assessment **L/T/P**: Lecture/ Tutorial/ Practical

Note: *****The Mini Project or internship (5-6 weeks) conducted during summer break after II Semester will be assessed during III Semester.***

##The Startup and Entrepreneurial Activity Assessment will be done in the IV semester, under which a student will have to undergo a startup/entrepreneurship activity of at least 60 hours till the III semester.

SECOND YEAR SYLLABUS
SEMESTER-III

BMC301: Python Programming		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course, the student will be able to		
CO 1	Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.	K_1, K_2
CO 2	Express proficiency in the handling of strings and functions	K_1, K_2
CO 3	Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples and sets.	K_3
CO 4	Use OO concepts while programming in Python	K_1, K_2
CO 5	Work with Python using GUI.	K_4
DETAILED SYLLABUS		**3-0-0**
Unit	**Topic**	**Proposed Lecture**
I	**Introduction to Python:** Introduction and Basics; Setting up path Python Data Variables & Operators: Data Variables and its types, id() and type() functions, Coding Standards, **Input-Output:** Printing on screen, Reading data from keyboard; **Control Structures:** if-else, elif, Nested if, Iteration Control structures, Break, Continue & Pass.	**08**
II	**String Manipulation:** Accessing Strings, Basic Operations, String slices, Function and Methods. **Lists:** Introduction, Accessing list, Operations, Working with lists, Function and Methods. **Tuple:** Introduction, accessing tuples, Operations, Working, Functions and Methods.	**08**
III	**Dictionaries:** Introduction, accessing values in dictionaries, Working with dictionaries, Properties, Functions. **Functions:** Defining & Calling a function, Passing arguments to functions – Mutable & Immutable Data Types, Different types of arguments, Recursion, Scope of variables;	**08**
IV	**Modules and Packages:** User-defined modules and Standard Library: random, numpy, scipy, sys, Math Module, String Module, List Module, Date & Time Module, Regular Expressions: match, search, replace; **File Handling:** Introduction, File Types, Creating, Opening, Closing, Renaming, Accessing and deleting files, File pointers, File Modes, Binary files.	**08**
V	**Exception Handling:** Exception, Exception Handling, Except clause, Try? finally clause, User Defined Exceptions. Basics of Python for Data Analysis, Introduction to series and dataframes.	**08**

Suggested Readings:

1. Basin H., "Python for Beginners", New Age International Publishers.
2. Ramalho L., "Fluent Python", SPD.
3. Severance C., "Python for Everybody", SPD.
4. Brown M. C., "The Complete Reference", Mc Graw Hill.
5. Kanetkar Y. and Kanetkar A., "Let Us Python", Bpb.
6. Lutz M., "Learning Python", SPD.

BMC302: Software Engineering		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course, the student will be able to understand		
CO 1	Explain various software characteristics and analyze different software Development Models.	K_1, K_2
CO 2	Demonstrate the contents of a SRS and apply basic software quality assurance practices to ensure that design, development meet or exceed applicable standards.	K_1, K_2
CO 3	Compare and contrast various methods for software design.	K_2, K_3
CO 4	Formulate testing strategy for software systems, employ techniques such as unit testing, Test driven development and functional testing.	K_3
CO 5	Manage software development process independently as well as in teams and make use of various software management tools for development, maintenance and analysis.	K_5
DETAILED SYLLABUS		**3-1-0**
Unit	**Topic**	**Proposed Lecture**
I	**Introduction:** Introduction to Software Engineering, Software Components, Software Characteristics, Software Crisis, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes, Software Quality Attributes. Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models.	**08**
II	**Software Requirement Specifications (SRS):** Requirement Engineering Process: Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modelling, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, IEEE Standards for SRS. Software Quality Assurance (SQA): Verification and Validation, SQA Plans, Software Quality Frameworks, ISO 9000 Models, SEI-CMM Model.	**08**
III	**Software Design:** Basic Concept of Software Design, Architectural Design, Low Level Design: Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion Measures, Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design. Software Measurement and Metrics: Various Size Oriented Measures: Halestead's Software Science, Function Point (FP) Based Measures, Cyclomatic Complexity Measures: Control Flow Graphs.	**08**

IV	**Software Testing:** Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, Top Down and Bottom- Up Testing Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing), Functional Testing (Black Box Testing), Test Data Suit Preparation, Alpha and Beta Testing of Products. Static Testing Strategies: Formal Technical Reviews (Peer Reviews), Walk Through, Code Inspection, Compliance with Design and Coding Standards.	**08**
V	**Software Maintenance and Software Project Management:** Software as an Evolutionary Entity, Need for Maintenance, Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re-Engineering, Reverse Engineering. Software Configuration Management Activities, Change Control Process, Software Version Control, An Overview of CASE Tools. Estimation of Various Parameters such as Cost, Efforts, Schedule/Duration, Constructive Cost Models (COCOMO), Resource Allocation Models, Software Risk Analysis and Management.	**08**

Suggested Readings:

1. R S Pressman, "Software Engineering: A Practitioners Approach", McGraw Hill.
2. Pankaj Jalote, "Software Engineering", Wiley
3. Rajib Mall, "Fundamentals of Software Engineering", PHI Publication.
4. K K Aggarwal and Yogesh Singh, "Software Engineering", New Age International Publishers.
5. Ghezzi, M. Jarayeri, D. Manodrioli, "Fundamentals of Software Engineering", PHI Publication.
6. Ian Sommerville, "Software Engineering", Addison Wesley.
7. Kassem Saleh, "Software Engineering", Cengage Learning
8. Pfleeger, "Software Engineering", Macmillan Publication

