Program Structure for Second Year
Engineering Semester III & IV
UNIVERSITY OF MUMBAI
(With Effect from 2020-2021)

Semester IV

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Pract.	Tut.	Theory	Pract.	Tut.	Total
ITC401	Engineering Mathematics-IV	3	--	1	3	--	1	4
ITC402	Computer Network and Network Design	3	--	--	3	--	--	3
ITC403	Operating System	3	--	--	3	--	--	3
ITC404	Automata Theory	3	--	--	3	--	--	3
ITC405	Computer Organization and Architecture	3	--	--	3	--	--	3
ITL401	Network Lab	--	2	--	--	1	--	1
ITL402	Unix Lab	--	2	--	--	1	--	1
ITL403	Microprocessor Lab	--	2	--	--	1	--	1
ITL404	Python Lab (SBL)	--	4	--	--	2	--	2
ITM401	Mini Project – 1 B for Python based automation projects	--	4[$]	--	--	2	--	2
Total		**15**	**14**	**1**	**15**	**7**	**1**	**23**

Course Code	Course Name	Examination Scheme							
		Theory					Term Work	Pract/ oral	Total
		Internal Assessment			End Sem. Exam.	Exam. Duration (in Hrs)			
		Test 1	Test 2	Avg.					
ITC401	Engineering Mathematics-IV	20	20	20	80	3	25	--	125
ITC402	Computer Network and Network Design	20	20	20	80	3	--	--	100
ITC403	Operating System	20	20	20	80	3	--	--	100
ITC404	Automata Theory	20	20	20	80	3	--	--	100
ITC405	Computer Organization and Architecture	20	20	20	80	3	--	--	100
ITL401	Network Lab	--	--	--	--	--	25	25	50
ITL402	Unix Lab	--	--	--	--	--	25	25	50
ITL403	Microprocessor Lab	--	--	--	--	--	25	25	50
ITL404	Python Lab (SBL)	--	--	--	--	--	25	25	50
ITM401	Mini Project – 1 B for Python based automation projects	--	--	--	--	--	25	25	50
Total		**--**	**--**	**100**	**400**	**--**	**150**	**75**	**775**

$ indicates work load of Learner (Not Faculty), for Mini Project. Students can form groups with minimum 2 (Two) and not more than 4 (Four) Faculty Load : 1 hour per week per four groups

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Pract.	Tut.	Theory	TW/Pract	Tut.	Total
ITC401	Engineering Mathematics-IV	03	-	01	03	-	01	04

Course Code	Course Name	Examination Scheme							
		Theory				Term Work	Pract	Oral	Total
		Internal Assessment			End Sem Exam				
		Test1	Test2	Avg of Test 1 & 2					
ITC401	Engineering Mathematics-IV	20	20	20	80	25	-	-	125

Pre-requisite: Engineering Mathematics-I, Engineering Mathematics-II, Engineering Mathematics-III, Binomial Distribution.

Course Objectives:

Sr. No.	Course Objectives
The course aims:	
1	To study Matrix algebra and its application in engineering problems.
2	To learn Line and Contour integrals and expansion of complex valued function in a power series.
3	To study Z-Transforms and Inverse Z-Transforms with its properties.
4	To acquaint with the concepts of probability distributions and sampling theory for small samples.
5	To study and apply Linear and Non-linear programming Techniques to solve the optimization problems

Course Outcomes:

Sr. No.	Course Outcomes	Cognitive levels of attainment as per Bloom's Taxonomy
On successful completion, of course, learner/student will be able to:		
1	Apply the concepts of eigen values and eigen vectors to solve engineering problems.	L1, L2, L3
2	Illustrate the use of concepts of Complex Integration for evaluating integrals, computing residues & evaluate various contour integrals.	L3
3	Apply the concept of Z- transformation and its inverse in engineering problems.	L1,L2,L3

4	Apply the concept of probability distribution to engineering problems & testing hypothesis of small samples using sampling theory.	L3
5	Apply the concept of Linear Programming to solve the optimization problems	L1, L2, L3
6	Use the Non-Linear Programming techniques to solve the optimization problems.	L3

Module	Detailed Contents	Hours	CO Mapping
01	**Module: Linear Algebra (Theory of Matrices)** 1.1 Characteristic Equation, Eigenvalues and Eigenvectors and properties (without proof) 1.2 Cayley-Hamilton Theorem (without proof), verification and reduction of higher degree polynomials 1.3 Similarity of matrices, diagonalizable and non-diagonalizable matrices **Self-learning Topics:** Derogatory and non-derogatory matrices, Functions of Square Matrix, Linear Transformations, Quadratic forms.	7	CO1
02	**Module: Complex Integration** 2.1 Line Integral, Cauchy's Integral theorem for simple connected and multiply connected regions (without proof), Cauchy's Integral formula (without proof). 2.2 Taylor's and Laurent's series (without proof). 2.3 Definition of Singularity, Zeroes, poles of *f(z)*, Residues, Cauchy's Residue Theorem (without proof) **Self-learning Topics:** Application of Residue Theorem to evaluate real integrations.	7	CO2
03	**Module: Z Transform** 3.1 Definition and Region of Convergence, Transform of Standard Functions: $\{k^n a^k\}, \{a^{\lvert k\rvert}\}, \{^{k+n}_{n}C.\, a^k\}, \{c^k \sin(\alpha k+\beta)\}, \{c^k \sinh \alpha k\}, \{c^k \cosh \alpha k\}$. 3.2 Properties of Z Transform: Change of Scale, Shifting Property, Multiplication, and Division by k, Convolution theorem. 3.3 Inverse Z transform: Partial Fraction Method, Convolution Method. **Self-learning Topics:** Initial value theorem, Final value theorem, Inverse of Z Transform by Binomial Expansion	5	CO3
04	**Module: Probability Distribution and Sampling Theory** 4.1 Probability Distribution: Poisson and Normal distribution 4.2 Sampling distribution, Test of Hypothesis, Level of Significance, Critical region, One-tailed, and two-tailed test, Degree of freedom. 4.3 Students' t-distribution (Small sample). Test the significance of mean and Difference between the means of two samples. Chi-Square Test: Test of goodness of fit and independence of attributes, Contingency table. **Self-learning Topics:** Test significance for Large samples, Estimate parameters of a population., Yate's Correction.	7	CO4
05	**Module: Linear Programming Problems**	6	

	5.1 Types of solutions, Standard and Canonical of LPP, Basic and Feasible solutions, slack variables, surplus variables, Simplex method. 5.2 Artificial variables, Big-M method (Method of penalty) 5.3 Duality, Dual of LPP and Dual Simplex Method **Self-learning Topics: Sensitivity** Analysis, Two-Phase Simplex Method, Revised Simplex Method		CO5
06	**Module: Nonlinear Programming Problems** 6.1 NLPP with one equality constraint (two or three variables) using the method of Lagrange's multipliers 6.2 NLPP with two equality constraints 6.3 NLPP with inequality constraint: Kuhn-Tucker conditions **Self-learning Topics:** Problems with two inequality constraints, Unconstrained optimization: One dimensional search method (Golden Search method, Newton's method). Gradient Search method	7	CO6

References:

1. Advanced Engineering Mathematics, Erwin Kreyszig, John Wiley & Sons.
2. Advanced Engineering Mathematics, R. K. Jain and S. R. K. Iyengar, Narosa.
3. Complex Variables and Applications, Brown and Churchill, McGraw-Hill education.
4. Probability, Statistics and Random Processes, T. Veerarajan, McGraw-Hill education.
5. Operations Research: An Introduction, Hamdy A Taha, Pearson.
6. Engineering Optimization: Theory and Practice, S.S Rao, Wiley-Blackwell.
7. Operations Research, Hira and Gupta, S. Chand Publication.

Online References:

Sr. No.	Website Name
1.	https://www.nptel.ac.in

Term Work:

General Instructions:

1. Students must be encouraged to write at least 6 class tutorials on entire syllabus.
2. A group of 4-6 students should be assigned a self-learning topic. Students should prepare a presentation/problem solving of 10-15 minutes. This should be considered as mini project in Engineering Mathematics. This project should be graded for 10 marks depending on the performance of the students.

