COEP Technological University

Department of Computer Science and Engineering

Curriculum Structure

Computer Science and Engineering
Honors and Minors

(Effective from: A.Y. 2026-27)

HONORS

Data Science

Sem	Course Type	Course Code	Course Name	L	T	P	S	Cr	Evaluation Scheme				
									Theory			Laboratory	
									MSE	TA	ESE	ISE	ESE
III	HON-01		Making Sense of Data	3	0	0	1	3	30	20	50	--	--
IV	HON-02		Big Data Analytics	3	0	2	1	4	30	20	50	50	50
V	HON-03		Data Mining	3	0	2	1	4	30	20	50	50	50
VI	HON-04		Reinforcement Learning	3	0	2	1	4	30	20	50	50	50
VII	HON-05		Explainable AI	3	0	0	1	3	30	20	50	--	--
			Total	**15**	**00**	**06**	**05**	**18**					

Cyber Security

Sem	Course Type	Course Code	Course Name	L	T	P	S	Cr	Evaluation Scheme				
									Theory			Laboratory	
									MSE	TA	ESE	ISE	ESE
III	HON-01		Fundamentals of Information and Coding Theory	3	0	0	1	3	30	20	50	--	--
IV	HON-02		Ethical Hacking	3	0	2	1	4	30	20	50	50	50
V	HON-03		Malware Analysis	3	0	2	1	4	30	20	50	50	50
VI	HON-04		IOT Security	3	0	2	1	4	30	20	50	50	50
VII	HON-05		Cyber Laws	3	0	0	1	3	30	20	50	--	--
			Total	**15**	**00**	**06**	**05**	**18**					

INTERDISCIPLINARY MINOR

Computer Engineering

Sem	Course Type	Course Code	Course Name	L	T	P	S	Cr	Evaluation Scheme				
									Theory			Laboratory	
									MSE	TA	ESE	ISE	ESE
III	IDP-01		Fundamentals of Software Engineering	3	0	0	1	3	30	20	50	--	--
IV	IDP-02		Fundamentals of Programming and Algorithms	3	0	2	1	4	30	20	50	50	50
V	IDP-03		Fundamentals of Computer Networks	3	0	2	1	4	30	20	50	50	50
VI	IDP-04		Fundamentals of System Programming	3	0	2	1	4	30	20	50	50	50
VII	IDP-05		Fundamentals of Cyber Security	3	0	0	1	3	30	20	50	--	--
			Total	**15**	**00**	**06**	**05**	**18**					

(HON) Making Sense of Data

Teaching Scheme
Lectures: 3 Hrs/ Week
Self-study: 1 Hr/Week

Evaluation Scheme
Theory: MSE: 30 Marks, TA: 20 Marks, ESE: 50 Marks

Course outcomes

1. Understand and identify different types of data, data distribution.
2. Analyze dataset using estimation methods, hypothesis testing, and analysis of variance to support statistical decision-making.
3. Demonstrate the use of data preprocessing methods and data analytic tools for effective data preparation and analysis.
4. Apply data visualization techniques to represent and interpret univariate, bivariate, and multivariate data effectively.
5. Demonstrate clustering and association rule mining techniques to discover patterns and relationships in data.

Course content

Introduction: **[6 Hrs]**

Overview, Sources of Data, Process for Making Sense of Data. Describing Data: Observations and Variables, Types of Variables, types of data, Data Distributions: Discrete Distributions such as Binomial, Poisson, Geometric etc.; Continuous Distributions such as Exponential, Normal etc.; Expectation: Moments; Central Limit theorem and its significance; Some sampling distributions like chi-square, t, F.

Statistical Inference: **[6 Hrs]**

Estimation - Introduction, classical methods of estimation, single sample: estimating the mean and variance, two samples: estimating the difference between two means and ratio of two variances; Tests of hypotheses - Introduction, testing a statistical hypothesis, tests on single sample and two samples concerning means and variances; ANOVA - One–way, Two–way with/without interactions.

Data Analytics Tools **[6 Hrs]**

Data analytics using Python: Statistical Procedures, NumPy, Pandas, SciPy, Matplotlib.

Data Pre-Processing **[6 Hrs]**

Understanding the Data, Dealing with Missing Values, Data Formatting, Data Normalization, Data Binning, Importing and Exporting data in Python, Turning categorical variables into quantitative variables in Python, Accessing Databases with Python.