BMC303: Computer Networks		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course, the student will be able to understand		
CO 1	Describe communication models TCP/IP, ISO-OSI model, network topologies along with communicating devices and connecting media.	K2
CO 2	Apply knowledge of error detection, correction and learn concepts of flow control along with error control.	K3
CO 3	Classify various IP addressing techniques, subnetting along with network routing protocols and algorithms.	K4
CO 4	Understand various transport layer protocols and their design considerations along with congestion control to maintain Quality of Service.	K2
CO 5	Understand applications-layer protocols and elementary standards of cryptography and network security.	K2
DETAILED SYLLABUS		**3-0-0**
Unit	**Topic**	**Proposed Lecture**
I	**Data Communications**: Introduction: Data communication Components and characteristics, Data representation and Data flow. **Networks:** LAN, WAN, MAN, Topologies. **Protocols and Standards**: ISO-OSI model and TCP-IP Model. **Network Connecting Devices:** HUB, Bridge, Switch, Router and Gateways. **Transmission Media:** Guided and unguided Media **Classification and Arrangement:** Wired LANs and Wireless LANs	**08**
II	**Data Link Layer:** **Error Detection and Error Correction:** Types of errors, LRC, VRC, Checksum, CRC, and Hamming Code. **Flow Control and Error Control:** Stop and Wait Protocol, Sliding Window, Go-back-N-ARQ Protocol and Selective-Repeat ARQ Protocol. **Channel Allocation Protocols:** Random Access, Controlled and Channelization techniques such as ALOHA, CSMA, CSMA/CD, CDMA/CA, TDMA, FDMA, Token Passing, etc.	**08**
III	**Network Layer:** **Switching Techniques:** Circuit Switching, Packet Switching, and Message Switching. **Logical addressing:** IPv4 and IPv6 Address schemes, Classes and subnetting **Network Layer Protocols:** ARP, RARP, BOOTP and DHCP **Routing Techniques:** Interdomain and Intradomain routing with examples.	**08**
IV	**Transport Layer:** **Introduction to Transport Layer**: Process-to-Process Delivery:	**08**

	Reliable and unreliable Connection, Port and Socket Addressing **Transport Layer Protocols with packet formats:** User Datagram Protocol (UDP), Transmission Control Protocol (TCP), Stream Control Transmission Protocol (SCTP). **Congestion Control:** Techniques for handling the Congestion Control. **Quality of Service (QoS):** Flow Characteristics and techniques to improve QoS.	
V	**Application Layer:** **Basic Concept of Application Layer:** Domain Name System, World Wide Web, Hyper Text Transfer Protocol, Electronic mail, File Transfer Protocol, Remote login. **Introduction to Cryptography:** Definition, Goal, Applications, Attacks, Encryption, decryption, public-key and private key cryptography.	**08**

Suggested Readings:

1. Behrouz Forouzan, "Data Communication and Networking", McGraw Hill
2. Andrew Tanenbaum "Computer Networks", Prentice Hall.
3. William Stallings, "Data and Computer Communication", Pearson.
4. Kurose and Ross, "Computer Networking- A Top-Down Approach", Pearson.
5. Peterson and Davie, "Computer Networks: A Systems Approach", Morgan Kaufmann
6. W. A. Shay, "Understanding Communications and Networks", Cengage Learning.
7. D. Comer, "Computer Networks and Internets", Pearson.
8. Behrouz Forouzan, "TCP/IP Protocol Suite", McGraw Hill.

ELECTIVE-1

MASTER OF COMPUTER APPLICATION (MCA)

BMC011: Cryptography & Network Security		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course, the student will be able to understand		
CO 1	Understand various security attacks and their protection mechanism.	K_2
CO 2	Apply and analyze various encryption algorithms.	K_3, K_4
CO 3	Understand functions and algorithms to authenticate messages and study and apply different digital signature techniques.	K_1, K_2, K_3
CO 4	Analyze different types of key distributions.	K_4
CO 5	Study and appraise different IP and system security mechanism.	K_1, K_5
DETAILED SYLLABUS		**3-0-0**
Unit	**Topic**	**Proposed Lecture**
I	Introduction to security attacks, Services and mechanism, Classical encryption techniques substitution ciphers and transposition ciphers, Cryptanalysis, Steganography, Stream and block ciphers. **Modern Block Ciphers:** Block ciphers principles, Shannon's theory of confusion and diffusion, Feistel structure, Data encryption standard(DES), Strength of DES, Idea of differential cryptanalysis, Block cipher modes of operations, Triple DES	**08**
II	Introduction to group, field, finite field of the form GF(p), Modular arithmetic, Prime and relative prime numbers, Extended Euclidean Algorithm, Advanced Encryption Standard (AES). Fermat's and Euler's theorem, Primality testing, Chinese Remainder theorem, Discrete Logarithmic Problem, Principals of public key crypto systems, RSA algorithm, Security of RSA	**08**
III	**Message Authentication Codes:** Authentication requirements, Authentication functions, Message authentication code, Hash functions, Birthday attacks, Security of hash functions, Secure hash algorithm (SHA). **Digital Signatures:** Digital Signatures, Elgamal Digital Signature Techniques, Digital signature standards (DSS), Proof of digital signature algorithm.	**08**
IV	**Key Management and distribution:** Symmetric key distribution, Diffie-Hellman Key Exchange, Public key distribution, X.509 Certificates, Public key Infrastructure. **Authentication Applications:** Kerberos Electronic mail security: pretty good privacy (PGP), S/MIME.	**08**
V	**IP Security:** Architecture, Authentication header, Encapsulating security payloads, Combining security associations, Key management. Introduction to Secure Socket Layer, Secure electronic transaction (SET). **System Security:** Introductory idea of Intrusion, Intrusion detection, Viruses and related threats, firewalls.	**08**

Suggested Readings:

1. Stallings W., "Cryptography and Network Security: Principals and Practice", Pearson Education.
2. Frouzan B. A., "Cryptography and Network Security", McGraw Hill.
3. Kahate A., "Cryptography and Network Security", Tata McGraw Hill.