The distribution of Term Work marks will be as follows –

1.	Attendance (Theory and Tutorial)	05 marks
2.	Class Tutorials on entire syllabus	10 marks
3.	Mini project	10 marks

Assessment:

Internal Assessment Test:

Assessment consists of two class tests of 20 marks each. The first class test (Internal Assessment I) is to be conducted when approx. 40% syllabus is completed and second class test (Internal Assessment II) when additional 35% syllabus is completed. Duration of each test shall be one hour.

End Semester Theory Examination:

1. Question paper will comprise of total 06 questions, each carrying 20 marks.
2. Total 04 questions need to be solved.
3. Question No: 01 will be compulsory and based on entire syllabus wherein 4 sub-questions of 5 marks each will be asked.
4. Remaining questions will be randomly selected from all the modules.
5. Weightage of each module will be proportional to number of respective lecture hours as mentioned in the syllabus.

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical /Oral	Tutorial	Total
ITC402	Computer Network and Network Design	03	--	--	03	--	--	03

Course Code	Course Name	Examination Scheme						
		Theory Marks				Term Work	Pract. /Oral	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg.				
ITC402	Computer Network and Network Design	20	20	20	80	--	--	100

Course Objectives:

Sr. No.	Course Objectives
The course aims:	
1	Understand the division of network functionalities into layers.
2	Understand the types of transmission media along with data link layer concepts, design issues and protocols
3	Analyze the strength and weaknesses of routing protocols and gain knowledge about IP addressing
4	Understand the data transportation, issues and related protocols for end to end delivery of data.
5	Understand the data presentation techniques used in presentation layer & client/server model in application layer protocols.
6	Design a network for an organization using networking concepts

Course Outcomes:

Sr. No.	Course Outcomes	Cognitive levels of attainment as per Bloom's Taxonomy
On successful completion, of course, learner/student will be able to:		
1	Describe the functionalities of each layer of the models and compare the Models.	L1
2	Categorize the types of transmission media and explain data link layer concepts, design issues and protocols.	L2, L3, L4
3	Analyze the routing protocols and assign IP address to networks.	L4
4	Explain the data transportation and session management issues and related protocols used for end to end delivery of data.	L1, L2
5	List the data presentation techniques and illustrate the client/server model in application layer protocols.	L1, L3
6	Use of networking concepts of IP address, Routing, and application services to design a network for an organization	L3

Prerequisite: PCOM

DETAILED SYLLABUS:

Sr. No.	Module	Detailed Content	Hours	CO Mapping
0	**Prerequisite**	Terminologies of communication	**02**	-
I	**Introduction to Computer Networks**	Uses Of Computer Networks, Network Hardware, Network Software, Protocol Layering, Reference Models: OSI, TCP/IP, Comparison of OSI & TCP/IP, Network Devices. **Self-learning Topics:** Identify the different devices used in Network connection. College campus	**03**	CO1
II	**Physical Layer & Data Link Layer**	**Physical layer:** Guided Media, Unguided Media, Wireless Transmission: Electromagnetic Spectrum. Switching: Circuit-Switched Networks, Packet Switching, Structure Of A Switch **DLL Design Issues** (Services, Framing, Error Control, Flow Control), Error Detection and Correction(Hamming Code,Parity, CRC, Checksum) , Elementary Data Link protocols : Stop and Wait, Sliding Window(Go Back N, Selective Repeat), Piggybacking, HDLC **Medium Access Protocols:** Random Access, Controlled Access, Channelization. Ethernet Protocol: Standard Ethernet, Fast Ethernet (100 Mbps), Gigabit Ethernet, 10-Gigabit Ethernet. **Self-learning Topics:** Differentiate link layer in IOT network and Normal Network.	**08**	CO2
III	**Network Layer**	Network Layer Services, Packet Switching, Network Layer Performance, IPv4 Addressing (classful and classless), Subnetting, Supernetting ,IPv4 Protocol, DHCP, Network Address Translation (NAT). **Routing algorithms**: Distance Vector Routing, Link state routing,Path Vector Routing. **Protocols** –RIP,OSPF,BGP. **Next Generation IP:** IPv6 Addressing,IPv6 Protocol, Transition fromIPV4 to IPV6 **Self-learning Topics:** Study difference between IPV4 and IPV6. Network Class A, B, C, D, E and subnet mask.	**08**	CO3

IV	**Transport Layer & Session Layer**	**Transport Layer:** Transport Layer Services, Connectionless & Connection-oriented Protocols, Transport Layer protocols: User Datagram Protocol: UDP Services, UDP Applications, Transmission Control Protocol: TCP Services, TCP Features, Segment, A TCP Connection, Windows in TCP, Flow Control, Error Control, TCP Congestion Control, TCP Timers. **Session Layer:** Session layer design issues, Session Layer protocol - Remote Procedure Call (RPC), **Self-learning Topics:** List real time example of UDP and TCP.	**07**	CO4
V	**Presentation Layer & Application Layer**	**Presentation layer** :Compression: Comparison between Lossy Compression and Lossless Compression, Huffman Coding, Speech Compression, LZW, RLE, Image Compression – GIF,JPEG. **Application layer:** Standard Client-Server Protocols: World Wide Web, HTTP, FTP, Electronic Mail, Domain Name System (DNS), SNMP **Self-learning Topics:** Difference between HTTP and FTP Protocol.	**05**	CO5
VI	**Network Design Concepts**	Introduction to VLAN ,VPN A case study to design a network for an organization meeting the following guidelines: Networking Devices, IP addressing: Subnetting, Supernetting, Routing Protocols to be used, Services to be used: TELNET, SSH, FTP server, Web server, File server, DHCP server and DNS server. **Self-learning Topics:** Study the Network Design of your college campus.	**06**	CO6

Text Books:

1. Andrew S Tanenbaum, Computer Networks -, 4th Edition, Pearson Education.
2. Behrouz A. Forouzan, Data Communications and Networking ,4th Edition,Mc Graw Hill education.

References:

1. S. Keshav, An Engineering Approach to Computer Networks, 2nd Edition, Pearson Education.
2.B. A. Forouzan, "TCP/IP Protocol Suite", Tata McGraw Hill edition, Third Edition.
3. Ranjan Bose, Information Theory, Coding and Cryptography, Ranjan Bose, Tata McGrawHill , Second Edition.
4. Khalid Sayood, Introduction to Data Compression, Third Edition, Morgan Kaufman.

Online References:

Sr. No.	Website Name
1.	https://www.nptel.ac.in
2.	https://swayam.gov.in
3.	https://www.coursera.org/

Assessment:
Internal Assessment (IA) for 20 marks:

- IA will consist of Two Compulsory Internal Assessment Tests. Approximately 40% to 50% of syllabus content must be covered in First IA Test and remaining 40% to 50% of syllabus content must be covered in Second IA Test

➢ **Question paper format**

- Question Paper will comprise of a total of **six questions each carrying 20 marks Q.1** will be **compulsory** and should **cover maximum contents of the syllabus**
- **Remaining questions** will be **mixed in nature** (part (a) and part (b) of each question must be from different modules. For example, if Q.2 has part (a) from Module 3 then part (b) must be from any other Module randomly selected from all the modules)
- A total of **four questions** need to be answered

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical /Oral	Tutorial	Total
ITC403	Operating System	03	--	--	03	--	--	03

Course Code	Course Name	Examination Scheme						
		Theory Marks				Term Work	Pract. /Oral	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg.				
ITC403	Operating System	20	20	20	80	--	--	100

Course Objectives:

Sr. No.	Course Objectives
The course aims:	
1	To understand the major components of Operating System &its functions.
2	To introduce the concept of a process and its management like transition, scheduling, etc.
3	To understand basic concepts related to Inter-process Communication (IPC) like mutual exclusion, deadlock, etc. and role of an Operating System in IPC.
4	To understand the concepts and implementation of memory management policies and virtual memory.
5	To understand functions of Operating System for storage management and device management.
6	To study the need and fundamentalsof special-purpose operating system with the advent of new emerging technologies.

