SFIT/IQAC/FE-G1-SS/2024-25/R0

Date:25/07/2024

St. Francis Institute of Technology

(Engineering College)

(An Autonomous Institute, Affiliated to University of Mumbai)

S.V.P. Road, Borivli (W), Mumbai

B. E. Second Year Scheme and Syllabus
Computer Engineering

Approved by: - Board of Studies

Approved by: - Academic Council of St. Francis Institute of Technology

w.e.f. Academic Year 2024 – 2025

Table of Contents

AC:
Item No. ____

St. Francis Institute of Technology

Syllabus for Approval

Date:

Sr. No	Heading	Particulars
1.	Title of the Course	**Second Year B.E. Computer Engineering**
2.	Eligibility for Admission	**After Passing First Year Engineering as per the Ordinance 0.6242**
3.	Passing Marks	**40%**
4.	Ordinances / Regulations (if any)	**Ordinance 0.6242**
5.	No. of Years / Semesters	**8 semesters**
6.	Level	~~P.G. /~~ **U.G./** ~~Diploma /~~ Certificate (Strike out which is not applicable)
7.	Pattern	~~Yearly~~ / **Semester** (Strike out which is not applicable)
8.	Status	**New /** ~~Revised~~ (Strike out which is not applicable)
9.	To be implemented from Academic Year	**With effect from Academic Year: 2024-2025**

Dr. Sincy George
Principal
St Francis Institute of Technology

Dr. Kavita Sonawane
HOD, Computer Engineering Dept.
St Francis Institute of Technology

Preamble

To meet the challenge of ensuring excellence in engineering education, the issue of quality needs to be addressed, debated and taken forward in a systematic manner. Accreditation is the principal means of quality assurance in higher education. The major emphasis of accreditation process is to measure the outcomes of the program that is being accredited. In line with this Faculty of Science and Technology (in particular Engineering) of University of Mumbai has taken a lead in incorporating philosophy of outcome based education in the process of curriculum development.

Faculty resolved that course objectives and course outcomes are to be clearly defined for each course, so that all faculty members in affiliated institutes understand the depth and approach of course to be taught, which will enhance learner's learning process. Choice based Credit and grading system enables a much-required shift in focus from teacher-centric to learner-centric education since the workload estimated is based on the investment of time in learning and not in teaching. It also focuses on continuous evaluation which will enhance the quality of education. Credit assignment for courses is based on 15 weeks teaching learning process, however content of courses is to be taught in 13 weeks and remaining 2 weeks to be utilized for revision, guest lectures, coverage of content beyond syllabus etc.

There was a concern that the earlier revised curriculum more focused on providing information and knowledge across various domains of the said program, which led to heavily loading of students in terms of direct contact hours. In this regard, faculty of science and technology resolved that to minimize the burden of contact hours, total credits of entire program will be of 170, wherein focus is not only on providing knowledge but also on building skills, attitude and self-learning. Therefore in the present curriculum skill based laboratories and mini projects are made mandatory across all disciplines of engineering in second and third year of programs, which will definitely facilitate self-learning of students. The overall credits and approach of curriculum proposed in the present revision is in line with AICTE model curriculum.

The present curriculum will be implemented for Second Year it will be implemented for 24-25, Third year for 24-25, and 25-26. For Final Year of Engineering it will be implemented for the academic year 2024-25. 2025-26, 2026-27.

Dr. Sincy George
Principal
St Francis Institute of Technology

Dr. Kavita Sonawane
HOD, Computer Engineering Dept.
St Francis Institute of Technology

Incorporation and Implementation of Online Contents from NPTEL/ Swayam Platform

The curriculum revision is mainly focused on knowledge component, skill based activities and project based activities. Self-learning opportunities are provided to learners. In the revision process this time in particular Revised syllabus of 'C ' scheme wherever possible additional resource links of platforms such as NPTEL, Swayam are appropriately provided. In an earlier revision of curriculum in the year 2012 and 2016 in Revised scheme 'A' and 'B' respectively, efforts were made to use online contents more appropriately as additional learning materials to enhance learning of students.

In the current revision based on the recommendation of AICTE model curriculum overall credits are reduced to 171, to provide opportunity of self-learning to learner. Learners are now getting sufficient time for self-learning either through online courses or additional projects for enhancing their knowledge and skill sets.

The Principals/ HoD's/ Faculties of all the institute are required to motivate and encourage learners to use additional online resources available on platforms such as NPTEL/ Swayam. Learners can be advised to take up online courses, on successful completion they are required to submit certification for the same. This will definitely help learners to facilitate their enhanced learning based on their interest.

Dr. Sincy George
Principal
St Francis Institute of Technology

Dr. Kavita Sonawane
HOD, Computer Engineering Dept.
St Francis Institute of Technology

Preface by Board of Studies in Computer Engineering

Dear Students and Teachers, we, the members of Board of Studies Computer Engineering, are very happy to present Second Year Computer Engineering syllabus effective from the Academic Year 2020-21 (REV-2019'C' Scheme). We are sure you will find this syllabus interesting and challenging.

Computer Engineering is one of the most sought-after courses amongst engineering students hence there is a continuous requirement of revision of syllabus. The syllabus focuses on providing a sound theoretical background as well as good practical exposure to students in the relevant areas. It is intended to provide a modern, industry-oriented education in Computer Engineering. It aims at producing trained professionals who can successfully acquainted with the demands of the industry worldwide. They obtain skills and experience in up-to-date the knowledge to analysis, design, implementation, validation, and documentation of computer software and systems.

The revised syllabus falls in line with the objectives of affiliating University, AICTE, UGC, and various accreditation agencies by keeping an eye on the technological developments, innovations, and industry requirements.