BMC012: Data Warehousing & Data Mining		
Course Outcome (CO)	**Bloom's Knowledge Level (KL)**	
At the end of course, the student will be able to understand		
CO1	Demonstrate knowledge of Data Warehouse and its components.	K_1, K_2
CO2	Discuss the process of Warehouse Planning and Implementation.	K_1, K_2
CO3	Discuss and implement various supervised and Non supervised learning algorithms on data.	K_6
CO4	Explain the various process of Data Mining and decide best according to type of data.	K_2, K_5
CO5	Explain process of knowledge discovery in database (KDD). Design Data Mining model.	K_2, K_5
	DETAILED SYLLABUS	**4-0-0**
Unit	**Topic**	**Proposed Lecture**
I	**Data Warehousing**: Overview, Definition, Data Warehousing Components, Building a Data Warehouse, Warehouse Database, Mapping the Data Warehouse to a Multiprocessor Architecture, Difference between Database System and Data Warehouse, Multi Dimensional Data Model, Data Cubes, Stars, Snow Flakes, Fact Constellations, Concept.	08
II	**Data Warehouse Process and Technology:** Warehousing Strategy, Warehouse /management and Support Processes, Warehouse Planning and Implementation, Hardware and Operating Systems for Data Warehousing, Client/Server Computing Model & Data Warehousing. Parallel Processors & Cluster Systems, Distributed DBMS implementations, Warehousing Software, Warehouse Schema Design	08
III	**Data Mining**: Overview, Motivation, Definition & Functionalities, Data Processing, Form of Data Pre-processing, Data Cleaning: Missing Values, Noisy Data, (Binning, Clustering, Regression, Computer and Human inspection), Inconsistent Data, Data Integration and Transformation. Data Reduction:-Data Cube Aggregation, Dimensionality reduction, Data Compression, Numerosity Reduction, Discretization and Concept hierarchy generation, Decision Tree	08
IV	**Classification:** Definition, Data Generalization, Analytical Characterization, Analysis of attribute relevance, Mining Class comparisons, Statistical measures in large Databases, Statistical-Based Algorithms, Distance-Based Algorithms, Decision Tree-Based Algorithms. **Clustering**: Introduction, Similarity and Distance Measures, Hierarchical and Partitional Algorithms. Hierarchical Clustering- CURE and Chameleon. Density Based Methods DBSCAN, OPTICS. Grid Based Methods- STING, CLIQUE. Model Based Method – Statistical Approach, Association rules: Introduction, Large Item sets, Basic Algorithms, Parallel and Distributed Algorithms, Neural Network approach.	08
V	**Data Visualization and Overall Perspective**: Aggregation, Historical information, Query Facility, OLAP function and Tools. OLAP Servers, ROLAP, MOLAP, HOLAP, Data Mining interface, Security, Backup and	

	Recovery, Tuning Data Warehouse, Testing Data Warehouse. Warehousing applications and Recent Trends: Types of Warehousing Applications, Web Mining, Spatial Mining and Temporal Mining.	08

Suggested Readings:

1. Alex Berson, Stephen J. Smith "Data Warehousing, Data-Mining & OLAP", TMH.
2. Mark Humphries, Michael W. Hawkins, Michelle C. Dy, "Data Warehousing: Architecture and Implementation", Pearson.
3. I.Singh, "Data Mining and Warehousing", Khanna Publishing House.
4. Margaret H. Dunham, S. Sridhar,"Data Mining:Introductory and Advanced Topics" Pearson Education 5. Arun K. Pujari, "Data Mining Techniques" Universities Press.
5. Pieter Adriaans, Dolf Zantinge, "Data-Mining", Pearson Education

MASTER OF COMPUTER APPLICATION (MCA)

BMC013: Software Project Management		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course, the student will be able to understand		
CO 1	Identify project planning objectives, along with various cost/effort estimation models.	K_3
CO 2	Organize & schedule project activities to compute critical path for risk analysis	K_3
CO 3	Monitor and control project activities.	K_4, K_5
CO 4	Formulate testing objectives and test plan to ensure good software quality under SEI-CMM	K_6
CO 5	Configure changes and manage risks using project management tools.	K_2, K_4
DETAILED SYLLABUS		**3-0-0**
Unit	**Topic**	**Proposed Lecture**
I	**Project Evaluation and Project Planning:** Importance of Software Project Management – Activities – Methodologies – Categorization of Software Projects – Setting objectives – Management Principles – Management Control – Project portfolio Management – Cost-benefit evaluation technology – Risk evaluation – Strategic program Management – Stepwise Project Planning.	**08**
II	**Project Life Cycle and Effort Estimation:** Software process and Process Models – Choice of Process models – Rapid Application development – Agile methods – Dynamic System Development Method – Extreme Programming– Managing interactive processes – Basics of Software estimation – Effort and Cost estimation techniques – COSMIC Full function points – COCOMO II – a Parametric Productivity Model.	**08**
III	**Activity Planning and Risk Management:** Objectives of Activity planning – Project schedules – Activities – Sequencing and scheduling – Network Planning models – Formulating Network Model – Forward Pass & Backward Pass techniques – Critical path (CRM) method – Risk identification – Assessment – Risk Planning –Risk Management – – PERT technique – Monte Carlo simulation – Resource Allocation – Creation of Critical paths – Cost schedules.	**08**
IV	**Project Management and Control:** Framework for Management and control – Collection of data – Visualizing progress – Costmonitoring – Earned Value Analysis – Prioritizing Monitoring – Project tracking – Change control Software Configuration Management – Managing contracts – Contract Management.	**08**
V	**Staffing in Software Projects:** Managing people – Organizational behavior – Best methods of staff selection – Motivation – The Oldham – Hackman job characteristic model – Stress – Health and Safety – Ethical and Professional concerns – Working in teams – Decision making – Organizational structures – Dispersed and Virtual teams – Communications genres – Communication plans – Leadership.	**08**