Course Outcomes:

Sr. No.	Course Outcomes	Cognitive levels of attainment as per Bloom's Taxonomy
On successful completion, of course, learner/student will be able to:		
1	Understand the basic concepts related to Operating System.	L1, L2
2	Describe the process management policies and illustrate scheduling of processes by CPU.	L1
3	Explain and apply synchronization primitives and evaluate deadlock conditionsas handled by Operating System.	L2
4	Describe and analyze the memory allocation and management functions of Operating System.	L1
5	Analyze and evaluate the services provided by Operating System for storage management.	L4, L5
6	Compare the functions of various special-purpose Operating Systems.	L2

Prerequisite: Programming Language C

DETAILED SYLLABUS:

Sr. No.	Module	Detailed Content	Hours	CO Mapping
0	Prerequisite	Programming Language C; Basic of Hardware i.e. ALU, RAM, ROM, HDD, etc.; Computer-System Organization.	**02**	-
I	Fundamentals of Operating System	Introduction to Operating Systems; Operating System Structure and Operations; Functions of Operating Systems; Operating System Services and Interface; System Calls and its Types; System Programs; Operating System Structure; System Boot. **Self-learning Topics:** Study of any three different OS. System calls with examples for different OS.	**03**	CO1
II	Process Management	Basic Concepts of Process; Operation on Process; Process State Model and Transition; Process Control Block; Context Switching; Introduction to Threads; Types of Threads, Thread Models; Basic Concepts of Scheduling; Types of Schedulers; Scheduling Criteria; Scheduling Algorithms. **Self-learning Topics:** Performance comparison of Scheduling Algorithms, Selection of Scheduling Algorithms for different situations, Real-time Scheduling	**06**	CO2
III	ProcessCoordination	Basic Concepts of Inter-process Communication and Synchronization; Race Condition; Critical Region and Problem; Peterson's Solution; Synchronization Hardware and Semaphores; Classic Problems of Synchronization; Message Passing; Introduction to Deadlocks; System Model, Deadlock Characterization; Deadlock Detection and Recovery; Deadlock Prevention; Deadlock Avoidance. **Self-learning Topics:** Study a real time case study for Deadlock detection and recovery.	**09**	CO3
IV	Memory Management	Basic Concepts of Memory Management; Swapping; Contiguous Memory Allocation; Paging; Structure of Page Table; Segmentation; Basic Concepts of Virtual Memory; Demand Paging, Copy-on Write; Page Replacement Algorithms; Thrashing. **Self-learning Topics:** Memory Management for any one Operating System, Implementation of Page Replacement Algorithms.	**09**	CO4

V	Storage Management	Basic Concepts of File System; File Access Methods; Directory Structure; File-System Implementation; Allocation Methods; Free Space Management; Overview of Mass-Storage Structure; Disk Structure; Disk Scheduling; RAID Structure; Introduction to I/O Systems. **Self-learning Topics:** File System for Linux and Windows, Features of I/O facility for different OS.	**06**	CO5
VI	Special-purpose Operating Systems	Open-source and Proprietary Operating System; Fundamentals of Distributed Operating System; Network Operating System;Embedded Operating Systems;Cloud and IoT Operating Systems; Real-Time Operating System;Mobile Operating System; Multimedia Operating System;Comparison between Functions of various Special-purpose Operating Systems. **Self-learning Topics:** Case Study on any one Special-purpose Operating Systems.	**04**	CO6

Text Books:

1. A. Silberschatz, P. Galvin, G. Gagne, Operating System Concepts, 10th ed., Wiley, 2018.
2. W. Stallings, Operating Systems: Internal and Design Principles, 9th ed., Pearson, 2018.
3. A. Tanenbaum, Modern Operating Systems, Pearson, 4th ed., 2015.

Reference Books:

1. N. Chauhan, Principles of Operating Systems, 1st ed., Oxford University Press, 2014.
2. A. Tanenbaum and A. Woodhull, Operating System Design and Implementation, 3rd ed., Pearson.
3. R. Arpaci-Dusseau and A. Arpaci-Dusseau, Operating Systems: Three Easy Pieces, CreateSpace Independent Publishing Platform, 1st ed., 2018.

Online References:

Sr. No.	Website Name
1.	https://www.nptel.ac.in
2.	https://swayam.gov.in
3.	https://www.coursera.org/

Assessment:

Internal Assessment (IA) for 20 marks:

- IA will consist of Two Compulsory Internal Assessment Tests. Approximately 40% to 50% of syllabus content must be covered in First IA Test and remaining 40% to 50% of syllabus content must be covered in Second IA Test

- **Question paper format**

  - Question Paper will comprise of a total of **six questions each carrying 20 marks Q.1** will be **compulsory** and should **cover maximum contents of the syllabus**

  - **Remaining questions** will be **mixed in nature** (part (a) and part (b) of each question must be from different modules. For example, if Q.2 has part (a) from Module 3 then part (b) must be from any other Module randomly selected from all the modules)

  - A total of **four questions** need to be answered

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical /Oral	Tutorial	Total
ITC404	Automata Theory	03	--	--	03	--	--	03

Course Code	Course Name	Examination Scheme						
		Theory Marks				Term Work	Pract. /Oral	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg.				
ITC404	Automata Theory	20	20	20	80	--	--	100

Course Objectives:

Sr. No.	Course Objectives
The course aims:	
1	To learn fundamentals of Regular and Context Free Grammars and Languages.
2	To understand the relation between Regular Language and Finite Automata and machines.
3	To learn how to design Automata's as Acceptors, Verifiers and Translators.
4	To understand the relation between Regular Languages, Contexts free Languages, PDA and TM.
5	To learn how to design PDA as acceptor and TM as Calculators.
6	To learn applications of Automata Theory.

Course Outcomes:

Sr. No.	Course Outcomes	Cognitive levels of attainment as per Bloom's Taxonomy
On successful completion, of course, learner/student will be able to:		
1	Explain, analyze and design Regular languages, Expression and Grammars.	L2, L4, L6
2	Design different types of Finite Automata and Machines as Acceptor, Verifier and Translator.	L6
3	Analyze and design Context Free languages and Grammars.	L4, L6
4	Design different types of Push down Automata as Simple Parser.	L6
5	Design different types of Turing Machines as Acceptor, Verifier, Translator and Basic computing machine.	L6
6	Develop understanding of applications of various Automata.	L6

Prerequisite: Basic Mathematical Fundamentals: Sets, Logic, Relations, Functions.

DETAILED SYLLABUS:

Sr. No.	Module	Detailed Content	Hours	CO Mapping

0	Prerequisite	Basic Mathematical Fundamentals: Sets, Logic, Relations, Functions.	**02**	-
I	Introduction and Regular Languages	Languages: Alphabets and Strings. Regular Languages: Regular Expressions, Regular Languages, Regular Grammars, RL and LL grammars, Closure properties **Self-learning Topics:** Practice exercise on Regular Expressions. Identify the tools also.	**05**	CO1
II	Finite Automata	Finite Automata: FA as language acceptor or verifier, NFA (with and without ε) , DFA, RE to NFA, NFA to DFA, Reduced DFA , NFA-DFA equivalence, FA to RE. Finite State Machines with output : Moore and Mealy machines. Moore and Mealy M/C conversion. Limitations of FA. **Self-learning Topics:** Practice exercise on FA and NFA	**09**	CO2
III	Context Free Grammars	Context Free Languages: CFG, Leftmost and Rightmost derivations, Ambiguity, Simplification and Normalization (CNF & GNF) and Chomsky Hierarchy (Types 0 to 3) **Self-learning Topics:** Practice numerical or exercise on CFG	**08**	CO3
IV	Push Down Automata	Push Down Automata: Deterministic (single stack) PDA, Equivalence between PDA and CFG. Power and Limitations of PDA. **Self-learning Topics:** List the examples of PDA.	**05**	CO4
V	Turing Machine	Turing Machine: Deterministic TM, Variants of TM, Halting problem, Power of TM. **Self-learning Topics:** Practice numerical of TM.	**07**	CO5
VI	Applications of Automata	Applications of FA, CFG, PDA & TM. Introduction to Compiler & Its phases. **Self-learning Topics:** Case study on any one compiler.	**03**	CO2,CO 3, CO4,CO 5, CO6

Text books

1. J.C.Martin, “Introduction to languages and the Theory of Computation”, TMH.
2. Kavi Mahesh, “Theory of Computation A Problem Solving Approach”, Wiley India
3. A. V. Aho, R. Shethi, Monica Lam , J.D. Ulman , “Compilers Principles, Techniques and Tools “,Pearson Education.