Data Visualization **[6 Hrs]**

Graphic representation of data, Characteristics and charts for effective graphical displays, Chart types- Single var: Dot plot, Jitter plot, Error bar plot, Box-and-whisker plot, Histogram, Two variable: Bar chart, Scatter plot, Line plot, Log-log plot, More than two variables: Stacked plots, Parallel coordinate plot.

Identifying and Understanding Groups **[6 Hrs]**

Clustering: Agglomerative Hierarchical Clustering, k-Means Clustering, Association rules: Grouping by Combinations of Values, Extracting and Assessing Rules.

Self-study **[12 Hrs]**

Objectives of EDA, Sign Test, Wilcoxon Signed Rank Test, Seaborn Library, Feature extraction, Feature construction, Feature selection, Handling outliers, Violin Plot, Density Plot, Heat Map, Radar Chart,Silhouette Score, Elbow Method.

Textbooks

1. Glenn J. Myatt, Wayne P. Johnson, Making Sense of Data I: A Practical Guide to Exploratory Data Analysis and Data Mining, Wiley 2009.
2. Glenn J. Myatt, Wayne P. Johnson, Making Sense Of Data II A Practical Guide To Data Visualization, Advanced Data Mining Methods, And Applications,Wiley 2009.

Reference Books

1. H. Davenport, Jeanne G. Harris and Robert Morison, “Analytics at Work: Smarter Decisions, Better Results”, Harvard Business Press, 2010
2. Anil Maheshwari, “Data Analytics made accessible,” Amazon Digital Publication, 2014.

Web references

1. https://numpy.org
2. https://pandas.pydata.org
3. https://scipy.org

(HON) Big Data Analytics

TeachingScheme
Lectures:3 Hrs/Week
Laboratory: 2 Hrs/ Week
Self-Study: 1 Hrs/ Week

EvaluationScheme
Theory: MSE: 30Marks,TA: 20Marks,ESE: 50 Marks
Laboratory: CIE: 50 Marks, ESE: 50 Marks

Course outcomes

1. Identify , evaluate the need for Big Data and analyze its challenges,
2. Apply the Hadoop framework to process datasets and demonstrate use of ecosystem tools in distributed computing
3. Design and implement Big Data solutions using Spark, and assess performance in analyzing large datasets.
4. Develop and optimize data analysis using Apache Pig , Hive for large-scale datasets.

Course content

Introduction to Big Data [5Hrs]

Types of Digital Data, Overview of Big Data Analytics, Evolution of Big Data, Characteristics of Big Data - Volume, Variety, Velocity, Veracity, Valence, Value, Applications of Big Data, Challenges with Big data, Introduction to Enabling Technologies for Big Data.

Big Data Technology Stack [7Hrs]

Introduction to Big Data Technology Stack, Overview of Big Data distribution packages, Introduction to Big Data Platforms, Big Data Storage Platforms for Large Scale Data Storage, CAP Theorem, Eventual Consistency, Consistency Trade-Offs, ACID and BASE, Overview of Zookeeper and Paxos, Cassandra

Apache Hadoop [8Hrs]

History of Hadoop, Overview Apache Hadoop and Apache Spark, Hadoop Storage Framework – Hadoop Distributed File System (HDFS), HDFS Architecture, HDFS Concept, Hadoop Data Processing Framework – Map Reduce, Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution, Map Reduce Types and Formats, Map Reduce Features, MapReduce Programming Model with Spark.

Introduction to Big Data Streaming [7Hrs]

Big Data Pipelines for Real-Time computing, Introduction to Apache Spark Streaming, Kafka, Streaming Ecosystem, Spark Components, Resilient Distributed Dataset and data frames.

Apache PIG [7Hrs]

What is ETL, Introduction to Apache PIG, Execution Modes of PIG, Comparison of PIG with SQL and No-SQL Databases, PIG Data Types, Data Models in PIG, Grunt, PIG Latin, Overview of PIG User Defined Functions
Content

Apache HIVE [6Hrs]

Self-study **[12Hrs]**

Big Data Insights Using R and ML Algorithms: Regression Model, Clustering, Collaborative Filtering, Associate Rule Making, Decision Tree

Textbooks

1. Big Data Analytics, Seema Acharya, SubhasiniChellappan, Second Edition, 2019, Wiley India Pvt.Ltd, ISBN 978-81-2657-951-8.
2. Hadoop: The Definitive Guide, Tom White, Fourth Edition, 2015, O'Reilly, ISBN 978-93-5213-067-2.