The salient features of the revised syllabus are:

1. Reduction in credits to 170 is implemented to ensure that students have more time for extracurricular activities, innovations, and research.
2. Introduction of Skill Based Lab and Mini Project to showcase their talent by doing innovative projects that strengthen their profile and increases the chance of employability
3. Students are encouraged to take up part of course through MOOCs platform SWAYAM

We would like to place on record our gratefulness to the faculty, students, industry experts and stakeholders for having helped us in the formulation of this syllabus.

Board of Studies in Computer Engineering

Dr. Kavita Sonawane	:	Chairperson
Dr. Sudip Thepade	:	Subject Expert
Dr. Sunil Mane	:	Subject Expert
Dr. Narendra Shekokar	:	VC Nominee
Mr. Mukesh Jain	:	Industry Expert
Dr. Raj Dabre	:	Alumni
Dr. Padmaja Joshi	:	Special Courses -Expert

1. Program Structure for Second Year B.E Computer Engineering (with Effect from 2024-25)

1.1 Second Year Scheme of the Syllabus

Table 1: Contact hours and credit distribution of courses of S.E Computer Engineering

Course Code	Course	Contact Hours				Credits Assigned			
		Theory (Th)	Practical (P)	Tutorial (T)	Total	Theory (Th)	Practical (P)	Tutorial (T)	Total
Semester IV									
CSC401	Engineering Mathematics IV	3	-	1*	4	3	-	1	4
CSC402	Analysis of Algorithm	3	-	-	3	3	-	-	3
CSC403	Database Management System	3	-	-	3	3	-	-	3
CSC404	Operating System	3	-	-	3	3	-	-	3
CSC405	Microprocessor	3	-	-	3	3	-	-	3
CSL401	Analysis of Algorithm Lab	-	2	-	2	-	1	-	1
CSL402	Database Management System Lab	-	2	-	2	-	1	-	1
CSL403	Operating System Lab	-	2	-	2	-	1	-	1
CSL404	Microprocessor Lab	-	2	-	2	-	1	-	1
CSL405	Skill Base Lab Course: Python Programming	-	2*+2	-	4	-	2	-	2
CSM401	Mini Project 1-B	-	4$	-	4	-	2	-	2
	Total	**15**	**16**	**1**	**32**	**15**	**7**	**1**	**24**

*Should be conducted batch wise and
$ indicates workload of Learner (Not Faculty), Students can form groups with minimum 2 (Two) and not more than 4 (Four), Faculty Load: 1 hour per week per four groups

1.2 Evaluation and Examination Scheme

Table 2: Marks distribution of courses for S.E Computer Engineering

			Theory					Lab		Total
S.No.	**Course Code**	**Course**	**ISE1**	**ISE2**	**MSE**	**ESE**	**Total**	**ISE (LAB)**	**P/O E**	
Semester IV										
1	CSC401	Engineering Mathematics IV	10	10	20	60	100	-	-	100
2	CSC402	Analysis of Algorithm	10	10	20	60	100	-	-	100
3	CSC403	Database Management System	10	10	20	60	100	-	-	100
4	CSC404	Operating System	10	10	20	60	100	-	-	100
5	CSC405	Microprocessor	10	10	20	60	100	-	-	100
6	CSL401	Analysis of Algorithm Lab	-	-	-	-	-	25	25	50
7	CSL402	Database Management System Lab	-	-	-	-	-	25	25	50
8	CSL403	Operating System Lab	-	-	-	-	-	25	25	50
9	CSL404	Microprocessor Lab	-	-	-	-	-	25	-	25
10	CSL405	Skill Base Lab Course: Python Programming	-	-	-	-	-	25	-	25
11	CSM401	Mini Project 1-B	-	-	-	-	-	25	25	50
		Total	**50**	**50**	**100**	**300**	**500**	**125**	**125**	**750**

Note:

Evaluations includes In Semester Evaluation (ISE), Mid-Semester Examination (MSE), End Semester Examination (ESE), Practical Examination (PE), Oral Examination (OrE) and Project Examination (PrE). Detailed evaluation pattern given in the Examination Conduction rules and Guidelines.

- **ISE** - Includes home assignments, group assignments, quizzes, presentations, experiments, mock tests, tutorials, etc.
- **MSE** - A written assessment covering up to 50% of the syllabus, conducted at mid-semester.
- **ESE** - A written assessment covering 100% of the syllabus, conducted at the end of the semester. Modules covered in the mid-semester exam will have a 30-40 % weightage, with the remaining 60 -70% weightage assigned to the rest of the modules.
- **PE, OrE and PrE** - Practical, oral, project examinations are conducted by a pair of internal and external examiner at the end of the semester

2 Engineering Mathematics-IV (CSC401)

2.1 Teaching Scheme

Course Code	Course Name	Contact Hours	Credits
CSC401	Engineering Mathematics - IV	04	04

2.2 Examination and Evaluation Scheme

Formative Assessment		Summative Assessment				Total Marks
ISE		MSE		ESE		
Marks	Duration	Marks	Duration (Hrs.)	Marks	Duration (Hrs.)	
20	CA	20	1	60	2.5	100

2.3 Course Objectives

S. No.	Objectives
1	Matrix algebra to understand engineering problems.
2	Line and Contour integrals and expansion of a complex valued function in a power series.
3	Z-Transforms and Inverse Z-Transforms with its properties.
4	The concepts of probability distributions and sampling theory for small samples.
5	Linear and Non-linear programming problems of optimization.

2.4 Course Outcomes

The student will be able to:	
CO1	*apply* the concepts of eigenvalues and eigenvectors in engineering problems.
CO2	*use* the concepts of Complex Integration for evaluating integrals, computing residues & evaluate various contour integrals.
CO3	*apply* the concept of Z- transformation and inverse in engineering problems.
CO4	*use* the concept of probability distribution and sampling theory to engineering problems
CO5	*apply* the concept of Linear Programming Problems to optimization.
CO6	*solve* Non-Linear Programming Problems for optimization of engineering problems.