References

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, “Introduction to Automata Theory, Languages and Computation”, Pearson Education.
2. Daniel I.A. Cohen, “Introduction to Computer Theory”, John Wiley & Sons.
3. Vivek Kulkarni,” Theory of Computation”, Oxford University.
4. N.Chandrashekhar, K.L.P. Mishra, “Theory of Computer Science, Automata Languages & Computations”, PHI publications.
5.J. J. Donovan, “ Systems Programming”, TMH.

Online References:

Sr. No.	Website Name
1.	https://www.nptel.ac.in
2.	https://online.stanford.edu
3.	https://www.coursera.org/

Assessment:

Internal Assessment (IA) for 20 marks:

- IA will consist of Two Compulsory Internal Assessment Tests. Approximately 40% to 50% of syllabus content must be covered in First IA Test and remaining 40% to 50% of syllabus content must be covered in Second IA Test

➢ **Question paper format**

- Question Paper will comprise of a total of **six questions each carrying 20 marks Q.1** will be **compulsory** and should **cover maximum contents of the syllabus**
- **Remaining questions** will be **mixed in nature** (part (a) and part (b) of each question must be from different modules. For example, if Q.2 has part (a) from Module 3 then part (b) must be from any other Module randomly selected from all the modules)
- A total of **four questions** need to be answered

.

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical /Oral	Tutorial	Total
ITC405	Computer Organization and Architecture	03	--	--	03	--	--	03

Course Code	Course Name	Examination Scheme						
		Theory Marks				Term Work	Pract. /Oral	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg.				
ITC405	Computer Organization and Architecture	20	20	20	80	--	--	100

Course Objectives:

Sr. No.	Course Objectives
The course aims:	
1	Learn the fundamentals of Digital Logic Design.
2	Conceptualize the basics of organizational and features of a digital computer.
3	Study microprocessor architecture and assembly language programming.
4	Study processor organization and parameters influencing performance of a processor.
5	Analyse various algorithms used for arithmetic operations.
6	Study the function of each element of memory hierarchy and various data transfer techniques used in digital computer.

Course Outcomes:

Sr. No.	Course Outcomes	Cognitive levels of attainment as per Bloom's Taxonomy
On successful completion, of course, learner/student will be able to:		
1	Demonstrate the fundamentals of Digital Logic Design	L1, L2
2	Describe basic organization of computer, the architecture of 8086 microprocessor and implement assembly language programming for 8086 microprocessors.	L1
3	Demonstrate control unit operations and conceptualize instruction level parallelism.	L1, L2
4	List and Identify integers and real numbers and perform computer arithmetic operations on integers.	L1,L4
5	Categorize memory organization and explain the function of each element of a memory hierarchy.	L4
6	Examine different methods for computer I/O mechanism.	L3

Prerequisite: Basics of Electrical Engineering, Fundamentals of Computer.

DETAILED SYLLABUS:

Sr. No.	Module	Detailed Content	Hours	CO Mapping
0	**Prerequisite**	Basics of Electrical Engineering, Fundamentals of Computer	**02**	
I	**Fundamentals of Logic Design**	Number systems: Introduction to Number systems, Binary Number systems, Signed Binary Numbers, Binary, Octal, Decimal and Hexadecimal number and their conversions, 1's and 2's complement Combinational Circuits: NOT,AND,OR,NAND,NOR,EX-OR,EX-NOR Gates. Half & Full Adder and subtractor, Reduction of Boolean functions using K-map method (2,3,4 Variable), introduction to Multiplexers and Demultiplexers, Encoders & Decoders. Sequential Circuits: Introduction to Flip Flops: SR, JK, D, T, master slave flip flop, Truth Table. **Self-learning Topics:** Number System, Quine-McCluskey,Flip-Flop conversion, Counter Design.	**07**	CO1
II	**Overview of Computer Architecture & Organization**	Introduction of Computer Organization and Architecture. Basic organization of computer and block level description of the functional units. Evolution of Computers, Von Neumann model. Performance measure of Computer Architecture, Amdahl's Law Architecture of 8086 Family, Instruction Set, Addressing Modes, Assembler Directives, Mixed-Language Programming, Stack, Procedure, Macro. **Self-learning Topics:** Interfacing of I/O devices with 8086(8255,ADC,DAC).	**08**	CO2
III	**Processor Organization and Architecture**	CPU Architecture, Instruction formats, basic instruction cycle with Interrupt processing. Instruction interpretation and sequencing. Control Unit: Soft wired (Microprogrammed) and hardwired control unit design methods. Microinstruction sequencing and execution. Micro operations, concepts of nano programming. Introduction to parallel processing concepts, Flynn's classifications, instruction pipelining, pipeline hazards. **Self-learning Topics**: Study the examples on instruction pipelining for practice.	**07**	CO3
IV	**Data Representation and Arithmetic Algorithms**	Booth's algorithm. Division of integers: Restoring and non-restoring division, signed division, basics of floating-point representation IEEE 754 floating point (Single & double precision) number representation. **Self-learning Topics:** Implement Booth's Algorithm and Division methods.	**04**	CO4
V	**Memory Organization**	Introduction to Memory and Memory parameters. Classifications of primary and secondary memories. Types of RAM and ROM, Allocation policies, Memory hierarchy and characteristics. Cache memory: Concept, architecture (L1, L2, L3), mapping techniques. Cache Coherency, Interleaved and Associative memory	**07**	CO5

		Self-learning Topics: Case study on Memory Organization, Numerical on finding EAT, Address mapping.		
VI	**I/O Organization**	Input/output systems, I/O module-need & functions and Types of data transfer techniques: Programmed I/O, Interrupt driven I/O and DMA **Self-learning Topics:** Comparison of all I/O methods.	**04**	CO6

Text Books:

1. R. P. Jain,"Modern Digital Electronics", TMH
2. M. Morris Mano,"Digital Logic and Computer Design", PHI
3. Carl Hamacher, Zvonko Vranesic and Safwat Zaky, Computer Organization, Fifth Edition, Tata McGraw-Hill.
4. William Stallings, Computer Organization and Architecture: Designing for Performance, Eighth Edition,, Pearson
5. John Uffenbeck, 8086/8088 family: Design Programming and Interfacing, (Pearson Education

References:

1. A. Anand Kumar, "Fundamentals of Digital Circuits",. PHI
2. Donald P Leach, Albert Paul Malvino, "Digital Principals & Applications", TMH.
3. B. Govindarajulu,, Computer Architecture and Organization: Design Principles and Applications, Computer Architecture and Organization: Design Principles and Applications, Tata McGraw-Hill
4. Dr. M. Usha, T. S. Srikanth, Computer System Architecture and Organization, First Edition, Wiley-India.
5. John P. Hayes, Computer Architecture and Organization, Third Edition., McGraw-Hill
6. K Bhurchandi, Advanced Microprocessors & Peripherals, Tata McGraw-Hill Education

Online References:

Sr. No.	Website Name
1.	https://www.nptel.ac.in
2.	https://www.geeksforgeeks.org
3.	https://www.coursera.org/

Assessment:

Internal Assessment (IA) for 20 marks:

- IA will consist of Two Compulsory Internal Assessment Tests. Approximately 40% to 50% of syllabus content must be covered in First IA Test and remaining 40% to 50% of syllabus content must be covered in Second IA Test

➢ **Question paper format**

- Question Paper will comprise of a total of **six questions each carrying 20 marks Q.1** will be **compulsory** and should **cover maximum contents of the syllabus**
- **Remaining questions** will be **mixed in nature** (part (a) and part (b) of each question must be from different modules. For example, if Q.2 has part (a) from Module 3 then part (b) must be from any other Module randomly selected from all the modules)
- A total of **four questions** need to be answered

