MANGALORE UNIVERSITY

National Education Policy – 2020
[NEP-2020]

CURRICULUM STRUCTURE
FOR

Syllabus for III and IV semester
BCA-Artificial Intelligence and Machine Learning

CURRICULUM STRUCTURE FOR III AND IV SEMESTER FOR BCA-ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Semester	Course No	Theory/Practical	Credits	Paper Title	S.A	I.A
III	BCA-AIML-C11T	Theory	**3**	Computer Networks	**60**	**40**
	BCA-AIML-C12T	Theory	**3**	Operating Systems	**60**	**40**
	BCA-AIML-C13T	Theory	**3**	Python Programming	**60**	**40**
	BCA-AIML-C14P	Practical	**2**	Operating Systems Lab	**25**	**25**
	BCA-AIML-C15P	Practical	**2**	Python Programming Lab	**25**	**25**
IV	BCA-AIML-C16T	Theory	**3**	Software Engineering	**60**	**40**
	BCA-AIML-C17T	Theory	**3**	Artificial Intelligence-1	**60**	**40**
	BCA-AIML-C18T	Theory	**3**	Internet of Things	**60**	**40**
	BCA-AIML-C19P	Practical	**2**	Artificial Intelligence-1 Lab	**25**	**25**
	BCA-AIML-C20P	Practical	**2**	Internet of Things Lab	**25**	**25**

SEMESTER - III

Program Name	**BCA-AIML**	Semester	**III**
Course Title	**Computer Networks (Theory)**		
Course Code:	**BCA-AIML-C11T**	No.of Credits	**03**
Contact hours	**3 Hours per week**	Duration of SEA/Exam	**2 hours**
Formative Assessment Marks	**40**	Summative Assessment Marks	**60**

Course Outcomes (COs):

At the end of the course, students will be able to:

- Explain the transmission technique of digital data between two or more computers and a computer network that allows computers to exchange data.
- Apply the basics of data communication and various types of computer networks in real world applications.
- Compare the different layers of protocols.
- Compare the key networking protocols and their hierarchical relationship in the conceptual model like TCP/IP and OSI.

Unit	Description	Hours
1	**Introduction**, Uses of Computer Networks-Business Applications, Home Applications, Mobile Users, Social Issues; **Network Hardware**-Local Area Networks, Wide Area Networks, Wireless Networks, Home Networks, Internetworks; **Network Software**-Protocol Hierarchies, Connection-oriented and Connectionless Services, **Reference Models**-the OSI Reference Model, The TCP/IP Reference Model, A Comparison of the OSI and TCP/IP Reference Models. **Example Networks**-The Internet (Architecture of the Internet), Ethernet, Wireless LANs:801.11	**11**

2	**The Physical Layer**-Transmission Media-Twisted-Pair, Coaxial Cable, Fiber Optics, Radio Transmission, Microwave Transmission, Infrared & Millimeter Waves; Switching (Circuit Switching, Message Switching, Packet Switching); **Data Link Layer, Design Issues**-Services Provided to the Network Layer, Framing, Error Control, Flow Control; **Data Link Protocols**-An Unrestricted Simplex Protocol, HDLC;;	**11**
3	**The Network Layer**-Services Provided to the Transport Layer, Implementation of Connectionless Service, and Implementation of Connection-Oriented Service. Routing Algorithms-Shortest Path Algorithm, Distance Vector Routing; **Internetworking**-How Networks can be connected, Connectionless Internetworking, Tunneling, Internetwork Routing, Fragmentation; **The Network Layer in the Internet**-the IP Protocol, IP Addresses, OSPF, BGP, IPv6. **The Transport Layer**-Services Provided to the Upper Layers, Transport Service Primitives; **Elements of Transport Protocols**-Addressing, Connection Establishment, Connection Release; **The Internet Transport Protocols**-UDP-Introduction to UDP; **TCP**-Introduction to TCP, The TCP Service Model, The TCP Protocol, TCP Connection Establishment, TCP Connection Release, TCP Connection Management, TCP Transmission Policy, TCP Congestion Control.	**10**
4	**The Application Layer-DNS**-The DNS Name Space, Resource Records, Name Servers; **Electronic Mail**-Architecture and Services, The User Agent, Message Formats, Message Transfer, Final Delivery; The World Wide Web-Architectural Overview; **Network Security**-Introduction to Cryptography, Substitution Ciphers, Transposition Ciphers, Two Fundamental Cryptographic Principles; **Digital Signatures**-Symmetric-Key Signature, Public-Key Signature; Wireless Security, **Web Security**-Threats, SSL.	**10**