2.5 Course Contents

Module	Unit	Detailed Contents	Hours
	Pre-requisite:		
		Engineering Mathematics-I, Engineering Mathematics-II, Engineering Mathematics-III, Binomial Distribution.	
1	**Linear Algebra (Theory of Matrices)**		**07**
	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	
	1.4	**Self-learning Topics:** Derogatory and non-derogatory matrices,	

		Functions of Square Matrix, Linear Transformations, Quadratic forms.	
2	**Complex Integration**		**07**
	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 off(z), Residues, Cauchy's Residue Theorem (without proof)	
	2.4	**Self-learning Topics:** Application of Residue Theorem to evaluate real integrations.	
3	**Z Transform**		**06**
	3.1	Definition and Region of Convergence, Transform of Standard Functions: $\{k^n\ a^k\}$, $\{a^{\ |k|}\}$, $\{^{k+n}{}_n C.a^k\}$,$\{c\ k \sin(\alpha k + \beta)\}$,$\{c\ k \sinh \alpha k\}$,$\{c\ k \cosh \alpha k\}$.	
	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.	
	3.4	**Self-learning Topics:** Initial value theorem, Final value theorem, Inverse of Z Transform by Binomial Expansion	
4	**Probability Distribution and Sampling Theory**		**07**
	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.	
	4.4	**Self-learning Topics:** Test significance for Large samples, Estimate parameters of a population, Yate's Correction	
5	**Linear Programming Problems**		**06**
	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	
	5.4	**Self-learning Topics:** Sensitivity Analysis, Two-Phase Simplex Method, Revised Simplex Method.	
6	**Nonlinear Programming Problems**		**06**
	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	
	6.4	**Self-learning Topics:** Problems with two inequality constraints, Unconstrained optimization: One-dimensional search method (Golden Search method, Newton's method). Gradient Search method	
		Total Hours	**39**

2.6 Suggested Learning Resources:

2.6.1 Textbooks

1. Erwin Kreyszig, "Advanced Engineering Mathematics", John Wiley & Sons.
2. R. K. Jain and S. R. K. Iyengar, "Advanced Engineering Mathematics", Narosa.

3. Brown and Churchill, “Complex Variables and Applications”, McGraw-Hill Education.
4. T. Veerarajan, “Probability, Statistics and Random Processes”, McGraw-Hill Education.
5. Hamdy A Taha, “Operations Research: An Introduction”, Pearson.
6. S.S. Rao, “Engineering Optimization: Theory and Practice”, Wiley-Blackwell.
7. Hira and Gupta, “Operations Research”, S. Chand Publication.

2.6.2 Reference Books

1. Erwin Kreyszig, “Advanced Engineering Mathematics”, John Wiley & Sons

2.6.3 Web Resources

1. https://onlinecourses.nptel.ac.in/noc25_ge10/preview

3 Analysis of Algorithms (CSC402)

3.1 Teaching Scheme

Course Code	Course Name	Contact Hours	Credits
CSC402	Analysis of Algorithms	03	03

3.2 Examination and Evaluation Scheme

Formative Assessment		Summative Assessment				Total Marks
ISE		MSE		ESE		
Marks	Duration	Marks	Duration (Hrs.)	Marks	Duration (Hrs.)	
20	CA	20	1	60	2.5	100

3.3 Course Objectives

S. No.	Objectives
1	To provide mathematical approaches for Analysis of Algorithms
2	To understand and solve problems using various algorithmic approaches
3	To analyze algorithms using various methods

3.4 Course Outcomes

The student will be able to:	
CO1	*analyze* the running time and space complexity of algorithms.
CO2	*describe*, *apply* and *analyze* the complexity of divide and conquer strategy.
CO3	*describe*, *apply* and *analyze* the complexity of greedy strategy.
CO4	*describe*, *apply* and *analyze* the complexity of dynamic programming strategy.
CO5	*explain* and *apply backtracking*, branch and bound.
CO6	*explain* and *apply* string matching techniques.

3.5 Course Contents

Module	Unit	Detailed Contents	Hours
	Pre-requisite: Data structures, Basic programming		
1	**Introduction**		**08**
	1.1	Performance analysis, space, and time complexity Growth of function, Big-Oh, Omega Theta notation Mathematical background for algorithm analysis. Complexity class: Definition of P, NP, NP-Hard, NP-Complete Analysis of selection sort, insertion sort.	
	1.2	Recurrences: The substitution method, Recursion tree method, Master method	
2	**Divide and Conquer Approach**		**06**
	2.1	General method, Merge sort, Quick sort, Finding minimum and maximum algorithms and their Analysis, Analysis of Binary search.	
3	**Greedy Method Approach**		**06**
	3.1	General Method, Single source shortest path: Dijkstra Algorithm	

		Fractional Knapsack problem, Job sequencing with deadlines, Minimum cost spanning trees: Kruskal and Prim's algorithms	
4	**Dynamic Programming Approach**		**09**
	4.1	General Method, Multistage graphs, Single source shortest path: Bellman Ford Algorithm All pair shortest path: Floyd Warshall Algorithm, Assembly-line scheduling Problem 0/1 knapsack Problem, Travelling Salesperson problem, Longest common subsequence	
5	**Backtracking and Branch and bound**		**06**
	5.1	General Method, Backtracking: N-queen problem, Sum of subsets, Graph coloring	
	5.2	Branch and Bound: Travelling Salesperson Problem, 15 Puzzle problem	
6	**String Matching Algorithms**		**04**
	6.1	The Naïve string-matching algorithm, The Rabin Karp algorithm, The Knuth-Morris-Pratt algorithm	
		Total Hours	**39**

3.6 Suggested Learning Resources:

3.6.1 Textbooks

1. T. H. Cormen, C.E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to algorithms", 2nd Edition, PHI Publication 2005.
2. Ellis Horowitz, Sartaj Sahni, S. Rajsekaran. "Fundamentals of computer algorithms" University Press.