Reference Books

1. Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, Bill Franks, First Edition,2012, John Wiley & sons, ISBN: 978-1-118-20878-6
2. Mining of Massive Datasets, Jure Leskovec, Anand Rajaraman and Jeffrey David Ulman, Second Edition 2016, Dreamtech Press, ISBN - 978-13-1663-849-1
3. Analytics in a Big Data World: The Essential Guide to Data Science and its Applications, Bart Baesens, First Edition 2014, Wiley, ISBN : 1118892704
4. Big Data Glossary, Pete Warden, First Edition 2011, O'Reily, ISBN
5. Harness the Power of Big Data The IBM Big Data Platform ",Paul Zikopoulos ,Dirk DeRoos , Krishnan Parasuraman , Thomas Deutsch , James Giles , David Corigan , Annotated Edition 2012, Tata McGraw Hill Publications, ISBN 978-0071808170
6. Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses, Michael Mineli, Michele Chambers, Ambiga Dhiraj, First Edition 2013, Wiley Publications, ISBN 978-1118147603
7. Big Data Analytics: Disruptive Technologies for Changing the Game, ArvindSathi, MC Press, 2012, ISBN 1583473807

Web references

1. IBM Skills Network – Big Data, Data Engineering and Data Science MicroMasters Programs
2. HKUST – Big Data Technology Course Resources
3. Introduction to Big Data: Aligning Strategy, Analytics and Operations (Coursera)
4. Mastering BigQuery (Starweaver)
5. Big Data Engineering Bootcamp with Hadoop, Spark, Kafka, GCP & Azure (Udemy)

(HON) Fundamentals of Information and Coding Theory

Teaching Scheme

Lectures: 3 Hrs/ Week

Self-Study: 1 Hrs/week

Evaluation Scheme

Theory: MSE: 30 Marks, TA: 20 Marks, ESE: 50 Marks

Course outcomes

Students will be able to:

1. Gainsubstantialknowledgeofinformationandentropy,andtheiruseininformation theory
2. Learn principles data compression
3. Understand techniques of design and performance evaluation of error correcting codes
4. Design and develop solutions for technical issues related to information coding
5. Get exposure to emerging topics in information theory, coding and compression.

Course content

Introduction to Information Theory **[8 Hrs]**

Introduction to Information Theory and Coding, Definition of Information Measure and Entropy, Information rate, Adjoint of an Information Source, Joint and Conditional Information Measure, Properties of Joint and Conditional Information Measures and A Morkov Source, Asymptotic Properties of Entropy and Problem Solving in Entropy.

Introduction to Coding **[8 Hrs]**

Classification of codes, Kraft-McMillan inequality, Source coding theorem, Shannon- Fano coding, Huffman coding, Mutual Information - Discrete memory less channels – BSC, BEC – Channel capacity, Shannon limit.

Data Compression **[7 Hrs]**

Adaptive Huffman Coding, Arithmetic Coding, LZW algorithm, Perceptual coding, Masking techniques, Channel Vocoder, Linear Predictive Coding, Video Compression and H.261.

Network Coding **[7 Hrs]**

The Buttery Network, Wireless and Satellite Communications, Source Separation, the Max-Flow Bound, Single-Source Linear Network Coding..

Error Control Coding: Block Codes **[6 Hrs]**

Definitions and Principles: Hamming weight, Hamming distance, Minimum distance decoding- Single parity codes, Hamming codes, Linear block codes, Cyclic codes – Syndrome calculation, Encoder and decoder – CRC

Error Control Coding: Convolutional Codes **[6 Hrs]**

Convolutional codes–code tree, trellis, state diagram-Encoding–Decoding: Sequential search and Viterbi algorithm.