Text Book:

1. Computer Networks, Andrew S. Tanenbaum, 4th Edition, Pearson Education, 2010.

Reference Books:

1. Data Communication & Networking, Behrouza A Forouzan, 3rd Edition, Tata McGraw Hill, 2001.
2. Data and Computer Communications, William Stallings, 10th Edition,

Pearson Education, 2017.
3. Data Communication and Computer Networks, Brijendra Singh, 3rd Edition, PHI, 2012.
4. Data Communication & Network, Dr. Prasad, Wiley Dreamtech.
5. http://highered.mheducation.com/sites/0072967757/index.htmls

Pedagogy: Lecture/ PPT/ Videos/ Animations/ Role Plays/ Think-Pair-Share/ Predict-Observe- Explain/ Demonstration/ Concept mapping/ Case Studies examples/ Tutorial/ Activity/ Flipped Classroom/ Jigsaw/ Field based Learning/ Project Based Learning/ Mini Projects/ Hobby Projects/ Forum Theatre/ Dance/ Problem Based Learning/ Game Based Learning/ Group Discussion/ Collaborative Learning/ Experiential Learning / Self Directed Learning etc

Program Name	**BCA-AIML**	Semester	**III**
Course Title	**Operating Systems (Theory)**		
Course Code:	**BCA-AIML-C12T**	No.of Credits	**03**
Contact hours	**3 Hours per week**	Duration of SEA/Exam	**2 hours**
Formative Assessment Marks	**40**	Summative Assessment Marks	**60**

Course Outcomes (COs):

At the end of the course, students will be able to:

- Explain the functions and structure of an operating system
- Understand and implement process scheduling and synchronization mechanisms
- Demonstrate knowledge of memory management and virtual memory concepts
- Gain the ability to manage file systems and storage effectively.
- Evaluate and optimize operating system performance

Unit	Description	Hours
1	**Introduction to Operating System:** Definition, Types of Operating Systems, System Components, Operating System Services, Systems calls, Operating System Structure. **Process Management:** Process Concept, Process Scheduling, Operations on Processes, Cooperating Process, Inter Process Communication (IPC). **Threads:** Introduction to threads, Multithreading Models	**11**
2	**CPU Scheduling:** Basic Concepts, Scheduling Criteria, Scheduling Algorithms (FCFS, SJF, Priority and Round Robin scheduling), Multiple processor scheduling, Thread scheduling, Multiprocessor scheduling, Real Time CPU scheduling. **Process Synchronization:** The Critical SectionProblem,Semaphores(TheClassical Definition of Wait & Signal, Binary	**11**

	Semaphores)ClassicalProblemsofSynchronization- Readers and Writers problem, Dining Philosopher Problem.	
3	**Deadlocks:**DeadlockCharacterization, Methods of Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock. **Memory Management:** Logical versus Physical Address Space, Swapping, Contiguous Allocation (MemoryAllocation,Fragmentation),Paging, Segmentation.	**10**
4	**Virtual Memory:** Demand Paging, Page Replacement, Page Replacement Algorithms, Allocation of Frames (Equal andProportionalAllocation), Thrashing(concept). **File System:** File Concept, Access Methods, Directory Structure, File System Structure, Allocation Methods, Free-Space Management.	**10**

Text Book:

1. Silberschatz, A., & Galvin, P. (2002). Operating System Concepts (6th ed.). McGraw-Hill.

Reference Books:

1. Tanenbaum, A. S., & Bos, H. (2014). Modern Operating Systems (4th ed.). Pearson.
2. Stallings, W. (2018). Operating Systems: Internals and Design Principles (9th ed.). Pearson.
3. Bovet, D. P., & Cesati, M. (2005). Understanding the Linux Kernel (3rd ed.). O'Reilly Media.
4. Deitel, H. M., Deitel, P. J., & Choffnes, D. R. (2003). Operating Systems (3rd ed.). Pearson.
5. Tanenbaum, A. S., & Woodhull, A. S. (2006). Operating Systems Design and Implementation (3rd ed.). Prentice Hall.