3.6.2 Reference Books

1. Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, "Algorithms", Tata McGraw-Hill Edition.
2. S. K. Basu, "Design Methods and Analysis of Algorithm", PHI.

3.6.3 Web Resources

1. https://nptel.ac.in/courses/106/106/106106131/
2. https://www.coursera.org/specializations/algorithms
3. https://www.coursera.org/specializations/algorithms
4. https://www.mooc-list.com/tags/algorithms

4 Database Management System (CSC403)

4.1 Teaching Scheme

Course Code	Course Name	Contact Hours	Credits
CSC403	Database Management System	03	03

4.2 Examination and Evaluation Scheme

Formative Assessment		Summative Assessment				Total Marks
ISE		MSE		ESE		
Marks	Duration	Marks	Duration (Hrs.)	Marks	Duration (Hrs.)	
20	CA	20	1	60	2.5	100

4.3 Course Objectives

S. No.	Objectives
1	Develop entity relationship data model and its mapping to relational model
2	Learn relational algebra and Formulate SQL queries
3	Apply normalization techniques to normalize the database
4	Understand concept of transaction, concurrency control and recovery techniques.

4.4 Course Outcomes

The student will be able to:	
CO1	*recognize* the need of database management system
CO2	*design* ER and EER diagram for real life applications
CO3	*construct* relational model and *write* relational algebra queries.
CO4	*formulate* SQL queries
CO5	*apply* the concept of normalization to relational database design.
CO6	*describe* the concept of transaction, concurrency and recovery.

4.5 Course Contents

Module	Unit	Detailed Contents	Hours
	Pre-requisite:		
		Data Structures	
1	**Introduction Database Concepts**		**03**
	1.1	Introduction, Characteristics of databases, system	
	1.2	File system v/s Database	
	1.3	Data abstraction and data Independence,	
	1.4	DBMS system architecture	
	1.5	Database Administrator	
2	**Entity–Relationship Data Model**		**06**
	2.1	The Entity-Relationship (ER) Model: Entity types: Weak and strong entity sets, Entity sets	
	2.2	Types of Attributes, Keys	
	2.3	Relationship constraints: Cardinality and Participation,	

	2.4	Extended Entity-Relationship (EER) Model: Generalization, Specialization and Aggregation	
3	**Relational Model and relational Algebra**		**08**
	3.1	Introduction to the Relational Model	
	3.2	Relational schema and concept of keys	
	3.3	Mapping the ER and EER Model to the Relational Model	
	3.4	Mapping the ER and EER Model to the Relational Model	
	3.5	Relational Algebra-operators, Relational Algebra Queries.	
4	**Structured Query Language (SQL)**		**06**
	4.1	Overview of SQL, Data Definition Commands, Integrity constraints: key constraints, Domain Constraints, Referential integrity, check constraint	
	4.2	Data Manipulation commands, Data Control commands	
	4.3	Set and string operations, aggregate function-group by, having,	
	4.4	Views in SQL, joins	
	4.5	Nested and complex queries, Triggers	
5	**Relational-Database Design**		**06**
	5.1	Pitfalls in Relational-Database design	
	5.2	Concept of normalization	
	5.3	Function Dependencies	
	5.4	First Normal Form, 2NF	
	5.5	3NF, BCNF.	
6	**Transactions Management and Concurrency and Recovery**		**10**
	6.1	Transaction concept, Transaction states, ACID properties, Transaction Control Commands	
	6.2	Concurrent Executions, Serializability-Conflict and View	
	6.3	Concurrency Control: Lock-based, Timestamp-based protocols	
	6.4	Recovery System: Log based recovery	
	6.5	Deadlock handling	
		Total Hours	**39**

4.6 Suggested Learning Resources:

4.6.1 Textbooks

1. Korth, Slberchatz, Sudarshan, Database System Concepts, 6thEdition, McGraw Hill
2. Elmasri and Navathe, Fundamentals of Database Systems, 5thEdition, Pearson Education.
3. Raghu Ramkrishnan and Johannes Gehrke, Database Management Systems, TMH

4.6.2 Reference Books

1. Peter Rob and Carlos Coronel, Database Systems Design, Implementation and Management, Thomson Learning, 5thEdition.
2. Dr. P.S. Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press.
3. G. K. Gupta, Database Management Systems, McGraw Hill, 2012

4.6.3 Web Resources

1. https://nptel.ac.in/courses/106/105/106105175/
2. https://swayam.gov.in/nd1_noc19_cs46/preview
3. https://www.classcentral.com/course/swayam-database-management-system-9914
4. https://www.mooc-list.com/tags/dbms

5 Operating System (CSC404)

5.1 Teaching Scheme

Course Code	Course Name	Contact Hours	Credits
CSC404	Operating System	03	03

5.2 Examination and Evaluation Scheme

Formative Assessment		Summative Assessment				Total Marks
ISE		MSE		ESE		
Marks	Duration	Marks	Duration (Hrs.)	Marks	Duration (Hrs.)	
20	CA	20	1	60	2.5	100

5.3 Course Objectives

S. No.	Objectives
1	To introduce basic concepts and functions of operating systems.
2	To understand the concept of process, thread and resource management
3	To understand the concepts of process synchronization and deadlock
4	To understand various Memory, I/O and File management techniques

5.4 Course Outcomes

The student will be able to:	
CO1	*understand* the objectives, functions and structure of OS
CO2	*analyze* the concept of process management and *evaluate* performance of process scheduling algorithms.
CO3	*understand* and *apply* the concepts of synchronization and deadlocks
CO4	*evaluate* performance of Memory allocation and replacement policies
CO5	*understand* the concepts of file management.
CO6	*apply* concepts of I/O management and *analyze* techniques of disk scheduling.