Suggested Readings:

1. Bob Hughes, Mike Cotterell and Rajib Mall: "Software Project Management" – Fifth Edition, McGraw Hill,New Delhi, 2012.
2. Robert K. Wysocki — "Effective Software Project Management" – Wiley Publication, 2011.
3. Walker Royce: — "Software Project Management" - Addison-Wesley, 1998.
4. Gopalaswamy Ramesh, — "Managing Global Software Projects" – McGraw Hill Education (India), FourteenthReprint 2013.
5. Koontz Harold & Weihrich Heinz, "Essentials of Management", McGraw Hill 5thEdition 2008.
6. Robbins and Coulter, "Management", Prentice Hall of India, 9th edition.
7. James A. F., Stoner, "Management", Pearson Education Delhi.
8. P. D. Chaturvedi, "Business Communication", Pearson Education.

BMC014: Cloud Computing		
Course Outcome (CO)	**Bloom's Knowledge Level (KL)**	
At the end of course, the student will be able to understand		
CO 1	Understand the concepts of Cloud Computing, key technologies, strengths and limitations of cloud computing.	K_1, K_2
CO 2	Develop the ability to understand and use the architecture to compute and storage cloud, service and models.	K_1, K_3
CO 3	Understand the application in cloud computing.	K_4, K_5
CO 4	Learn the key and enabling technologies that help in the development of cloud.	K_3, K_4
CO 5	Explain the core issues of cloud computing such as resource management and security.	K_2, K_6
DETAILED SYLLABUS		**3-1-0**
Unit	**Topic**	**Proposed Lecture**
I	**Introduction:** Cloud Computing – Definition of Cloud – Evolution of Cloud Computing – Underlying Principles of Parallel and Distributed, History of Cloud Computing - Cloud Architecture - Types of Clouds - Business models around Clouds – Major Players in Cloud Computing- issues in Clouds - Eucalyptus - Nimbus - Open Nebula, CloudSim.	08
II	**Cloud Services:** Types of Cloud services: Software as a Service- Platform as a Service –Infrastructure as a Service - Database as a Service - Monitoring as a Service –Communication as services. Service providers- Google, Amazon, Microsoft Azure, IBM, Sales force.	08
III	**Collaborating Using Cloud Services:** Email Communication over the Cloud - CRM Management – Project Management-Event Management - Task Management – Calendar - Schedules - Word Processing – Presentation – Spreadsheet - Databases – Desktop - Social Networks and Groupware.	08
IV	**Virtualization for Cloud:** Need for Virtualization – Pros and cons of Virtualization – Types of Virtualization –System VM, Process VM, Virtual Machine monitor – Virtual machine properties - Interpretation and binary translation, HLL VM - supervisors – Xen, KVM, VMware, Virtual Box, Hyper-V.	08
V	**Security, Standards and Applications:** Security in Clouds: Cloud security challenges – Software as a Service Security, Common Standards: The Open Cloud Consortium – The Distributed management Task Force – Standards for application Developers – Standards for Messaging – Standards for Security, End user access to cloud computing, Mobile Internet devices and the cloud. Hadoop – MapReduce – Virtual Box — Google App Engine – Programming Environment for Google App Engine	08

Suggested Readings:

1. David E.Y. Sarna, "Implementing and Developing Cloud Application", CRC press 2011.
2. Lee Badger, Tim Grance, Robert Patt-Corner, Jeff Voas, NIST, Draft cloud computing synopsis and recommendation, May 2011.
3. Anthony T Velte, Toby J Velte, Robert Elsenpeter, "Cloud Computing : A Practical Approach", Tata McGraw-Hill 2010.
4. Haley Beard, "Best Practices for Managing and Measuring Processes for On-demand Computing, Applications and Data Centers in the Cloud with SLAs", Emereo Pty Limited, July 2008.
5. G. J. Popek, R.P. Goldberg, "Formal requirements for virtualizable third generation Architectures, Communications of the ACM", No.7 Vol.17, July 1974