Pedagogy: Lecture/ PPT/ Videos/ Animations/ Role Plays/ Think-Pair-Share/ Predict-Observe- Explain/ Demonstration/ Concept mapping/ Case Studies examples/ Tutorial/ Activity/ Flipped Classroom/ Jigsaw/ Field based Learning/ Project Based Learning/ Mini Projects/ Hobby Projects/ Forum Theatre/ Dance/ Problem Based Learning/ Game Based Learning/ Group Discussion/ Collaborative Learning/ Experiential Learning / Self Directed Learning etc

Program Name	**BCA-AIML**	Semester	**III**
Course Title	**Python Programming (Theory)**		
Course Code:	**BCA-AIML-C13T**	No.of Credits	**03**
Contact hours	**3 Hours per week**	Duration of SEA/Exam	**2 hours**
Formative Assessment Marks	**40**	Summative Assessment Marks	**60**

Course Outcomes (COs):

At the end of the course, students will be able to:

- Explain the basic concepts of Python Programming.
- Demonstrate proficiency in the handling of loops and creation of functions.
- Identify the methods to create and manipulate lists, tuples and dictionaries.
- Discover the commonly used operations involving file handling.
- Interpret the concepts of Object-Oriented Programming as used in Python.
- Develop the emerging applications of relevant fields using Python.

Unit	Description	Hours
1	**Introduction to Python-** Features of Python, Flavors of python, Python Virtual machine, Memory management, Garbage Collection, Comparison between Python and C, Java and Python, Installing Python for Windows, Writing and executing Python program. **Python Basics**: Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence andAssociation, Data Types, Indentation, Comments, Console Input and Console Output, Type Conversions. **Input &Output**: Input/output Statements, Command line arguments. **Control Statements** – if, if..else, if..elif, while loop, for loop, else suite, break, continue, assert, return Statements **Python Functions**: Built in Functions. User-defined functions, Definition- Syntax, Function Calling, Passing	**11**

	Parameters/arguments, the return statement, Scope and Lifetime of Variables in Functions,Default Parameters; Key Word Arguments, Command line Arguments.	
2	**Strings**: Creating and Storing Strings, Accessing Sting Characters; the str()function; Operations on Strings-Concatenation, Comparison, Slicingand Joining, Traversing, PythonString Methods **Lists**: Creating Lists, Operations onLists, Built-in Functions on Lists, Implementation of Stacks and Queuesusing Lists, Nested Lists **Dictionaries**: Creating Dictionaries, Operations on Dictionaries, Built-inFunctions on Dictionaries, DictionaryMethods, Populating and Traversing Dictionaries. **Tuples and Sets**: Creating Tuples, Operations on Tuples, Built-inFunctions on Tuples, Tuple Methods, Creating Sets, Operations on Sets, Built-in Functions on Sets, SetMethods	**11**
3	**FileHandling**: File Types, Operations on Files– Create, Open, Read, Write, Close Files, FileNames and Paths **Classes and Objects**-Defining classes& Objects, constructors, types of methods and variables, Inner classes. **Inheritance and Polymorphism**: Type of Inheritance, super () method, method overloading & Overriding, Operator Overloading **Exception Handling** –Type of exceptions, assert Statement, Except Block, User defined exceptions, logging the exceptions.	**10**
4	**Graphical User Interface**: Root window, font & colors, Canvas and frames. Widgets: Button, Label, Message, Text, Scrollbar, Checkbutton, Radiobutton, Entry, Spinbox, Listbox and Menu, Creating Tables. **Database Connectivity**: Types of databases used with Python, Using MySQL from Python, Retrieving and Inserting, updating and deleting data in a table, Creating Database tables through Python. Using Oracle database from Python Stored Procedures **Data Analysis**: NumPy- Introductionto NumPy, Array Creation usingNumPy, Operations on Arrays;Pandas-Introduction to Pandas, Seriesand DataFrames.	**10**

Text Books:

1. Dr. R. Nageswara Rao, Core Python Programming (2nd ed.). DreamTech Press.
2. Gowrishankar, S., &Veena, A. (2019). Introduction to Python Programming. CRC Press.
3. Dr. R. Nageswara Rao, (2021),Core Python Programming (3rd ed.). DreamTech Press.