Lab Code	Lab Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITL401	Network Lab	--	02	--	--	01	--	01

Lab Code	Lab Name	Examination Scheme						
		Theory Marks				Term Work	Pract. /Oral	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg.				
ITL401	Network Lab	--	--	--	--	25	25	50

Lab Objectives:

Sr. No.	Lab Objectives
The Lab experiments aims:	
1	To get familiar with the basic network administration commands
2	To install and configure network simulator and learn basics of TCL scripting.
3	To understand the network simulator environment and visualize a network topology and observe its performance
4	To implement client-server socket programs.
5	To observe and study the traffic flow and the contents of protocol frames.
6	To design and configure a network for an organization

Lab Outcomes:

Sr. No.	Lab Outcomes	Cognitive levels of attainment as per Bloom's Taxonomy
On successful completion, of course, learner/student will be able to:		
1	Execute and evaluate network administration commands and demonstrate their use in different network scenarios	L3, L5
2	Demonstrate the installation and configuration of network simulator.	L1, L2
3	Demonstrate and measure different network scenarios and their performance behavior.	L1, L2
4	Implement the socket programming for client server architecture.	L3
5	Analyze the traffic flow of different protocols	L4
6	Design a network for an organization using a network design tool	L6

Prerequisite: C /Java

Hardware & Software Requirements:

Hardware Requirement:	**Software requirement:**
PC i3 processor and above	NS2.34, Protocol Analyzer (eg. Wireshark), C/Java/python

DETAILED SYLLABUS:

Sr. No.	Module	Detailed Content	Hours	LO Mapping
0	**Prerequisite**	Programming Language (C/Java), Basic commands of windows and Unix/Linux operating system. editor commands (eg nano/vi editor etc)	**02**	-
I	**Fundamentals of Computer Network**	Understanding Basic networking Commands: ifconfig ,ip, traceroute, tracepath, ping, netstat, ss, dig, nslookup, route, host, arp, hostname, curl or wget, mtr, whois, tcpdump • Execute and analyze basic networking commands.	**02**	LO1
II	**Basics of Network simulation**	Installation and configuration of NS2. Introduction to Tcl Hello Programming • Installation and configuring of NS-2 simulator and introduction to Tcl using Hello program	**02**	LO2
III	**Simulation of Network Topology with different Protocols**	Implementation of Specific Network topology with respect to 1. Number of nodes and physical layer configuration 2. Graphical simulation ofnetwork with RoutingProtocols(Distance Vector/ Link State Routing) and trafficconsideration (TCP, UDP)using NAM. 3. Analysis of networkperformance for quality ofservice parameters such aspacket-delivery-ratio, delayand throughput 4. Comparative analysis of routing protocols with respect to QOS parametersusing Xgraph/gnuplot fordifferent load conditions. • Write TCL scripts to create topologies. Create and run traffics and analyze the result using NS2 • Write TCL scripts for topology with Graphical simulation of traffic consideration (TCP, UDP) using NAM and plot the graph • Implement distance vector and link state routing protocols in NS2.	**06**	LO3 LO5
IV	**Socket Programming**	Socket Programming with C/Java/python 1. TCP Client, TCP Server 2. UDP Client, UDP Server • To study and Implement Socket Programming using TCP.	**04**	LO4

		• To study and Implement Socket Programming using UDP		
V	**Protocol Analyzer**	1. Study of various Network Protocol Analyzer Tools likeWireshark, tcpdump, Windump, Microsoft Message Analyzer, Ettercap, Nirsoft SmartSniff etc. 2. Install one of the Network protocol analyzer tools and analyze the traffic • Study various network protocol analyzer tools and analyze the network traffics using one of the network protocol analyzer tools.	**04**	LO5
VI	**Network Design**	Network Design for an organization using the following concepts: 1. Addressing (IP Address Assignment), 2. Naming (DNS) 3. Routing • Perform remote login using Telnet Server • Design a network for an organization using the concepts of Addressing (IP Address Assignment), Naming (DNS) and Routing. Also mention the internetworking devices used	**06**	LO6

Text Books:

1. Computer Network Simulation in NS2 Basic Concepts and Protocol Implementation.-Prof Neeraj Bhargava,Pramod Singh Rathore,Dr.Ritu Bhargava,Dr.Abhishek Kumar, First Edition.BPB Publication.
2. Packet analysis with Wire shark, Anish Nath, PACKT publishing
3. TCP/IP Protocol Suite 4th Edition by Behrouz A. Forouzan

References:

1. NS2.34 Manual
2. Practical Packet Analysis: Using Wireshark to Solve Real-World Network Problems by Chris Sanders

Term Work: Term Work shall consist of at least 10 to 12 practical's based on the above list. Also Term work Journal must include at least 2 assignments.

Term Work Marks: 25 Marks (Total marks) = 15 Marks (Experiment) + 5 Marks (Assignments) + 5 Marks (Attendance)

Practical & Oral Exam: An Oral & Practical exam will be held based on the above syllabus.

Lab Code	Lab Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITL402	Unix Lab	--	02	--	--	01	--	01

Lab Code	Lab Name	Examination Scheme						
		Theory Marks				Term Work	Pract. /Oral	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg.				
ITL402	Unix Lab	--	--	--	--	25	25	50

Lab Objectives:

Sr. No.	Lab Objectives
The Lab experiments aims:	
1	To understand architecture and installation of Unix Operating System
2	To learn Unix general purpose commands and programming in Unix editor environment
3	To understand file system management and user management commands in Unix.
4	To understand process management and memory management commands in Unix
5	To learn basic shell scripting.
6	To learn scripting using awk and perl languages.

Lab Outcomes:

Sr. No.	Lab Outcomes	Cognitive levels of attainment as per Bloom's Taxonomy
On successful completion, of course, learner/student will be able to:		
1	Understand the architecture and functioning of Unix	L1, L2
2	Identify the Unix general purpose commands	L4
3	Apply Unix commands for system administrative tasks such as file system management and user management.	L3
4	Execute Unix commands for system administrative tasks such as process management and memory management	L4
5	Implement basic shell scripts for different applications.	L3
6	Implement advanced scripts using awk & perl languages and grep, sed, etc. commandsfor performing various tasks.	L3

Prerequisite: Programming Language C

Hardware & Software Requirements:

Hardware Requirement:	**Software requirement:**
PC i3 processor and above	Unix, Editor, Bash shell, Bourne shell and C shell

DETAILED SYLLABUS:

Sr. No.	**Module**	**Detailed Content**	**Hours**	**LO Mapping**
0	Prerequisite	Basic Programming Skills, Concepts of Operating System	**02**	-
I	Introduction to Unix	Case Study: Brief History of UNIX, Unix Architecture; Installation of Unix Operating System	**03**	LO1
II	Basic Commands	a) Execution of Unix General Purpose Utility Commands like echo, clear, exit, date, time, uptime, cal, cat, tty, man, which, history, id, pwd, whoami, ping, ifconfig, pr, lp, lpr, lpstat, lpq, lprm, cancel, mail, etc. b) Working with Editor Vi/other editor.	**03**	LO2
III	Commands for File System Management and User Management	a) Study of Unix file system (tree structure), file and directory permissions, single and multiuser environment. b) Execution of File System Management Commands like ls, cd, pwd, cat, mkdir, rmdir, rm, cp, mv, chmod, wc, piping and redirection, grep, tr, echo, sort, head, tail, diff, comm, less, more, file, type, wc, split, cmp, tar, find, vim, gzip, bzip2, unzip, locate, etc. c) Execution of User Management Commands like who, whoami, su, sudo, login, logout, exit, passwd, useradd/adduser, usermod, userdel, groupadd, groupmod, groupdel, gpasswd, chown, chage, chgrp, chfn, etc.	**04**	LO3
IV	Commands for Process Management and Memory Management	a) Execution of Process Management Commands like ps, pstree, nice, kill, pkill, killall, xkill, fg, bg, pgrep, renice, etc. b) Execution of Memory Management Commands like free, /proc/meminfo, top, htop, df, du, vmstat, demidecode, sar, pagesize, etc.	**04**	LO4
V	Basic Scripts	a) Study of Shell, Types of Shell, Variables andOperators b) Execute the following Scripts (at least 6): (i) Write a shell script to perform arithmetic operations. (ii) Write a shell script to calculate simple interest. (iii) Write a shell script to determine largest among three integer numbers. (iv) Write a shell script to determine a given year is leap year or not. (v) Write a shell script to print multiplication table of given number using while statement.	**04**	L02, L03, L05