BMC015 : Compiler Design		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course , the student will be able to:		
CO 1	Acquire knowledge of different phases and passes of the compiler and also able to use the compiler tools like LEX, YACC, etc. Students will also be able to design different types of compiler tools to meet the requirements of the realistic constraints of compilers.	K_3, K_6
CO 2	Understand the parser and its types i.e. Top-Down and Bottom-up parsers and construction of LL, SLR, CLR, and LALR parsing table.	K_2, K_6
CO 3	Implement the compiler using syntax-directed translation method and get knowledge about the synthesized and inherited attributes.	K_4, K_5
CO 4	Acquire knowledge about run time data structure like symbol table organization and different techniques used in that.	K_2, K_3
CO 5	Understand the target machine's run time environment, its instruction set for code generation and techniques used for code optimization.	K_2, K_4
DETAILED SYLLABUS		**3-0-0**
Unit	**Topic**	**Proposed Lecture**
I	**Introduction to Compiler**: Phases and passes, Bootstrapping, Finite state machines and regular expressions and their applications to lexical analysis, Optimization of DFA-Based Pattern Matchers implementation of lexical analyzers, lexical-analyzer generator, LEX compiler, Formal grammars and their application to syntax analysis, BNF notation, ambiguity, YACC. The syntactic specification of programming languages: Context free grammars, derivation and parse trees, capabilities of CFG.	**08**
II	**Basic Parsing Techniques:** Parsers, Shift reduce parsing, operator precedence parsing, top down parsing, predictive parsers Automatic Construction of efficient Parsers: LR parsers, the canonical Collection of LR(0) items, constructing SLR parsing tables, constructing Canonical LR parsing tables, Constructing LALR parsing tables, using ambiguous grammars, an automatic parser generator, implementation of LR parsing tables.	**08**
III	**Syntax-directed Translation:** Syntax-directed Translation schemes, Implementation of Syntax-directed Translators, Intermediate code, postfix notation, Parse trees & syntax trees, three address code, quadruple & triples, translation of assignment statements, Boolean expressions, statements that alter the flow of control, postfix translation, translation with a top down parser. More about translation: Array references in arithmetic expressions, procedures call, declarations and case statements.	**08**
IV	**Symbol Tables**: Data structure for symbols tables, representing scope information. Run-Time Administration: Implementation of simple stack allocation scheme, storage allocation in block structured language. Error Detection & Recovery: Lexical Phase errors, syntactic phase errors semantic errors.	**08**
V	**Code Generation:** Design Issues, the Target Language. Addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, Code Generator. Code optimization: Machine-Independent Optimizations, Loop optimization, DAG representation of basic blocks, value numbers and algebraic laws, Global Data-Flow analysis.	**08**

Text books:

1. K. Muneeswaran,Compiler Design,First Edition,Oxford University Press.
2. J.P. Bennet, "Introduction to Compiler Techniques", Second Edition, Tata McGraw-Hill,2003.
3. Henk Alblas and Albert Nymeyer, "Practice and Principles of Compiler Building with C", PHI, 2001.
4. Aho, Sethi & Ullman, "Compilers: Principles, Techniques and Tools", Pearson Education
5. V Raghvan, " Principles of Compiler Design", TMH
6. Kenneth Louden," Compiler Construction", Cengage Learning.
7. Charles Fischer and Ricard LeBlanc," Crafting a Compiler with C", Pearson Education

ELECTIVE-2

BMC021: Artificial Intelligence		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course, the student will be able to understand		
CO 1	Define the meaning of intelligence and study various intelligent agents.	K_1
CO 2	Understand, analyze and apply AI searching algorithms in different problem domains.	K_2 , K_3, K_4
CO 3	Study and analyze various models for knowledge representation.	K_1, K_3
CO 4	Understand the basic concepts of machine learning to analyze and implement widely used learning methods and algorithms.	K_2 , K_4, K_6
CO 5	Understand the concept of pattern recognition and evaluate various classification and clustering techniques	K_2 , K_5
DETAILED SYLLABUS		**3-0-0**
Unit	**Topic**	**Proposed Lecture**
I	**Artificial Intelligence:** Introduction to artificial intelligence, Historical development and foundation areas of artificial intelligence, Tasks and application areas of artificial intelligence. Introduction, types and structure of intelligent agents, Computer Vision, Natural language processing.	**08**
II	**Searching Techniques:** Introduction, Problem solving by searching, Searching for solutions, Uniformed searching techniques, Informed searching techniques, Local search algorithms, Adversarial search methods, Search techniques used in games, Alpha-Beta pruning.	**08**
III	**Knowledge Representation and Reasoning:** Propositional logic, Predicate logic, First order logic, Inference in first order logic, Clause form conversion, Resolution. Chaining- concept, forward chaining and backward chaining, Utility theory and Probabilistic reasoning, Hidden Markov model, Bayesian networks.	**08**
IV	**Machine Learning:** Introduction, types and application areas, Decision trees, Statistical learning methods, Learning with complete data - concept and Naïve Bayes models, Learning with hidden data- concept and EM algorithm, Reinforcement learning.	**08**
V	**Pattern Recognition:** Introduction and design principles, Statistical pattern recognition, Parameter estimation methods - Principle component analysis and Linear discrimination analysis, Classification techniques - Nearest neighbor rule and Bayes classifier, K-means clustering, Support vector machine.	**08**

Suggested Readings:

1. Russell S. and Norvig P., "Artificial Intelligence – A Modern Approach", Pearson Education.
2. Rich E. and Knight K., "Artificial Intelligence", McGraw Hill Publications.
3. Charnik E. and McDermott D., "Introduction to Artificial Intelligence", Pearson Education.
4. Patterson D. W., "Artificial Intelligence and Expert Systems", Prentice Hall of India Publications.
5. Khemani D., "A First Course in Artificial Intelligence", McGraw Hill.
6. Winston P. H., "Artificial Intelligence", Pearson Education.
7. Thornton C. and Boulay B.," Artificial Intelligence- Strategies, Applications and Models through Search", New Age International Publishers.