5.5 Course Contents

Module	Unit	Detailed Contents	Hours
	Pre-requisite:		
		Data structures and Computer architecture	
1	**Operating system Overview**		**04**
	1.1	Introduction, Objectives, Functions and Evolution of Operating System	
	1.2	Operating system structures: Layered, Monolithic and Microkernel	
	1.3	Linux Kernel, Shell and System Calls	
2	**Process and Process Scheduling**		**09**
	2.1	Concept of a Process, Process States, Process Description, Process Control Block.	
	2.2	Uniprocessor Scheduling-Types: Preemptive and Non-preemptive scheduling algorithms (FCFS, SJF, SRTN, Priority, RR)	

	2.3	Threads: Definition and Types, Concept of Multithreading	
3	**Process Synchronization and Deadlocks**		**09**
	3.1	Concurrency: Principles of Concurrency, Inter-Process Communication, Process Synchronization.	
	3.2	Mutual Exclusion: Requirements, Hardware Support (TSL), Operating System Support (Semaphores), Producer and Consumer problem.	
	3.3	Principles of Deadlock: Conditions and Resource, Allocation Graphs, Deadlock Prevention, Deadlock Avoidance: Banker's Algorithm, Deadlock Detection and Recovery, Dining Philosophers Problem.	
4	**Memory Management**		**09**
	4.1	Memory Management Requirements, Memory Partitioning: Fixed, Partitioning, Dynamic Partitioning, Memory Allocation Strategies: Best-Fit, First Fit, Worst Fit, Paging and Segmentation, TLB	
	4.2	Virtual Memory: Demand Paging, Page Replacement Strategies: FIFO, Optimal, LRU, Thrashing	
5	**File Management**		**04**
	5.1	Overview, File Organization and Access, File Directories, File Sharing	
6	**I/O management**		**04**
	6.1	I/O devices, Organization of the I/O Function, Disk Organization, I/O Management and Disk Scheduling: FCFS, SSTF, SCAN, CSCAN, LOOK, C-LOOK.	
		Total Hours	**39**

5.6 Suggested Learning Resources:

5.6.1 Textbooks

1. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 8th Edition, 2014, ISBN-10: 0133805913 • ISBN-13: 9780133805918.
2. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating System Concepts,
3. John Wiley &Sons, Inc., 9thEdition, 2016, ISBN 978-81-265-5427-0

5.6.2 Reference Books

1. Achyut Godbole and Atul Kahate, Operating Systems, McGraw Hill Education, 3rd Edition
2. Andrew Tannenbaum, Operating System Design and Implementation, Pearson, 3rd Edition.
3. Maurice J. Bach, "Design of UNIX Operating System", PHI
4. Sumitabha Das, "UNIX: Concepts and Applications", McGraw Hill, 4thEdition

5.6.3 Web Resourses

1. https://swayam.gov.in/nd1_noc19_cs50/preview
2. https://nptel.ac.in/courses/117/106/117106113/
3. https://www.classcentral.com/course/swayam-introduction-to-operating-systems-6559

6 Microprocessor (CSC405)

6.1 Teaching Scheme

Course Code	Course Name	Contact Hours	Credits
CSC405	Microprocessor	03	03

6.2 Examination and Evaluation Scheme

Formative Assessment		Summative Assessment				Total Marks
ISE		MSE		ESE		
Marks	Duration	Marks	Duration (Hrs.)	Marks	Duration (Hrs.)	
20	CA	20	1	60	2.5	100

6.3 Course Objectives

S. No.	Objectives
1	To equip students with the fundamental knowledge and basic technical competence in the field of Microprocessors.
2	To emphasize on instruction set and logic to build assembly language programs.
3	To prepare students for higher processor architectures and embedded systems

6.4 Course Outcomes

The student will be able to:	
CO1	*describe* core concepts of 8086 microprocessor.
CO2	*interpret* the instructions of 8086 and write assembly and Mixed language programs.
CO3	*identify* the specifications of peripheral chip.
CO4	*design* 8086 based system using memory and peripheral chips.
CO5	*appraise* the architecture of advanced processors
CO6	*understand* hyperthreading technology

6.5 Course Contents

Module	Unit	Detailed Contents	Hours
	Pre-requisite:		
		Digital Logic and Computer Architecture	
1	**The Intel Microprocessors 8086 Architecture**		**08**
	1.1	8086CPU Architecture	
	1.2	Programmer's Model	
	1.3	Functional Pin Diagram	
	1.4	Memory Segmentation	
	1.5	Banking in 8086	
	1.6	Demultiplexing of Address/Data bus	
	1.7	Functioning of 8086 in Minimum mode and Maximum mode	
	1.8	Timing diagrams for Read and Write operations in minimum and maximum mode	
	1.9	Interrupt structure and its servicing	