Reference Books:

1. Brown, M. C. (2018). Python: The complete reference. McGraw Hill Education
2. Summerfield, M. (2010). Programming in Python 3: A complete introduction to the Python language (2nd ed.). Addison-Wesley.
3. Zelle, J. M. (2010). Python programming: An introduction to computer science (2nd ed.). Franklin, Beedle& Associates Inc.
4. Lutz, M. (2013). Learning Python (5th ed.). O'Reilly Media.
5. Matthes, E. (2019). Python crash course: A hands-on, project-based introduction to programming (2nd ed.). No Starch Press.
6. Ramalho, L. (2015). Fluent Python: Clear, concise, and effective programming. O'Reilly Media.

Pedagogy: Lecture/ PPT/ Videos/ Animations/ Role Plays/ Think-Pair-Share/ Predict-Observe- Explain/ Demonstration/ Concept mapping/ Case Studies examples/ Tutorial/ Activity/ Flipped Classroom/ Jigsaw/ Field based Learning/ Project Based Learning/ Mini Projects/ Hobby Projects/ Forum Theatre/ Dance/ Problem Based Learning/ Game Based Learning/ Group Discussion/ Collaborative Learning/ Experiential Learning / Self Directed Learning etc

Program Name	**BCA-AIML**	Semester	**III**
Course Title	**Operating Systems Lab**		
Course Code:	**BCA-AIML-C14P**	No.of Credits	**02**
Contact hours	**4 Hours per Week**	Duration of SEA/Exam	**3 hours**
Formative Assessment Marks	**25**	Summative Assessment Marks	**25**

PART A

PART B

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	**PART-A** **Writing:5 Marks Execution:3Marks**	**15 Marks**
Program-2	**PART-B** **Writing:8 Marks Execution:4Marks**	**20 Marks**
Practical Record		**05 Marks**
Total		**40Marks**

Program Name	**BCA-AIML**	Semester	**III**
Course Title	**Python Programming Lab**		
Course Code:	**BCA-AIML-C15P**	No.of Credits	**02**
Contact hours	**4 Hours per Week**	Duration of SEA/Exam	**3 hours**
Formative Assessment Marks	**25**	Summative Assessment Marks	**25**

PART A

PART B

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	**PART-A** **Writing:5 Marks Execution:5Marks**	**8 Marks**
Program-2	**PART-B** **Writing:6 Marks Execution:6Marks**	**12Marks**
Practical Record		**05 Marks**
Total		**25Marks**

SEMESTER IV

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Software Engineering (Theory)**		
Course Code:	**BCA-AIML-C16T**	No.of Credits	**03**
Contact hours	**3 Hours per week**	Duration of SEA/Exam	**2 hours**
Formative Assessment Marks	**40**	Summative Assessment Marks	**60**

Course Outcomes (COs):

At the end of the course, students will be able to:

- Explain the basic concepts and importance of software engineering.
- Describe various software development life cycle models.
- Analyze and specify software requirements for a project
- Design and architect software systems effectively
- Apply software testing techniques and use testing tools
- Manage software projects, including planning, scheduling, and risk management
- Ensure software quality through various assurance techniques

Unit	Description	Hours
1	**Introduction:** The Problem Domain- Industrial Strength Software, Software is Expensive, Late and Unreliable, Maintenance and Rework, Software Engineering Challenges-Scale, Quality and Productivity, Consistency and Repeatability, Change, The Software Engineering Approach-Phased Development Process, Managing the process. **Software Processes:** Software Process-Processes and Process Models, Component Software Processes, ETVX Approach for	**11**