BMC022: Big Data		
Course Outcome (CO)	**Bloom's Knowledge Level (KL)**	
At the end of course, the student will be able to understand		
CO1	Demonstrate knowledge of Big Data Analytics concepts and its applications in business.	K_1, K_2
CO2	Demonstrate functions and components of Map Reduce Framework and HDFS.	K_1, K_2
CO3	Develop queries in NoSQL environment.	K_6
CO4	Explain process of developing Map Reduce based distributed processing applications.	K_2, K_5
CO5	Explain process of developing applications using HBASE, Hive, Pig etc.	K_2,K_5
	DETAILED SYLLABUS	**4-0-0**
Unit	**Topic**	**Proposed Lecture**
I	**Introduction to Big Data**: Types of digital data, history of Big Data innovation, introduction to Big Data platform, drivers for Big Data, Big Data architecture and characteristics, 5 Vs of Big Data, Big Data technology components, Big Data importance and applications, Big Data features – security, compliance, auditing and protection, Big Data privacy and ethics, Big Data Analytics, Challenges of conventional systems, intelligent data analysis, nature of data, analytic processes and tools, analysis vs reporting, modern data analytic tools.	**08**
II	**Hadoop:** History of Hadoop, Apache Hadoop, the Hadoop Distributed File System, components of Hadoop, data format, analyzing data with Hadoop, scaling out, Hadoop streaming, Hadoop pipes, Hadoop Echo System. **Map-Reduce:** Map-Reduce framework and basics, how Map Reduce works, developing a Map Reduce application, unit tests with MR unit, test data and local tests, anatomy of a Map Reduce job run, failures, job scheduling, shuffle and sort, task execution, Map Reduce types, input formats, output formats, Map Reduce features, Real-world Map Reduce	**08**
III	**HDFS (Hadoop Distributed File System):** Design of HDFS, HDFS concepts, benefits and challenges, file sizes, block sizes and block abstraction in HDFS, data replication, how does HDFS store, read, and write files, Java interfaces to HDFS, command line interface, Hadoop file system interfaces, data flow, data ingest with Flume and Scoop, Hadoop archives, Hadoop I/O: Compression, serialization, Avro and file-based data structures. Hadoop Environment: Setting up a Hadoop cluster, cluster specification, cluster setup and installation, Hadoop configuration, security in Hadoop, administering Hadoop, HDFS monitoring & maintenance, Hadoop benchmarks, Hadoop in the cloud	**08**
IV	**Hadoop Eco System and YARN:** Hadoop ecosystem components, schedulers, fair and capacity, Hadoop 2.0 New Features – Name Node high availability, HDFS federation, MRv2, YARN, Running MRv1 in YARN. **NoSQL** Databases: Introduction to NoSQL MongoDB: Introduction, data types, creating, updating and deleing documents, querying, introduction to indexing, capped collections **Spark**: Installing spark, spark applications, jobs, stages and tasks, Resilient Distributed Databases, anatomy of a Spark job run, Spark on YARN **SCALA:** Introduction, classes and objects, basic types and operators, built-in control structures, functions and closures, inheritance.	**08**

V	**Hadoop Eco System Frameworks**: Applications on Big Data using Pig, Hive and HBase **Pig** : Introduction to PIG, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin, User Defined Functions, Data Processing operators, **Hive** - Apache Hive architecture and installation, Hive shell, Hive services, Hive metastore, comparison with traditional databases, HiveQL, tables, querying data and user defined functions, sorting and aggregating, Map Reduce scripts, joins & subqueries. **HBase** – Hbase concepts, clients, example, Hbase vs RDBMS, advanced usage, schema design, advance indexing, Zookeeper – how it helps in monitoring a cluster, how to build applications with Zookeeper. IBM Big Data strategy, introduction to Infosphere, BigInsights and Big Sheets, introduction to Big SQL.	**08**

Suggested Readings:

1. Michael Minelli, Michelle Chambers, and Ambiga Dhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses", Wiley.
2. Big-Data Black Book, DT Editorial Services, Wiley.
3. Dirk deRoos, Chris Eaton, George Lapis, Paul Zikopoulos, Tom Deutsch, "Understanding Big Data Analytics for Enterprise Class Hadoop and Streaming Data", McGrawHill.
4. Thomas Erl, Wajid Khattak, Paul Buhler, "Big Data Fundamentals: Concepts, Drivers and Techniques", Prentice Hall.
5. Bart Baesens "Analytics in a Big Data World: The Essential Guide to Data Science and its Applications (WILEY Big Data Series)", John Wiley & Sons
6. Arshdeep Bahga, Vijay Madisetti, "Big Data Science & Analytics: A Hands On Approach ", VPT
7. Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", CUP
8. Tom White, "Hadoop: The Definitive Guide", O'Reilly.
9. Eric Sammer, "Hadoop Operations", O'Reilly.
10. Chuck Lam, "Hadoop in Action", MANNING Publishers
11. Deepak Vohra, "Practical Hadoop Ecosystem: A Definitive Guide to Hadoop-Related Frameworks and Tools", Apress
12. E. Capriolo, D. Wampler, and J. Rutherglen, "Programming Hive", O'Reilly
13. Lars George, "HBase: The Definitive Guide", O'Reilly.
14. Alan Gates, "Programming Pig", O'Reilly.
15. Michael Berthold, David J. Hand, "Intelligent Data Analysis", Springer.
16. Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", John Wiley & sons.
17. Glenn J. Myatt, "Making Sense of Data", John Wiley & Sons
18. Pete Warden, "Big Data Glossary", O'Reilly