2	**Instruction Set and Programming**		**06**
	2.1	Addressing Modes	
	2.2	Instruction set-Data Transfer Instructions, String Instructions, Logical Instructions, Arithmetic Instructions, Transfer of Control Instructions, Processor Control Instructions	
	2.3	Assembler Directives and Assembly Language Programming, Macros, Procedures	
3	**Memory and Peripherals interfacing**		**08**
	3.1	Memory Interfacing - RAM and ROM Decoding Techniques – Partial and Absolute	
	3.2	8255-PPI-Block diagram, CWR, operating modes, interfacing with 8086.	
	3.3	8257-DMAC-Block diagram, DMA operations and transfer modes.	
	3.4	Programmable Interrupt Controller 8259-Block Diagram, Interfacing the 8259 in single and cascaded mode.	
4	**Intel 80386DX Processor**		**07**
	4.1	Architecture of 80386 microprocessor	
	4.2	80386 registers–General purpose Registers, EFLAGS and Control registers	
	4.3	Real mode, Protected mode, virtual 8086 mode	
	4.4	80386 memory management in Protected Mode – Descriptors and selectors, descriptor tables, the memory paging mechanism	
5	**Pentium Processor**		**06**
	5.1	Pentium Architecture	
	5.2	Superscalar Operation	
	5.3	Integer & Floating-Point Pipeline Stages	
	5.4	Branch Prediction Logic	
	5.5	Cache Organization and	
	5.6	MESI protocol	
6	**Pentium 4**		**04**
	6.1	Comparative study of 8086, 80386, Pentium I, Pentium II and Pentium III	
	6.2	Pentium 4: Net burst micro architecture.	
	6.3	Instruction translation look aside buffer and branch prediction	
	6.4	Hyper threading technology and its use in Pentium 4	
		Total Hours	**39**

6.6 Suggested Learning Resources:

6.6.1 Textbooks

1. John Uffenbeck, “8086/8088 family: Design Programming and Interfacing”, PHI
2. Yu-Cheng Liu, Glenn A. Gibson, “Microcomputer System: The 8086/8088 Family, Architecture, Programming and Design”, Prentice Hall
3. Walter A. Triebel, “The 80386DX Microprocessor: hardware, Software and Interfacing”, Prentice Hall
4. Tom Shanley and Don Anderson, “Pentium Processor System Architecture”, Addison -Wesley
5. K. M. Bhurchandani and A. K. Ray, “Advanced Microprocessors and Peripherals”, McGraw Hill

6.6.2 Reference Books

1. Barry B. Brey, “Intel Microprocessors”, 8th Edition, Pearson Education India

2. Douglas Hall, “Microprocessor and Interfacing”, Tata McGraw Hill.
3. Intel Manual
4. Peter Abel, “IBM PC Assembly language and Programming”, 5th Edition, PHI
5. James Antonakons, “The Pentium Microprocessor”, Pearson Education

6.6.3 Web Resources

1. https://swayam.gov.in/nd1_noc20_ee11/preview
2. https://nptel.ac.in/courses/108/105/108105102/
3. https://www.classcentral.com/course/swayam-microprocessors-and-microcontrollers-9894
4. https://www.mooc-list.com/tags/microprocessors

7 Analysis of Algorithms Lab (CSL401)

7.1 Teaching Scheme

Course Code	Course Name	Contact Hours	Credits
CSL401	Analysis of Algorithms Lab	02	01

7.2 Examination and Evaluation Scheme

Evaluation	ISE	PE	OrE	PrE	Total
Marks	25	25	-	-	50

7.3 Course Objectives

S. No.	Objective
1	Analyze the run time and space complexity of algorithms using asymptotic analysis, recurrence relations or Master's Method.
2	Describe the divide and conquer strategy and apply it to real time problems
3	Examine and illustrate the complexity of greedy strategy and identify algorithms that employ this paradigm.
4	Demonstrate the application of dynamic programming and explain when a problem situation calls for the paradigm
5	Identify, analyze and apply backtracking, branch and bound techniques to different problems.
6	Apply string matching algorithms to real time search problems.

7.4 Course Outcomes

The students will be able to:	
CO1	*implement* basic sorting algorithms and *analyze* their complexity
CO2	*demonstrate* the working of divide and conquer strategy by *implementing* the algorithms that follow the strategy
CO3	*implement* algorithms using greedy strategy and *identify* problems that employ this paradigm.
CO4	*demonstrate* the working of dynamic programming method and *explain* when a problem situation calls for the paradigm
CO5	*implement* backtracking, branch and bound techniques to different problems.
CO6	*implement* string matching algorithms and *apply* them to real time search problems.

7.5 Course Contents

S. No.	Suggested List of Experiments
1	Implement **Insertion Sort** and **Selection Sort** algorithms and derive its time complexity.
2	Implement **Merge Sort** and **Quick Sort** algorithms and derive its time complexity.
3	Write a program to find **Single Source Shortest Path** for a directed graph using **Greedy Technique**.
4	Implement **Prims** and **Kruskal's** algorithm for finding **Minimum Cost Spanning Tree** using **Greedy Method.**
5	Write a program to solve the **0/1 Knapsack** problem using **Dynamic Programming.**
6	Write a program to find **Longest Common Subsequence** for two sequences using

	Dynamic Programming
7	Write a program to solve the **Graph Coloring** problem using **Backtracking.**
8	Write a program to solve the **Sum of Subsets** problem using **Backtracking.**
9	Write a program to perform **String matching** using **Rabin Karp Algorithm.**
10	Write a program to perform **String Matching** using **Knutt Morris Pratt Algorithm.**

Note: Lab course shall consist of minimum of 08 experiments covering the syllabus of corresponding theory course but not limited to the suggested list.

ISE: Experiments=15 marks, Mock Practical/Viva/Assignment=5marks, Attendance =5 marks
Oral & Practical exam will be based on the entire syllabus.

7.6 Suggested Learning Resources:

7.6.1 Textbooks

1. T. H. Cormen, C.E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to algorithms", 2nd Edition, PHI Publication 2005.
2. Ellis Horowitz, Sartaj Sahni, S. Rajsekaran. "Fundamentals of computer algorithms" University Press.

7.6.2 Reference Books

1. Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, "Algorithms", Tata McGraw-Hill Edition.
2. S. K. Basu, "Design Methods and Analysis of Algorithm", PHI.