	Process Specification, Desired Characteristics of a Software Process-Predictability, Support Testability and Maintainability, Support Change, Early Defect Removal, Process Improvement and Feedback , Software Development Process Models-Waterfall Model, Prototyping, Iterative Development, Time boxing Model, Comparison of Models, Other software Processes-Project Management Process, The Inspection Process, Software Configuration Management Process, Requirements Change Management Process, Process Management Process.	
2	**Software Requirements Analysis and Specification:** Software Requirements-Needs for SRS, Requirement Process, Problem Analysis -Informal Approach, Data Flow Modeling, Prototyping, Requirements Specification- Characteristics of an SRS, Components of an SRS, Specification Language, Structure of a Requirement Document, Validation **Function Oriented Design:** Design Principles-Problem Partitioning and Hierarchy, Abstraction, Modularity, Top-down and Bottom-up strategies, Module- Level Concepts-Coupling, Cohesion,, Design Notation and Specification-Structure Charts, Specification, Structured Design Methodology-Restate the Problem as a DFD, Identify the Most Abstract Input and Output Data Elements, First Level Factoring, Factoring the Input, Output and Transform Branches, Design Heuristics, Transaction Analysis, Verification.	**11**
3	**Detailed Design:** Detailed Design and PDL-PDL, Logic/Algorithm Design, State Modeling of Classes, Verification-Design Walkthroughs, Critical Design Review, and Consistency Checkers. **Coding:** Programming Principles and Guidelines-Common Coding Errors, Structured Programming, Information Hiding, Some Programming Practices, Coding Standards, Verification-Code Inspections, Static Analysis, Proving Correctness, Unit Testing, Combining Different Techniques.	**10**
4	**Testing and Testing Tools**: Testing Fundamentals-Error, Fault and Failure, Test Oracles, Test Cases and Test Criteria, Psychology of Testing, Black Box Testing- Equivalence Class Partitioning, Boundary Value Analysis, Cause-Effect Graphing, Pairwise Testing, Special Cases, State-Based Testing, White Box Testing-Control Flow Based Criteria, Data Flow Based Testing, Mutation Testing, Test Case Generation and Tool	**10**

	Support, Testing Process-Levels of Testing, Test Plan, Test Case Specification, Test Case Execution and Analysis, Defect Logging and Tracking. **Introduction to Testing tools:** Overview of WinRunner, Silk Test, SQA Robot, LoadRunner, JMeter and Test Director (relevant sections only).	

Text Books:

1. Jalote, P. (2005). An Integrated Approach to Software Engineering (3rd ed.). Narosa Publishing House.
2. Prasad, K. V. K. K. (2004). Software Testing Tools. Dreamtech Press.

Reference Books:

1. Sommerville, I. (2015). Software Engineering (10th ed.). Pearson.
2. Pressman, R. S. (2014). Software Engineering: A Practitioner's Approach (8th ed.). McGraw-Hill Education.
3. Fairley, R. (2009). Managing and Leading Software Projects. John Wiley & Sons.
4. Van Vliet, H. (2008). Software Engineering: Principles and Practice (3rd ed.). Wiley.
5. Pfleeger, S. L., & Atlee, J. M. (2009). Software Engineering: Theory and Practice (4th ed.). Prentice Hall.

Pedagogy: Lecture/ PPT/ Videos/ Animations/ Role Plays/ Think-Pair-Share/ Predict-Observe- Explain/ Demonstration/ Concept mapping/ Case Studies examples/ Tutorial/ Activity/ Flipped Classroom/ Jigsaw/ Field based Learning/ Project Based Learning/ Mini Projects/ Hobby Projects/ Forum Theatre/ Dance/ Problem Based Learning/ Game Based Learning/ Group Discussion/ Collaborative Learning/ Experiential Learning / Self Directed Learning etc

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Artificial Intelligence-1(Theory)**		
Course Code:	**BCA-AIML-C17T**	No.of Credits	**03**
Contact hours	**3 Hours per week**	Duration of SEA/Exam	**2 hours**
Formative Assessment Marks	**40**	Summative Assessment Marks	**60**

Course Outcomes (COs):

At the end of the course, students will be able to:

- Identify and describe different types of intelligent agents and their environments, and implement basic agent-based systems.
- Formulate problems and apply both uninformed and informed search strategies to solve them.
- Represent knowledge using propositional and first-order logic and perform inference using these representations.
- Understand and implement fundamental machine learning algorithms such as linear regression, decision trees, and neural networks.
- Demonstrate knowledge of advanced AI topics, including reinforcement learning, natural language processing, and robotics.
- Analyze and discuss the ethical considerations and societal impacts of AI technologies.