BMC023 : Simulation and Modelling		
Course Outcome (CO)	**Bloom's Knowledge Level (KL)**	
At the end of course , the student will be able to understand		
CO 1	Study the concept of system, its components and types.	K_1
CO 2	Understand and analyze nature and techniques of major simulation models.	K_2, K_4
CO 3	Study and analyze the idea of continuous and discrete system simulation.	K_1, K_4
CO 4	Understand the notion of system dynamics and system dynamics diagrams.	K_2
CO 5	Finding critical path computation and understanding PERT networks	K_1, K_4
DETAILED SYLLABUS		**3-1-0**
Unit	**Topic**	**Proposed Lecture**
I	System definition and components, stochastic activities, continuous and discrete systems, System modeling, Types of models, static and dynamic physical models, static and dynamic mathematical models, full corporate model, types of system study.	**08**
II	System simulation, Need of simulation, Basic nature of simulation, techniques of simulation, comparison of simulation and analytical methods, types of system Simulation, real time simulation, hybrid simulation, simulation of pursuit problem, single-server queuing system and an inventory problem, Monte-Carlo simulation, Distributed Lag model, Cobweb model.	**08**
III	Simulation of continuous Systems, analog vs digital simulation, simulation of water reservoir system, simulation of a servo system, simulation of an auto-pilot. Discrete system simulation, fixed time step vs. event-to-event model, generation of random numbers, test of randomness, Monte-Carlo computation vs. stochastic simulation.	**08**
IV	System dynamics, exponential growth models, exponential decay models, logistic curves, system dynamics diagrams, world model.	**08**
V	Simulation of PERT networks, critical path computation, uncertainties in activity duration, resource allocation and consideration, Simulation languages, object oriented simulation	**08**

Suggested Readings:

1. Geoffrey Gordon, "System Simulation", PHI
2. Narsingh Deo, "System Simulation with digital computer", PHI.
3. Averill M. Law and W. David Kelton, "Simulation Modelling and Analysis", TMH.

BMC024: Software Testing & Quality Assurance		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course, the student will be able to understand		
CO 1	Test the software by applying testing techniques to deliver a product free from bugs.	K_3
CO 2	Investigate the scenario and select the proper testing technique.	K_1, K_4
CO 3	Explore the test automation concepts and tools and estimation of cost, schedule based on standard metrics.	K_2, K_4
CO 4	Understand how to detect, classify, prevent and remove defects.	K_1, K_2
CO 5	Choose appropriate quality assurance models and develop quality. Ability to conduct formal inspections, record and evaluate results of inspections.	K_3, K_4
DETAILED SYLLABUS		**3-0-0**
Unit	**Topic**	**Proposed Lecture**
I	**Software Testing Basics:** Testing as an engineering activity, Role of process in software quality, Testing as a process, Basic definitions, Software testing principles, The tester's role in a software development organization, Origins of defects, Defect classes, The defect repository and test design, Defect examples, Developer / Tester support for developing a defect repository.	**08**
II	**Testing Techniques and Levels of Testing:** Using White Box Approach to Test design– Static Testing Vs. Structural Testing, Code Functional Testing, Coverage and Control Flow Graphs, Using Black Box Approaches to Test Case Design, Random Testing, Requirements based testing, Decision tables, State-based testing, Cause-effect graphing, Error guessing, Compatibility testing, Levels of Testing -Unit Testing, Integration Testing, Defect Bash Elimination. System Testing - Usability and Accessibility Testing, Configuration Testing, Compatibility Testing.	**08**
III	**Software Test Automation And Quality Metrics:** Software Test Automation, Skills needed for Automation, Scope of Automation, Design and Architecture for Automation, Requirements for a Test Tool, Challenges in Automation Tracking the Bug, Debugging. Testing Software System Security - Six-Sigma, TQM - Complexity Metrics and Models, Quality Management Metrics, Availability Metrics, Defect Removal Effectiveness, FMEA, Quality Function Deployment, Taguchi Quality Loss Function, Cost of Quality.	**08**
IV	**Fundamentals of Software Quality Assurance:** SQA basics, Components of the Software Quality Assurance System, software quality in business context, planning for software quality assurance, product quality and process quality, software process models, 7 QC Tools and Modern Tools.	**08**
V	**Software Assurance Models:** Models for Quality Assurance, ISO-9000 series, CMM, CMMI, Test Maturity Models, SPICE, Malcolm Baldrige Model- P-CMM. **Software Quality Assurance Trends:** Software Process- PSP and TSP, OO Methodology, Clean room software engineering, Defect Injection and prevention, Internal Auditing and Assessments, Inspections & Walkthroughs, Case Tools and their affect on Software Quality.	**08**

Suggested Readings:

1. Srinivasan Desikan, Gopalaswamy Ramesh, "Software Testing: Principles and Practices", Pearson.
2. Daniel Galin, "Software Quality Assurance: From Theory to Implementation", Pearson

Addison Wesley.
3. Aditya P. Mathur, “Foundations of Software Testing”, Pearson.
4. Paul Ammann, Jeff Offutt, “Introduction to Software Testing”, Cambridge University Press.
5. Paul C. Jorgensen, “Software Testing: A Craftsman's Approach”, Auerbach Publications.
6. William Perry, “Effective Methods of Software Testing”, Wiley Publishing, Third Edition.
7. Renu Rajani, Pradeep Oak, “Software Testing – Effective Methods, Tools and Techniques”, Tata McGraw Hill.
8. Stephen Kan, “Metrics and Models in Software Quality”, Addison – Wesley, Second Edition.
9. S. A. Kelkar, “Software quality and Testing”, PHI Learning Pvt, Ltd.
10. Watts S Humphrey, “Managing the Software Process”, Pearson Education Inc.