7.6.3 Web Resources

1. https://nptel.ac.in/courses/106/106/106106131/
2. https://www.coursera.org/specializations/algorithms
3. https://www.coursera.org/specializations/algorithms
4. https://www.mooc-list.com/tags/algorithms

8 Database Management System Lab (CSL402)

8.1 Teaching Scheme

Course Code	Course Name	Contact Hours	Credits
CSL402	Database Management System Lab	02	01

8.2 Examination and Evaluation Scheme

Evaluation	ISE	PE	OrE	PrE	Total
Marks	25	25	-	-	50

8.3 Course Objectives

S. No.	Objective
1	To explore design and develop of relational model
2	To present SQL and procedural interfaces to SQL comprehensively
3	To introduce the concepts of transactions and transaction processing

8.4 Course Outcomes

The students will be able to:	
CO1	*design* ER /EER diagram and convert to relational model for the real world application.
CO2	*apply* DDL, DML, DCL and TCL commands.
CO3	*write* simple and complex queries.
CO4	*implement* Views and Triggers
CO5	*use* PL / SQL Constructs.
CO6	*demonstrate* the concept of concurrent transactions execution and frontend-backend connectivity

8.5 Course Contents

S. No.	Suggested List of Experiments
1	Identify the case study and detail statement of problem. Design an Entity-Relationship (ER) / Extended Entity-Relationship (EER) Model.
2	Mapping ER/EER to Relational schema model.
3	Create a database using Data Definition Language (DDL) and apply integrity constraints for the specified System
4	Apply DML Commands for the specified system
5	Perform Simple queries, string manipulation operations and aggregate functions.
6	Implement various Join operations.
7	Perform Nested and Complex queries
8	Perform DCL and TCL commands
9	Implement procedure and functions
10	Implementation of Views and Triggers.
11	Demonstrate Database connectivity
12	Implementation and demonstration of Transaction and Concurrency control techniques using locks.

Note: Lab course shall consist minimum of 08 experiments covering the syllabus of corresponding theory course but not limited to the suggested list.

ISE: Experiments=15 marks, Assignment=5, Attendance =5 marks
Oral & Practical exam will be based on the entire syllabus.

8.6 Suggested Learning Resources:

8.6.1 Textbooks

1. Korth, Slberchatz, Sudarshan, Database System Concepts, 6thEdition, McGraw Hill
2. Elmasri and Navathe, Fundamentals of Database Systems, 5thEdition, Pearson Education.
3. Raghu Ramkrishnan and Johannes Gehrke, Database Management Systems, TMH

8.6.2 Reference Books

1. Peter Rob and Carlos Coronel, Database Systems Design, Implementation and Management‖, Thomson Learning, 5thEdition.
2. Dr. P.S. Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press.
3. G. K. Gupta, Database Management Systems, McGraw Hill, 2012

8.6.3 Web Resources

1. https://nptel.ac.in/courses/106/105/106105175/
2. https://swayam.gov.in/nd1_noc19_cs46/preview
3. https://www.classcentral.com/course/swayam-database-management-system-9914
4. https://www.mooc-list.com/tags/dbms

9 Operating System Lab (CSL403)

9.1 Teaching Scheme

Course Code	Course Name	Contact Hours	Credits
CSL403	Operating System Lab	02	01

9.2 Examination and Evaluation Scheme

Evaluation	ISE	PE	OrE	PrE	Total
Marks	25	25	-	-	50

9.3 Course Objectives

S. No.	Objective
1	To gain practical experience with designing and implementing concepts of operating systems such as system calls, CPU scheduling, process management, memory management, file systems and deadlock handling using C language in Linux environment.
2	To familiarize students with the architecture of Linux OS.
3	To provide necessary skills for developing and debugging programs in Linux environment.
4	To learn programmatically to implement simple operation system mechanisms

9.4 Course Outcomes

The students will be able to:	
CO1	*demonstrate* basic Operating system Commands, Shell scripts, System Calls and API wrt Linux
CO2	*implement* various process scheduling algorithms and *evaluate* their performance.
CO3	*implement* and *analyze* concepts of synchronization and deadlocks.
CO4	*implement* various Memory Management techniques and *evaluate* their performance.
CO5	*implement* and *analyze* concepts of virtual memory.
CO6	*demonstrate* and *analyze* concepts of file management and I/O management techniques.

9.5 Course Contents

S. No.	Suggested List of Experiments
1	**Explore Linux Commands** **Explore usage of basic Linux Commands and system calls for file, directory and process management**. For eg: (mkdir, chdir, cat, ls, chown, chmod, chgrp, ps etc. system calls: open, read, write, close, getpid, setpid, getuid, getgid, getegid, geteuid. sort, grep, awk, etc.)
2	**Linux shell script** Write shell scripts to do the following: a. Display OS version, release number, kernel version b. Display top 10 processes in descending order c. Display processes with highest memory usage. d. Display current logged in user and log name.

	Display current shell, home directory, operating system type, current path setting, current working directory.
3	**Linux- API** Implement any one basic commands of linux like ls, cp, mv and others using kernel APIs.
4	**Linux- Process** a. Create a child process in Linux using the fork system call. From the child process obtain the process ID of both child and parent by using getpid and getppid system call. b. Explore wait and waitpid before termination of process.
5	**Process Management: Scheduling** a. Write a program to demonstrate the concept of non-preemptive scheduling algorithms. b. Write a program to demonstrate the concept of preemptive scheduling algorithms
6	**Process Management: Synchronization** a. Write a C program to implement solution of Producer consumer problem through Semaphore
7	**Process Management: Deadlock** a. Write a program to demonstrate the concept of deadlock avoidance through Banker's Algorithm b. Write a program demonstrate the concept of Dining Philospher's Problem
8	**Memory Management** a. Write a program to demonstrate the concept of MVT and MFT memory management techniques b. Write a program to demonstrate the concept of dynamic partitioning placement algorithms i.e. Best Fit, First Fit, Worst-Fit etc.
9	**Memory Management: Virtual Memory** a. Write a program to demonstrate the concept of demand paging for simulation of Virtual Memory implementation b. Write a program in C demonstrate the concept of page replacement policies for handling page faults eg: FIFO, LRU etc.
10	**File Management & I/O Management** a. Write a C program to simulate File allocation strategies typically sequential, indexed and linked files b. Write a C program to simulate file organization of multi-level directory structure. c. Write a program in C to do disk scheduling - FCFS, SCAN, C-SCAN

Note: Lab course shall consist minimum of 08 experiments covering the syllabus of corresponding theory course but not limited to the suggested list.