Unit	Description	Hours
1	**Introduction-**What is AI, the Foundations of Artificial Intelligence, Introduction to Artificial Intelligence, The History of AI, The State of the Art, Risks and Benefits of AI. Intelligent **Agents-**Agents and Environments, The Concept of Rationality, The Structure of Agents. **Problem-solving-Problem**-solving Agents, Search algorithms, Uninformed Search Strategies, Informed Search Strategies,	**11**

	Constraint Satisfaction Problems-Defining Constraint Satisfaction Problems, Backtracking Search for CSPs, Local Search for CSPs.	
2	**Knowledge, reasoning and planning-**Knowledge-Based Agents, Logic, Propositional Logic, Agents Based on Propositional Logic. **First-Order Logic (FOL)-**Representation Revisited, Syntax and Semantics of FOL, Using FOL, Knowledge Engineering in FOL. **Inference in FOL**-Propositional vs FOL, Unification and FOL, Forward Chaining, Backward Chaining. **Automated Planning**-Definition of Classical Planning, Algorithms for Classical Planning. Heuristics for Planning, Hierarchical Planning.	**11**
3	**Machine Learning: Leaning from Examples**- Forms of Leaning, supervised Learning, Learning Decision Trees, Model Selection and Optimization, Linear Regression and Classification, Developing ML Systems. **Learning Probabilistic Models**-Statistical Learning, Learning with Complete Data, Learning with Hidden Variables: The EM Algorithm. **Deep Learning-**Computation Graphs for Deep Learning, Convolutional Networks, Learning Algorithms, Recurrent Neural Networks, Unsupervised Learning and Transfer Learning.	**10**
4	**Reinforcement Learning**-Passive Reinforcement Learning, Active Reinforcement Learning, Apprenticeship and Inverse Reinforcement Learning, Applications of Reinforcement Learning **Natural Language Processing**-Language Models, Grammar. **Robotics**-Robots, Robot Hardware, Robotic Perception, Planning and Control, Planning Uncertain Movements, Reinforcement Learning in Robotics, Humans and Robots. **Philosophy, Ethics, and Safety of AI**-The Limits of AI, The Ethics of AI. **The Future of AI**-AI Components, AI Architectures	**10**

Text Book:

1. Russell, S., &Norvig, P. (2021). Artificial Intelligence: A Modern Approach (4th ed.). Pearson

Reference Books:

1. Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.
2. Murphy, K. P. (2012). Machine Learning: A Probabilistic Perspective. MIT Press.
3. Goodfellow, I., Bengio, Y., &Courville, A. (2016). Deep Learning. MIT Press.
4. Sutton, R. S., &Barto, A. G. (2018). Reinforcement Learning: An Introduction (2nd ed.). MIT Press.

Pedagogy: Lecture/ PPT/ Videos/ Animations/ Role Plays/ Think-Pair-Share/ Predict-Observe- Explain/ Demonstration/ Concept mapping/ Case Studies examples/ Tutorial/ Activity/ Flipped Classroom/ Jigsaw/ Field based Learning/ Project Based Learning/ Mini Projects/ Hobby Projects/ Forum Theatre/ Dance/ Problem Based Learning/ Game Based Learning/ Group Discussion/ Collaborative Learning/ Experiential Learning / Self Directed Learning etc

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Internet of Things(Theory)**		
Course Code:	**BCA-AIML-C18T**	No.of Credits	**03**
Contact hours	**3 Hours per week**	Duration of SEA/Exam	**2 hours**
Formative Assessment Marks	**40**	Summative Assessment Marks	**60**

Course Outcomes (COs):

At the end of the course, students will be able to:

- Explain the fundamental concepts, definitions, and characteristics of IoT, including its physical and logical design.
- Understand the different views of IoT architecture and be able to describe the functional, information, deployment, and operational aspects of IoT systems.
- Gain knowledge of various IoT protocols, including M2M, WSN, SCADA, and RFID protocols, and understand the importance of protocol standardization in IoT.
- Design and develop IoT devices using sensors, actuators, microcontrollers, and microprocessors, and manage resources effectively in IoT environments.
- Identify and explain the use of IoT in different application domains such as home automation, smart cities, healthcare, agriculture, and industrial IoT.
- Understand the security concerns and threats in IoT systems and be able to implement appropriate security measures to protect IoT data and devices.
- Gain hands-on experience in developing IoT solutions, conducting case studies, and applying theoretical knowledge to real-world scenarios