		(vi) Write a shell script to search whether element is present is in the list or not. (vii) Write a shell script to compare two strings. (viii) Write a shell script to read and check if the directory / file exists or not, if not make the directory / file. (ix) Write a shell script to implement menu-driven calculator using case statement. (x) Write a shell script to print following pattern: * * * * * * * * * * (xi) Write a shell script to perform operations on directory like: display name of current directory; display list of directory contents; create another directory, write contents on that and copy it to a suitable location in your home directory; etc.		
VI	Advanced Scripts	a) Execute the following scripts using grep / sed commands: (i) Write a script using grep command to find the number of words character, words and lines in a file. (ii) Write ascriptusing egrep command to display list of specific type of files in the directory. (iii) Write a script using sed command to replace all occurrences of particular word in given a file. (iv) Write a script using sedcommand to print duplicated lines in input. b) Execute the following scripts using awk / perl languages: (i) Write an awk script to print all even numbers in a given range. (ii) Write an awk script to develop a Fibonacci series (take user input for number of terms). (iii) Write a perl script to sort elements of an array. (iv) Write a perl script to check a number is prime or not.	**06**	LO2, L03, L06

Text Books:

1. S. Das, Unix Concepts and Applications, 4th ed., McGraw Hill, 2017.
2. R. Michael, Mastering Unix Shell Scripting, 2nd ed., Wiley, 2008.
3. D. Ambawade, D. Shah, Linux Labs and Open Source Technologies, Dreamtech Press, 2014.

References:

1. Y. Kanetkar, Unix Shell Programming, BPB Publications, 2003.
2. B. Forouzan and R. Gilberg, Unix and Shell Programming, Cengage Learning, 2003.

Term Work: Term Work shall consist of at least 10 to 12 practical's based on the above list. Also Term work Journal must include at least 2 assignments.

Term Work Marks: 25 Marks (Total marks) = 15 Marks (Experiment) + 5 Marks (Assignments) + 5 Marks (Attendance)

Practical & Oral Exam: An Oral & Practical exam will be held based on the above syllabus.

Lab Code	Lab Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITL403	Microprocessor Lab	--	02	--	--	01	--	01

Lab Code	Lab Name	Examination Scheme						
		Theory Marks				Term Work	Pract. /Oral	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg.				
ITL403	Microprocessor Lab	--	--	--	--	25	25	50

Lab Objectives:

Sr. No.	Lab Objectives
The Lab experiments aims:	
1	Learn assembling and disassembling of PC
2	Design, simulate and implement different digital circuits
3	Get hands on experience with Assembly Language Programming.
4	Study interfacing of peripheral devices with 8086 microprocessor.
5	Realize techniques for faster execution of instructions and improve speed of operation and performance of microprocessors.
6	Write and debug programs in TASM/MASM/hardware kits

Lab Outcomes:

Sr. No.	Lab Outcomes	Cognitive levels of attainment as per Bloom's Taxonomy
On successful completion, of course, learner/student will be able to:		
1	Demonstrate various components and peripheral of computer system	L2
2	Analyze and design combinational circuits	L4, L6
3	Build a program on a microprocessor using arithmetic & logical instruction set of 8086.	L3
4	Develop the assembly level programming using 8086 loop instruction set	L6
5	Write programs based on string and procedure for 8086 microprocessor.	L1
6	Design interfacing of peripheral devices with 8086 microprocessor.	L6

Prerequisite: Logic Design, Programming Languages(C, C++)

Hardware & Software Requirements:

NOTE: Programs can be executed on assembler or hardware boards.

Hardware Requirement:	**Software requirement:**
➢ Motherboard, RAM, Processor, Connectors, Cables, SMPS, HDD, Monitor, Graphics card (optional), and Cabinet. ➢ 8086 microprocessor experiment kits with specified interfacing study boards	➢ Microsoft Macro Assembler (TASM)/Turbo Assembler (TASM) ➢ Virtual simulator lab. ➢ Proteus design suite

DETAILED SYLLABUS:

Sr. No.	Module	Detailed Content	Hours	LO Mapping
I	PC Assembly	Study of PC Motherboard Technology (South Bridge and North Bridge), Internal Components and Connections used in computer system.	**02**	LO1
II	Implementation of combinational circuits	1. Verify the truth table of various logic gates (basic and universal gates) 2. Realize Half adder and Full adder 3. Implementation of MUX and DeMUX	**06**	LO2
III	Arithmetic and logical operations in 8086 Assembly language programming	1. Program for 16 bit BCD addition 2. Program to evaluate given logical expression. 3. Convert two digit Packed BCD to Unpacked BCD. (any two)	**05**	LO3
IV	Loop operations in 8086 Assembly language programming	1. Program to move set of numbers from one memory block to another. 2. Program to count number of 1's and 0's in a given 8 bit number 3. Program to find even and odd numbers from a given list 4. Program to search for a given number (any three)	**06**	LO4
V	String &Procedure in 8086 Assembly language programming	1. Check whether a given string is a palindrome or not. 2. Compute the factorial of a positive integer 'n' using procedure. OR Generate the first 'n' Fibonacci numbers.	**04**	LO5
VI	Interfacing with 8086 microprocessor	1. Interfacing Seven Segment Display 2. Interfacing keyboard matrix 3. Interfacing DAC (any one)	**03**	LO6

Text Books:

1. Scott Mueller, ”Upgrading and repairing PCs”, Pearson,
2. R. P. Jain, “Modern Digital Electronics”, Tata McGraw Hill.
3. John Uffenbeck, “8086/8088 family: Design Programming and Interfacing:”Pearson Education

Reference Books:

1. M. Morris Mano, “Digital Logic and computer Design”, PHI
2. K Bhurchandi, “Advanced Microprocessors & Peripherals”, Tata McGraw-Hill Education

Term Work: Term Work shall consist of at least 10 to 12 practical’s based on the above list. Also Term work Journal must include at least 2 assignments.

Term Work Marks: 25 Marks (Total marks) = 15 Marks (Experiment) + 5 Marks (Assignments) + 5 Marks (Attendance)

Practical & Oral Exam: An Oral & Practical exam will be held based on the above syllabus.

Lab Code	Lab Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITL404	Python Lab (SBL)	--	04	--	--	02	--	02

Lab Code	Lab Name	Examination Scheme						
		Theory Marks				Term Work	Pract. /Oral	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg.				
ITL404	Python Lab (SBL)	--	--	--	--	25	25	50

Lab Objectives:

Sr. No.	Lab Objectives
The Lab experiments aims:	
1	Basics of python including data types, operator, conditional statements, looping statements, input and output functions in Python
2	List, tuple, set, dictionary, string, array and functions
3	Object Oriented Programming concepts in python
4	Concepts of modules, packages, multithreading and exception handling
5	File handling, GUI & database programming
6	Data visualization using Matplotlib, Data analysis using Pandas and Web programming using Flask

Lab Outcomes:

Sr. No.	Lab Outcomes	Cognitive levels of attainment as per Bloom's Taxonomy
On successful completion, of course, learner/student will be able to:		
1	Understand the structure, syntax, and semantics of the Python language.	L1, L2
2	Interpret advanced data types and functions in python	L1, L2
3	illustrate the concepts of object-oriented programming as used in Python	L2
4	Create Python applications using modules, packages, multithreading and exception handling.	L6
5	Gain proficiency in writing File Handling programs ,also create GUI applications and evaluate database operations in python.	L1, L2
6	Design and Develop cost-effective robust applications using the latest Python trends and technologies	L6

Prerequisite: Structured Programming Approach & Java Programming Lab

Hardware & Software Requirements:

Hardware Requirements	Software Requirements	Other Requirements
PC With following Configuration 1. Intel Dual core Processor or higher 2. Minimum 2 GB RAM 3. Minimum 40 GB Hard disk 4. Network interface card	1. Windows or Linux Desktop OS 2. Python 3.6 or higher 3. Notepad ++ 4.Python IDEs like IDLE, Pycharm, Pydev, Netbeans or Eclipse 5. Mysql	1. Internet Connection for installing additional packages if required

DETAILED SYLLABUS:

Sr. No.	Module	Detailed Content	Hours	LO Mapping
0	**Prerequisite**	Python IDE installation and environment setup.	**02**	
I	**Basics of Python**	Introduction, Features, Python building blocks – Identifiers, Keywords, Indention, Variables and Comments, Basic data types (Numeric, Boolean, Compound) Operators: Arithmetic, comparison, relational, assignment, logical, bitwise, membership, identity operators, operator precedence Control flow statements: Conditional statements (if, if…else, nested if) Looping in Python (while loop, for loop, nested loops) Loop manipulation using continue, pass, break. Input/output Functions, Decorators, Iterators and Generators.	**08**	LO 1
II	**Advanced data types & Functions**	Lists: a) Defining lists, accessing values in list, deleting values in list, updating lists b) Basic list operations c) Built-in list functions Tuples: a) Accessing values in Tuples, deleting values in Tuples, and updating Tuples b) Basic Tuple operations c) Built-in Tuple functions Dictionaries: a) Accessing values in Dictionary, deleting values in Dictionary, and updating Dictionary b) Basic Dictionary operations c) Built-in Dictionary functions Sets: a) Accessing values in Set, deleting values in Set, updating Sets b) Basic Set operations, c) Built-in Set functions Strings: a) String initialization, Indexing, Slicing, Concatenation, Membership & Immutability b) Built-in String functions Arrays: a) Working with Single dimensional Arrays: Creating, importing, Indexing, Slicing, copying and processing array arrays. b) Working with Multi-dimensional Arrays using Numpy: Mathematical operations, Matrix operations, aggregate and other Built-in functions	**09**	LO 1 LO 2

		Functions: a) Built-in functions in python b) Defining function, calling function, returning values, passing parameters c) Nested and Recursive functions d) Anonymous Functions (Lambda, Map, Reduce, Filter)		
III	**Object Oriented Programming**	Overview of Object-oriented programming, Creating Classes and Objects, Self-Variable, Constructors, Inner class, Static method, Namespaces. Inheritance: Types of Inheritance (Single, Multiple, Multi-level, Hierarchical), Super() method, Constructors in inheritance, operator overloading, Method overloading, Method overriding, Abstract class, Abstract method, Interfaces in Python.	**08**	LO 1 LO 3
IV	**Exploring concept of modules, packages, multithreading and exception handling**	Modules: Writing modules, importing objects from modules, Python built-in modules (e.g. Numeric and Mathematical module, Functional Programming module, Regular Expression module), Namespace and Scoping. Packages: creating user defined packages and importing packages. Multi-threading: process vs thread, use of threads, types of threads, creating threads in python, thread synchronization, deadlock of threads. Exception handling: Compile time errors, Runtime errors, exceptions, types of exception, try statement, except block, raise statement, Assert statement, User-Defined Exceptions.	**06**	LO 1 LO 4
V	**File handling, GUI & database programming**	File Handling: Opening file in different modes, closing a file, writing to a file, accessing file contents using standard library functions, reading from a file – read (), readline (), readlines (), Renaming and Deleting a file, File Exceptions, Pickle in Python. Graphical user interface (GUI): different GUI tools in python (Tkinter, PyQt, Kivy etc.), Working with containers, Canvas, Frame, Widgets (Button, Label, Text, Scrollbar, Check button, Radio button, Entry, Spinbox, Message etc.) Connecting GUI with databases to perform CRUD operations. (on supported databases like SQLite, MySQL, Oracle, PostgreSQL etc.).	**09**	LO 1 LO 5
VI	**Data visualization, analysis and web programming using python**	Visualization using Matplotlib: Matplotlib with Numpy, working with plots (line plot, bar graph, histogram, scatter plot, area plot, pie chart etc.), working with multiple figures. Data manipulation and analysis using Pandas: Introduction to Pandas, importing data into Python, series, data frames, indexing data frames, basic operations with data frame, filtering, combining and merging data frames, Removing Duplicates. SciPy: Linear algebra functions using Numpy and Scipy. Web programming: Introduction to Flask, Creating a Basic Flask Application, Build a Simple REST API using Flask	**10**	LO 1 LO 6

List of Experiments/Mini-Project.

1)	Write python programs to understand a) Basic data types, Operators, expressions and Input Output Statements b) Control flow statements: Conditional statements (if, if…else, nested if) c) Looping in Python (while loop, for loop, nested loops) d) Decorators, Iterators and Generators.
2)	Write python programs to understand a) Different List and Tuple operations using Built-in functions b) Built-in Set and String functions c) Basic Array operations on 1-D and Multidimensional arrays using Numpy d) Implementing User defined and Anonymous Functions
3)	Write python programs to understand a) Classes, Objects, Constructors, Inner class and Static method b) Different types of Inheritance c) Polymorphism using Operator overloading, Method overloading, Method overriding, Abstract class, Abstract method and Interfaces in Python.
4)	Write python programs to understand a) Creating User-defined modules/packages and import them in a program b) Creating user defined multithreaded application with thread synchronization and deadlocks c) Creating a menu driven application which should cover all the built-in exceptions in python
5)	Write python programs to understand a) Different File Handling operations in Python b) Designing Graphical user interface (GUI) using built-in tools in python (Tkinter, PyQt, Kivy etc.). c) GUI database connectivity to perform CRUD operations in python (Use any one database like SQLite, MySQL, Oracle, PostgreSQL etc.)
6)	Write python programs to implement a) Different types of plots using Numpy and Matplotlob b) Basic operations using pandas like series, data frames, indexing, filtering, combining and merging data frames. c) Different Linear algebra functions using Scipy. d) A Basic Flask Application to build a Simple REST API.

- **Mini Project**

Mini-project have to be developed in a group of three students which should cover all above topics.
Suggested Mini-Project Topics:

1. Railway reservation system	27 IT Team Workspace	52. Business Directory	78. Practice Test Management.
2. Inventory Management system.	29 Job Requisition and Interview Management	53. Education Directory	79. Asset Management System
3 Classroom Management	28 Knowledge Base	54. Dental Clinic Management	80. Travel Agency System.
4 Clinical Trial Initiation and Management	29 Lending Library	55. Fund Raising Management	81. Placement Management System.

5 Competitive Analysis Web Site	30 Physical Asset Tracking and Management	56. Clinic/ Health Management	82. Polls Management
6 Discussion Forum website	31 Project Tracking Workspace	57. Cable Management System	83. Customer Management
7 Disputed Invoice Management	32. Shopping Cart .	58. Survey Creation and Analytics	84. Project Management System.
8 Employee Training Scheduling and Materials	33 Knowledge Base	59. Museum Management System	85. Network Marketing System
9 Equity Research Management	34 Lending Library	60. Multi-Level Marketing System	86. Yoga Health Care Management
10 Integrated Marketing Campaign Tracking	35 Physical Asset Tracking and Management	61. Learning Management System	87. Personal Finance Management System
11 Manufacturing Process Managements	36 Project Tracking Workspace	62. Knowledge Management System	88. Real Estate Management System
12 Product and Marketing Requirements Planning	37 Room and Equipment Reservations	63. Missing Person Site	89. Stock Mutual Funds Management
13 Request for Proposal Software	38 Sales Lead Pipeline	64. Disaster Management Site	90. Careers and Employment Management System
14 Sports League Management	39. Yellow Pages & Business Directory	65. Job Management Site	91. Music Albums Management System
15 Absence Request and Vacation Schedule Management	40. Time & Billing	66. Financial Portfolio Management	92. Classified Ads Managements
16 Budgeting and Tracking Multiple Projects	41. Class Room Management	67. Market Research Management	93. Property Management System
17 Bug Database Management	42. Expense Report Database	68. Order Management System	94. Sales & Retail Management
18 Call Center Management Software	43. Sales Contact Management Database	69. Point of Sale	95. Dating Site
19 Change Request Management	44. Inventory Management Database	70. Advertisement /Banner Management and Analytics	96. Hotel Management System
20 Compliance Process Support Site	45. Issue Database	71. Export Management System	97. Search Engine
21 Contacts Management Software	46. Event Management Database	72. Invoice Management	98. Online News Paper Site
22 Document Library and Review	47. Service Call Management Database	73. Recruitment Management System	99. Image Gallery
23 Event Planning and Management	48. Accounting Ledger Database	74. Articles / Blog / Wiki Web site	100. Staffing and Human Capital Management
24 Expense Reimbursement and Approval	49. Asset Tracking Database	75. Online Planner	101. Development of a feature-rich, practical Online Survey Tool (OST)
25 Help Desk and Ticket Management	50. Cycle Factory Works Management	76. Mock Tests and Examination Management	102 Development of a Web/Email based Search Engine
26 Inventory Tracking	51. Sales Corporation Management	77. Examination System	103. Development of a web-based Recruitment Process System for the HR group for a company