Self-study **[12 Hrs]**

Extension of An Information Source and Markov Source, Extended Huffman coding, Psychoacoustic model, Acyclic Networks, Repetition codes, Principle of Turbo coding

Textbooks

1. T.M.CoverandJ.A.Thomas,“ElementsofInformationTheory”,John Wiley &Sons,second edition
2. RanjanBose,“InformationTheory,CodingandCryptography”,2E,Tata- McGraw Hill,second edition
3. MuralidharKulkarniandK.S.Shivaprakasha,“InformationTheoryand Coding”, WileyIndia Pvt Ltd

Reference Books

1. RaymondW.Yeung,"InformationTheoryandNetworkCoding",Springer,2008, ISBN: 978-0-387-79234-7, 978-0-387-79233-0, 978-1-4419-4630-0.

Web references

1. https://nptel.ac.in/courses/117101053
2. https://ocw.mit.edu/courses/6-441-information-theory-spring-2016/
3. https://onlinecourses.nptel.ac.in/e-learning/preview/noc20_ee94

(HON) Ethical Hacking

Teaching Scheme

Lectures: 3 Hrs/ Week

Laboratory: 2 Hrs/ Week

Self-Study: 1 Hrs/week

Evaluation Scheme

Theory: MSE: 30 Marks, TA: 20 Marks, ESE: 50 Marks

Laboratory: CIE: 100 Marks

Course outcomes

Students will be able to:

1. Identify legal and ethical issues related to vulnerability and penetration testing.
2. Report on the strengths and vulnerabilities of the tested network.
3. Exploit the vulnerabilities related to computer system and networks using state of the art tools and technologies
4. Execute a penetration test using standard hacking tools in an ethical manner
5. Evaluate best practices in security concepts to maintain confidentiality, integrity and availability of computer systems

Course content

Introduction to Hacking **[10 Hrs]**

Ethical hacking process, Hackers behaviour & mindset, Maintaining Anonymity, Hacking Methodology, Information Gathering, Active and Passive Sniffing, Physical security vulnerabilities and countermeasures. Internal and External testing. Preparation of Ethical Hacking and Penetration Test Reports and Documents.

Social Engineering and Attacks **[10 Hrs]**

Social Engineering attacks and countermeasures. Password attacks, Privilege Escalation and Executing Applications, Network Infrastructure Vulnerabilities, IP spoofing, DNS spoofing, Wireless Hacking: Wireless footprint, Wireless scanning and enumeration, Gaining access (hacking 802.11), WEP, WPA, WPA2.

Application Vulnerabilities and Attacks **[10 Hrs]**

DoS attacks. Web server and application vulnerabilities, SQL injection attacks, Vulnerability Analysis and Reverse Engineering, Buffer overflow attacks. Client-side browser exploits, Exploiting Windows Access Control Model for Local Elevation Privilege. Exploiting vulnerabilities in Mobile Application

Introduction to Metasploit **[10 Hrs]**

Metasploit framework, Metasploit Console, Payloads, Metrpreter, Introduction to Armitage, Installing and using Kali Linux Distribution, Introduction to penetration testing tools in Kali Linux. Case Studies of recent vulnerabilities and attacks

Self-study **[12 Hrs]**

Laws, Ethics, and Compliance in Ethical Hacking, Penetration Testing Methodologies Vulnerability Assessment vs. Penetration Testing, information gathering using publicly available sources, OSINT frameworks and methodologies

Textbooks

1. Ethical Hacking and Penetration Testing Guide", Baloch R, CRC Press, 2015.
2. "Hacking for Dummies " Beaver, K., Third edition John Wiley & Sons, 2013

Reference Books

1. "Network Forensics Tracking Hackers through Cyberspace ", Davidoff, S. and Ham, J., Prentice Hall, 2012.
2. "Computer Forensics JumpStart", Michael G. Solomon, K Rudolph, Ed Tittel, Broom N., and Barrett, D, Willey Publishing Inc, 2011

Web references

1. https://onlinecourses.nptel.ac.in/e-learning/preview/noc22_cs13
2. https://www.offsec.com/metasploit-unleashed/#metasploit-unleashed--free-ethical-hacking-course
3. https://tryhackme.com/

Laboratory assignments

Suggested List of Assignments:

1. Perform passive reconnaissance on a target organization using publicly available information.
2. Discover active hosts and services in a controlled network.
3. Identify vulnerabilities in a test system.
4. Evaluate password strength and security policies.
5. Identify common web application vulnerabilities in a deliberately vulnerable application.
6. Understand how improper input validation can affect database security.
7. Examine client-side input validation weaknesses.

