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

National Education Policy 2020

[Effective from the Session: 2025-26]

MASTER OF COMPUTER APPLICATION (MCA)

MCA SECOND YEAR, 2021-22

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.

MASTER OF COMPUTER APPLICATION (MCA)

Elective-1	BMC011	Cryptography & Network Security
	BMC012	Data Warehousing & Data Mining
	BMC013	Software Project Management
	BMC014	Cloud Computing
	BMC015	Compiler Design

Elective-2	BMC021	Artificial Intelligence
	BMC022	Big Data
	BMC023	Simulation & Modeling
	BMC024	Software Testing & Quality Assurance
	BMC025	Digital Image Processing

Elective-3	BMC031	Privacy & Security in Online Social Media
	BMC032	Soft Computing
	BMC033	Pattern Recognition
	BMC034	Data Analytics
	BMC035	Software Quality Engineering

Elective-4	BMC041	Blockchain Architecture
	BMC042	Neural Network
	BMC043	Internet of Things
	BMC044	Modern Application Development
	BMC045	Distributed Database Systems

Elective-5	BMC051	Mobile Computing
	BMC052	Computer Graphics and Animation
	BMC053	Natural Language Processing
	BMC054	Machine Learning
	BMC055	Quantum Computing

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

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

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.

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.