Text Books:

1. Dr. R. Nageswara Rao," Core Python Programming" , Dreamtech Press, Wiley Publication
2. M. T. Savaliya , R. K. Maurya, "Programming through Python", StarEdu Solutions.
3. E Balagurusamy, "Introduction to computing and problem-solving using python", McGraw Hill Publication.

References:

1. Zed A. Shaw, "Learn Python 3 the Hard Way", Zed Shaw's Hard Way Series.
2. Martin C. Brown," Python: The Complete Reference", McGraw-Hill Publication.
3. Paul Barry," Head First Python", 2nd Edition, O'Reilly Media, Inc.

Online resources:

1) https://docs.scipy.org/doc/numpy/user/quickstart.html
2) https://matplotlib.org/tutorials/
3) https://pandas.pydata.org/docs/getting_started/
4) https://www.geeksforgeeks.org/python-build-a-rest-api-using-flask/

Term Work:

The Term work shall consist of at least 15 practical based on the above list. The term work Journal must include at least 2 Programming assignments. The Programming assignments should be based on real world applications which cover concepts from more than one modules of syllabus**.**

Term Work Marks: 25 Marks (Total marks) = 15 Marks (Experiment) + 5 Marks (Assignments/tutorial/write up) + 5 Marks (Attendance)

Practical & Oral Exam: An Oral & Practical exam will be held based on the above syllabus.

Course Code	Course Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITM401	Mini Project – 1 B for Python based automation projects	--	04	--	--	02	--	02

Course Code	Course Name	Examination Scheme						
		Theory Marks				Term Work	Pract. /Oral	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg.				
ITM401	Mini Project – 1 B for Python based automation projects	--	--	--	--	25	25	50

Course Objectives

1. To acquaint with the process of identifying the needs and converting it into the problem.
2. To familiarize the process of solving the problem in a group.
3. To acquaint with the process of applying basic engineering fundamentalsto attempt solutions to the problems.
4. To inculcate the process of self-learning and research.

Course Outcome: Learner will be able to…

1. Identify problems based on societal /research needs.
2. Apply Knowledge and skill to solve societal problems in a group.
3. Develop interpersonal skills to work as member of a group or leader.
4. Draw the proper inferences from available results through theoretical/ experimental/simulations.
5. Analyse the impact of solutions in societal and environmental context for sustainable development.
6. Use standard norms of engineering practices
7. Excel in written and oral communication.
8. Demonstrate capabilities of self-learning in a group, which leads to life long learning.
9. Demonstrate project management principles during project work.

Guidelines for Mini Project

- Students shall form a group of 3 to 4 students, while forming a group shall not be allowed less than three or more than four students, as it is a group activity.
- Students should do survey and identify needs, which shall be converted into problem statement for mini project in consultation with faculty supervisor/head of department/internal committee of faculties.
- Students hall submit implementation plan in the form of Gantt/PERT/CPM chart, which will cover weekly activity of mini project.
- A log book to be prepared by each group, wherein group can record weekly work progress, guide/supervisor can verify and record notes/comments.
- Faculty supervisor may give inputs to students during mini project activity;however, focus shall be on self-learning.
- Students in a group shall understand problem effectively, propose multiple solution and select best possible solution in consultation with guide/ supervisor.

- Students shall convert the best solution into working model using various components of their domain areas and demonstrate.
- The solution to be validated with proper justification and report to be compiled in standard format of University of Mumbai.
- With the focus on the self-learning, innovation, addressing societal problems and entrepreneurship quality development within the students through the Mini Projects, it is preferable that a single project of appropriate level and quality to be carried out in two semesters by all the groups of the students. i.e. Mini Project 1 in semester III and IV. Similarly, Mini Project 2 in semesters V and VI.
- However, based on the individual students or group capability, with the mentor's recommendations, if the proposed Mini Project adhering to the qualitative aspects mentioned above gets completed in odd semester, then that group can be allowed to work on the extension of the Mini Project with suitable improvements/modifications or a completely new project idea in even semester. This policy can be adopted on case by case basis.

Guidelines for Assessment of Mini Project:

Term Work

- The review/ progress monitoring committee shall be constituted by head of departments of each institute. The progress of mini project to be evaluated on continuous basis, minimum two reviews in each semester.
- In continuous assessment focus shall also be on each individual student, assessment based on individual's contribution in group activity, their understanding and response to questions.
- Distribution of Term work marks for both semesters shall be as below;
    - Marks awarded by guide/supervisor based on log book : 10
    - Marks awarded by review committee : 10
    - Quality of Project report : 05

Review/progress monitoring committee may consider following points for assessment based on either one year or half year project as mentioned in general guidelines.

One-year project:

- In first semester entire theoretical solution shall be ready, including components/system selection and cost analysis. Two reviews will be conducted based on presentation given by students group.
    - First shall be for finalisation of problem
    - Second shall be on finalisation of proposed solution of problem.
- In second semester expected work shall be procurement of component's/systems, building of working prototype, testing and validation of results based on work completed in an earlier semester.
    - First review is based on readiness of building working prototype to be conducted.
    - Second review shall be based on poster presentation cum demonstration of working model in last month of the said semester.

Half-year project:

- In this case in one semester students' group shall complete project in all aspects including,
    - Identification of need/problem
    - Proposed final solution
    - Procurement of components/systems
    - Building prototype and testing
- Two reviews will be conducted for continuous assessment,
    - First shall be for finalisation of problem and proposed solution
    - Second shall be for implementation and testing of solution.

Assessment criteria of Mini Project.

Mini Project shall be assessed based on following criteria;

1. Quality of survey/ need identification
2. Clarity of Problem definition based on need.
3. Innovativeness in solutions
4. Feasibility of proposed problem solutions and selection of best solution
5. Cost effectiveness
6. Societal impact
7. Innovativeness
8. Cost effectiveness and Societal impact
9. Full functioning of working model as per stated requirements
10. Effective use of skill sets
11. Effective use of standard engineering norms
12. Contribution of an individual's as member or leader
13. Clarity in written and oral communication

- In **one year, project**, first semester evaluation may be based on first six criteria's and remaining may be used for second semester evaluation of performance of students in mini project.
- In case of **half year project** all criteria's in generic may be considered for evaluation of performance of students in mini project.

Guidelines for Assessment of Mini Project Practical/Oral Examination:

- Report should be prepared as per the guidelines issued by the University of Mumbai.
- Mini Project shall be assessed through a presentation and demonstration of working model by the student project group to a panel of Internal and External Examiners preferably from industry or research organisations having experience of more than five years approved by head of Institution.
- Students shall be motivated to publish a paper based on the work in Conferences/students competitions.

Mini Project shall be assessed based on following points;

1. Quality of problem and Clarity
2. Innovativeness in solutions
3. Cost effectiveness and Societal impact
4. Full functioning of working model as per stated requirements
5. Effective use of skill sets
6. Effective use of standard engineering norms
7. Contribution of an individual's as member or leader
8. Clarity in written and oral communication