Unit	Description	Hours
1	**Introduction:** Concepts and definitions, Characteristics of IoT, Physical design of IoT, Logical design of IoT, Functional blocks of IoT **M2M:**Machine-to-Machine, Difference between IoT and M2M **IoT Systems:** Architecture, IoT levels and deployment templates, Examples of IoT systems **IoT Enabling Technologies:** Wireless Sensor Networks, Cloud Computing, Big Data Analytics, Communication Protocols **IoT and Cyber-Physical Systems:**Integration, Industrial IoT	**11**
2	**IoT Reference Architecture**: Introduction, Different architecture views, Functional view, Information view, Deployment and operational view **IoT Protocols**: Protocol standardization, M2M and WSN protocols, SCADA and RFID protocols, IoT data link layer & network layer protocols, Transport layer protocols, Session layer protocols **Networking Technologies**: PAN technologies, Zigbee, Bluetooth, BLE, LPWAN technologies, Cellular IoT	**11**
3	**IoT Devices**: IoT device design and architecture, Embedded systems, Sensors and actuators, - Microcontrollers, Microprocessors **Resource Management in IoT**: Resource allocation for IoT, Data management, Edge computing, Cloud computing for IoT, Fog computing **Hardware Platforms**:Arduino, Raspberry Pi, Other development boards	**10**
4	**IoT Applications**: Home automation, Smart cities, Environment monitoring, Energy management, Healthcare, Agriculture, Industrial IoT **IoT Security**:IoT security overview, Threats to IoT, Security requirements, Security architecture Cryptographic algorithms, IoT security lifecycle **IoT Case Studies**:Case study 1: Smart home, Case study 2: Smart city, Case study 3: Industrial IoT, Future trends in IoT	**10**

Text Book:

1. Bahga, A., &Madisetti, V. (2014). Internet of Things: A Hands-on Approach. VPT

Reference Books:

1. Buyya, R., &Dastjerdi, A. V. (2016). Internet of Things: Principles and Paradigms. Morgan Kaufmann.
2. McEwen, A., &Cassimally, H. (2014). Designing the Internet of Things. Wiley.
3. Minoli, D. (2013). Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications. Wiley.
4. Hersent, O., Boswarthick, D., &Elloumi, O. (2012). The Internet of Things: Key Applications and Protocols. Wiley.
5. Greengard, S. (2015). The Internet of Things. MIT Press

Pedagogy: Lecture/ PPT/ Videos/ Animations/ Role Plays/ Think-Pair-Share/ Predict-Observe- Explain/ Demonstration/ Concept mapping/ Case Studies examples/ Tutorial/ Activity/ Flipped Classroom/ Jigsaw/ Field based Learning/ Project Based Learning/ Mini Projects/ Hobby Projects/ Forum Theatre/ Dance/ Problem Based Learning/ Game Based Learning/ Group Discussion/ Collaborative Learning/ Experiential Learning / Self Directed Learning etc

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Artificial Intelligence-1 Lab**		
Course Code:	**BCA-AIML-C19P**	No.of Credits	**02**
Contact hours	**4 Hours per week**	Duration of SEA/Exam	**3 hours**
Formative Assessment Marks	**25**	Summative Assessment Marks	**25**

PART A

PART B

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	**PART-A** **Writing:5 Marks Execution:5Marks**	**8 Marks**
Program-2	**PART-B** **Writing:6 Marks Execution:6Marks**	**12Marks**
Practical Record		**05 Marks**
Total		**25Marks**

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Internet of Things Lab**		
Course Code:	**BCA-AIML-C20P**	No.of Credits	**02**
Contact hours	**4 Hours per Week**	Duration of SEA/Exam	**3 hours**
Formative Assessment Marks	**25**	Summative Assessment Marks	**25**

PART A

PART B

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	**PART-A** **Writing:5 Marks Execution:5Marks**	**8 Marks**
Program-2	**PART-B** **Writing:6 Marks Execution:6Marks**	**12Marks**
Practical Record		**05 Marks**
Total		**25Marks**