BMC025: Digital Image Processing		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course, the student will be able to understand		
CO 1	Explain the basic concepts of two-dimensional signal acquisition, sampling, quantization and color model.	K_1, K_2
CO 2	Apply image processing techniques for image enhancement in both the spatial and frequency domains.	K_2, K_3
CO 3	Apply and compare image restoration techniques in both spatial and frequency domain.	K_2, K_3
CO 4	Compare edge based and region based segmentation algorithms for ROI extraction.	K_3, K_4
CO 5	Explain compression techniques and descriptors for image processing.	K_2, K_3
DETAILED SYLLABUS		**3-0-0**
Unit	**Topic**	**Proposed Lecture**
I	**Digital Image Fundamentals:** Steps in Digital Image Processing – Components – Elements of Visual Perception – Image Sensing and Acquisition – Image Sampling and Quantization – Relationships between pixels – Color image fundamentals – RGB, HSI models, Two-dimensional mathematical preliminaries, 2D transforms – DFT, DCT.	**08**
II	**Image Enhancement:** Spatial Domain: Gray level transformations – Histogram processing – Basics of Spatial Filtering–Smoothing and Sharpening Spatial Filtering, Frequency Domain: Introduction to Fourier Transform– Smoothing and Sharpening frequency domain filters – Ideal, Butterworth and Gaussian filters, Homomorphic filtering, Color image enhancement.	**08**
III	**Image Restoration:** Image Restoration – degradation model, Properties, Noise models – Mean Filters – Order Statistics –Adaptive filters – Band reject Filters – Band pass Filters – Notch Filters – Optimum Notch Filtering – Inverse Filtering – Wiener filtering	**08**
IV	**Image Segmentation:** Edge detection, Edge linking via Hough transform – Thresholding – Region based segmentation – Region growing – Region splitting and merging – Morphological processing- erosion and dilation, Segmentation by morphological watersheds – basic concepts – Dam construction – Watershed segmentation algorithm.	**08**
V	**Image Compression and Recognition:** Need for data compression, Huffman, Run Length Encoding, Shift codes, Arithmetic coding, JPEG standard, MPEG. Boundary representation, Boundary description, Fourier Descriptor, Regional Descriptors – Topological feature, Texture – Patterns and Pattern classes – Recognition based on matching.	**08**

Suggested Readings:

1. Rafael C. Gonzalez, Richard E. Woods, "Digital Image Processing", Pearson, Third Edition, 2010.
2. Anil K. Jain, "Fundamentals of Digital Image Processing", Pearson, 2002.
3. Kenneth R. Castleman, "Digital Image Processing" Pearson, 2006.
4. D, E. Dudgeon and R M. Mersereau, "Multidimensional Digital Signal Processing", Prentice Hall Professional Technical Reference, 1990.
5. William K. Pratt, "Digital Image Processing" John Wiley, New York, 2002.
6. Milan Sonka et al, "Image processing, analysis and machine vision Brookes/Cole", Vikas Publishing House, 2nd edition,1999.

BMC351: Python Programming Lab		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course, the student will be able to understand		
CO 1	Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.	K1, K2
CO 2	Express proficiency in the handling of strings and functions	K1, K2
CO 3	Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples and sets.	K3
CO 4	Use OO concepts while programming in Python	K1, K2
CO 5	Work with Python using GUI.	K4

Programs based on the concepts of:

1. Building Python Modules
2. Obtaining user Data
3. Printing desired output

Programs based on the concepts of:

1. Conditional if statements
2. Nested if statements
3. Using else if and elif

Programs based on the concepts of Iteration using different kinds of loops Usage of Data Structures:

1. Strings
2. Lists
3. Tuples
4. Sets
5. Dictionary

Program based on the concepts of User-defined modules and Standard Library (random, numpy, scipy, sys, Math Module, String Module, List Module).

Program based on Input Output.

Program based on exception Handling.

Program based on Simple Data Analysis

Program based on Pandas.

Note: The Instructor may add/delete/modify/tune experiments, wherever he/she feels in a justified manner.

BMC352: Software Engineering Lab		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course, the student will be able to understand		
CO 1	Identify ambiguities, inconsistencies and incompleteness from a requirements specification and state functional and non-functional requirement.	K_2, K_4
CO 2	Identify different actors and use cases from a given problem statement and draw use case diagram to associate use cases with different types of relationship.	K_3, K_5
CO 3	Draw a class diagram after identifying classes and association among them.	K_4, K_5
CO 4	Graphically represent various UML diagrams and associations among them and identify the logical sequence of activities undergoing in a system, and represent them pictorially.	K_4, K_5
CO 5	Able to use modern engineering tools for specification, design, implementation and testing.	K_3, K_4

DETAILED SYLLABUS

For any given case/ problem statement do the following;

1. Prepare a SRS document in line with the IEEE recommended standards.
2. Draw the use case diagram and specify the role of each of the actors.
3. Prepare state the precondition, post condition and function of each use case.
4. Draw the activity diagram.
5. Identify the classes. Classify them as weak and strong classes and draw the class diagram.
6. Draw the sequence diagram for any two scenarios.
7. Draw the collaboration diagram.
8. Draw the state chart diagram.
9. Draw the component diagram.
10. Draw the deployment diagram.

Note: The Instructor may add/delete/modify/tune experiments, wherever he/she feels in a justified manner.