(IDP) Fundamentals of Software Engineering

Teaching Scheme

Lectures: 3 Hrs/ Week

Self-Study: 1 Hrs/week

Evaluation Scheme

Theory: MSE: 30 Marks, TA: 20 Marks, ESE: 50 Marks

Course outcomes

Students will be able to:

1. Understand fundamental concepts of the System Development Lifecycle (SDLC) and illustrate their application in software engineering contexts
2. Construct user interface prototypes for real-world scenarios by applying appropriate analysis and design methods.
3. Formulate and assess procedures to assure product quality and maintainability before and after deployment
4. Demonstrate acquired knowledge to develop transferable skills across diverse industrial and commercial domains

Course content

Software Development process **[6 Hrs]**

Software Engineering basics, Software Crisis and Myths, Software Process and development, Software life cycle and Models, Analysis and comparison of various models, agile process

Requirement Engineering **[6 Hrs]**

Requirements Engineering, requirement engineering process, Introduction to Analysis model.

System Architecture and Design Overview **[8 Hrs]**

Architecture 4+1 view, architecture styles, Design process, quality concepts, Analysis model to Design Model transformation, Standardisation using UML.

Software Metrics **[8 Hrs]**

Introduction to Software Metrics, Size-oriented metrics and function point metrics. Effort and cost estimation techniques -LOC-based and Function-point based measures - The COCOMO model.

Testing and Maintenance **[6 Hrs]**

Validation and Verification activities, Testing Principles and strategies, Testing levels & types- White Box & Black Box Testing, Maintenance, Types of Maintenance.

Quality Assurance **[6 Hrs]**

Quality Standards Models Overview - ISO, Process improvement Models: CMM & CMMI. Quality Control and Quality Assurance.

Self-study **[12 Hrs]**

Laws, Ethics, and Compliance in Ethical Hacking, Penetration Testing Methodologies Vulnerability Assessment vs. Penetration Testing, information gathering using publicly available sources, OSINT frameworks and methodologies

Textbooks

1. “Pressman R., "Software Engineering, A Practitioners Approach", 6th Edition, Tata McGraw Hill Publication, 2004, ISBN 007-124083-124083-7.
2. “G. Booch, J. Rumbaugh and I. Jacobson. The Unified Modeling Language User Guide, Addison Wesley, 1999

Reference Books

1. Shari Pfleeger, “Software Engineering”, 2nd Edition. Pearson‟s Education, 2001.
2. Ian Sommerville, “Software Engineering”, 6th Edition, Addison-Wesley, 2000
3. Pankaj Jalote, “An Integrated Approach to Software Engineering”, Narosa publication house
4. Fred Brooks, “Mythical Manmonths”, www.cs.drexel.edu/~yfcai/CS4517/.

Web references

1. nptel.ac.in/courses/106105182
2. CS 682 Software Engineering, IITB

(IDP) Fundamentals of Programming and Algorithms

Teaching Scheme

Lectures: 3 Hrs/ Week

Laboratory: 2 Hrs/ Week

Self-Study: 1 Hrs/week

Evaluation Scheme

Theory: MSE: 30 Marks, TA: 20 Marks, ESE: 50 Marks

Laboratory: CIE: 100 marks

Course outcomes

0. Understand fundamental data structures and their applications.
1. Develop problem-solving and computational thinking skills.
2. Apply programming concepts to solve engineering problems.
3. Analyze algorithm efficiency using complexity measures.
4. Design solutions using appropriate algorithmic paradigms.

Course content

Programming Fundamentals and Problem Solving: [6 Hrs]

Problem-solving techniques and computational thinking, Algorithms and flowcharts, Program development cycle, Data types, variables, constants, Operators and expressions, Input/output operations, Conditional statements, Iterative constructs (for, while, do-while), Debugging and testing basics, Functions and modular programming Parameter passing techniques, Scope and lifetime of variables, Recursive functions, String handling and manipulation, File handling basics, Error and exception handling concepts

Fundamental Data Structures: [6 Hrs]

Abstract Data Types (ADT), Arrays and multidimensional arrays, Structures and pointers, Dynamic memory allocation, Singly linked lists, Doubly linked lists, Circular linked lists, Stacks and queues, Applications of stacks and queues

Algorithm Analysis and Design Fundamentals:

Characteristics of algorithms, Time and space complexity, Asymptotic notations: Big-O, Omega, Theta, Best, average and worst-case analysis, Recurrence relations, Brute force approach, Divide and conquer strategy

Searching and Sorting Algorithms: [6 Hrs]

Linear search, Binary search, Bubble sort, Selection sort, Insertion sort, Merge sort, Quick sort, Heap sort, Performance comparison of algorithms

Trees and Graphs [6 Hrs]

Binary trees and traversals, Binary Search Trees, Graph representations, Breadth First Search (BFS), Depth First Search (DFS), topological sort, Dijikstra's Single source shortest paths

Advanced Algorithmic Techniques: [8 Hrs]

Dynamic Programming: Introduction, All pairs shortest paths, 0-1 Knapsack, Traveling salesperson problem, Chained matrix multiplication, Longest common subsequence, Bellman-Ford algorithm. Back tracking: subset sum

problem, Hamiltonian cycle, Graph coloring

Self-study **[2 Hrs]**

Stack applications in expression evaluation and function calls, Growth of functions and algorithm efficiency, Algorithm visualization tools, Applications of trees in file systems and databases, Analysis of algorithms used in Google Maps navigation.

Textbooks

1. T. W. Pratt , "Programming Languages", 4th Edition ,Prentice-Hall Of India, ISBN 9780130287199.
2. Thomas Cormen, Charles Leiserson, Ronald Rivest and Cliford Stein, “Introduction to Algorithms”, PHI, 3rd edition, ISBN-13: 978-8120340077
3. E. Horowitz, S. Sahni, S.Anderson-freed, “Fundamentals of Data Structures in C”, Second Edition, University Press, ISBN 978-81-7371-605-8
4. B. Kernighan, D. Ritchie, “The C Programming Language”, Prentice Hall of India, Second Edition, ISBN 81-203-0596-5

Reference Books

- Ellis Horowitz, S. Sahni, D. Mehta “Fundamentals of Data Structures in C++”, Galgotia Book Source, New Delhi 1995 ISBN 16782928
- Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, “Fundamentals of Computer Algorithms”, Universities Press, 2nd edition (2008) , ISBN-13: 978-8173716126

Web references

1. https://nptel.ac.in/courses/106105164
2. https://nptel.ac.in/courses/106104128
3. https://nptel.ac.in/courses/106102064

Laboratory assignments

1. Recursive implementation of factorial, Fibonacci, and Tower of Hanoi.
2. String manipulation and pattern matching programs.
3. File handling operations: create, read, update, and append records.
4. Implementation of one-dimensional and two-dimensional array operations.
5. Dynamic memory allocation using pointers.
6. Implementation of singly linked list operations.
7. Implementation of doubly linked list and circular linked list operations.
8. Stack implementation-using arrays and linked lists.
9. Implement a queue (that is write queue.c and queue.h only) of characters, such that on an enqueue, the char is added at the end of queue, and on a dequeue the first element is taken out, but the queue uses only a 'head' pointer and not a 'tail pointer.
10. Write the following functions for a binary search tree implementation: Searches the maximum value in the tree, preorder traversal without using recursion, Search the string in the tree and returns a pointer to the node, print the binary tree so that it looks like a tree.
11. Implementation and analysis of linear, binary search algorithms and comparison of Bubble, Selection, and Insertion Sort.
12. Implementation and comparison of Merge Sort, Quick Sort, and Heap Sort.
13. Greedy algorithm implementation: Activity Selection and Fractional Knapsack.
14. Dynamic Programming: Fibonacci and Longest Common Subsequence, Traveling salesperson problem

Laboratory instructions or disclaimers if any

Guidelines for Laboratory /TW Assessment:

1. Continuous assessment of laboratory work is done based on overall performance & Laboratory performance of students.
2. Each Laboratory assignment assessment should assign grade/marks based on parameters with appropriate weightage.
3. Suggested parameters for overall assessment as well as each Laboratory assignment assessment include- timely completion, performance, innovation, efficiency, punctuality, and neatness.

Guidelines for Laboratory Conduction

1. Operating System recommended: - 64-bit Open-source Linux or its derivative
2. Programming tools recommended: - C, C++, Python