ISE: Experiments=15 marks, Assignment=5, Attendance =5 marks
Oral & Practical exam will be based on the entire syllabus.

9.6 Suggested Learning Resources:

9.6.1 Textbooks

1. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 8th Edition, 2014, ISBN-10: 0133805913 • ISBN-13: 9780133805918.
2. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating System Concepts, John Wiley & Sons, Inc., 9thEdition, 2016, ISBN 978-81-265-5427-0
3. Raghu Ram krishnan and Johannes Gehrke, Database Management Systems, TMH

9.6.2 Reference Books

1. Achyut Godbole and Atul Kahate, Operating Systems, McGraw Hill Education, 3rd Edition
2. Andrew Tannenbaum, Operating System Design and Implementation, Pearson, 3rd Edition.
3. Maurice J. Bach, “Design of UNIX Operating System”, PHI
4. Sumitabha Das, “UNIX: Concepts and Applications”, McGraw Hill, 4thEdition

9.6.3 Web Resourses

1. https://swayam.gov.in/nd1_noc19_cs50/preview
2. https://nptel.ac.in/courses/117/106/117106113/
3. https://www.classcentral.com/course/swayam-introduction-to-operating-systems-6559

10 Microprocessor Lab (CSL404)

10.1 Teaching Scheme

Course Code	Course Name	Contact Hours	Credits
CSL404	Microprocessor Lab	02	01

10.2 Examination and Evaluation Scheme

Evaluation	ISE	PE	OrE	PrE	Total
Marks	25	-	-	-	25

10.3 Course Objectives

S. No.	Objective
1	To emphasize on use of Assembly language program.
2	To prepare students for advanced subjects like embedded system and IOT.

10.4 Course Outcomes

The students will be able to:	
CO1	*use* appropriate instructions to program microprocessor to perform various task
CO2	*develop* the program in assembly/ mixed language for Intel 8086 processor
CO3	*demonstrate* the execution and debugging of assembly/ mixed language program
CO4	*use* near and far procedure in assembly language to perform various tasks.
CO5	*demonstrate* the use of a macro in an assembly language program.
CO6	*demonstrate* the use of assembly language instructions to produce various graphical effects to the text.

10.5 Course Contents

S. No.	Suggested List of Experiments
1	Use of programming tools (Debug/TASM/MASM/8086kit) to perform basic arithmetic operations on 8-bit/16-bit data
2	Code conversion (Hex to BCD and BCD to Hex)/ (ASCII to BCD and BCD to ASCII)
3	Assembly programming for 16-bit addition, subtraction, multiplication and division (menu based)
4	Assembly program based on string instructions (overlapping/non-overlapping block transfer/ string search/ string length)
5	Assembly program to display the contents of the flag register.
6	Any Mixed Language programs.
7	Assembly program to find the GCD/ LCM of two numbers
8	Assembly program to sort numbers in ascending/ descending order
9	Any program using INT 10H
10	Assembly program to find minimum/ maximum number from a given array.
11	Assembly Program to display a message in different color with blinking
12	Assembly program using procedure.
13	Assembly program using macro.
14	Program and interfacing using 8255.
15	Program and interfacing of ADC/ DAC/ Stepper motor.

Note: Lab course shall consist minimum of 08 experiments covering the syllabus of corresponding theory course but not limited to the suggested list.

ISE: Experiments=15 marks, Assignment=5, Attendance =5 marks
Oral & Practical exam will be based on the entire syllabus.

10.6 Suggested Learning Resources:

10.6.1 Textbooks

1. Yu-Cheng Liu, Glenn A. Gibson, "Microcomputer System: The 8086/8088 Family, Architecture, Programming and Design", Prentice Hall
2. Walter A. Triebel, "The 80386DX Microprocessor: hardware, Software and Interfacing", Prentice Hall
3. Tom Shanley and Don Anderson, "Pentium Processor System Architecture", AddisonWesley.
4. K. M. Bhurchandani and A. K. Ray, "Advanced Microprocessors and Peripherals", McGraw Hill

10.6.2 Reference Books

1. Barry B. Brey, "Intel Microprocessors", 8 thEdition, Pearson Education India
2. James Antonakons, "The Pentium Microprocessor", Pearson Education
3. Douglas Hall, "Microprocessor and Interfacing", Tata McGraw Hill Intel Manual

10.6.3 Web Resources

1. https://swayam.gov.in/nd1_noc20_ee11/preview
2. https://nptel.ac.in/courses/108/105/108105102/
3. https://www.classcentral.com/course/swayam-microprocessors-and-microcontrollers-9894
4. https://www.mooc-list.com/tags/microprocessors

11 Skill Based Lab Course: Python Programming (CSL405)

11.1 Teaching Scheme

Course Code	Course Name	Contact Hours	Credits
CSL405	Skill Base Lab Course: Python Programming	2*+2	02

11.2 Examination and Evaluation Scheme

Evaluation	ISE	PE	OrE	PrE	Total
Marks	25	-	-	-	25

11.3 Course Objectives

S. No.	Objective
1	Basics of Python programming
2	Decision Making, Data structure and Functions in Python
3	Object Oriented Programming using Python
4	Web framework for developing

11.4 Course Outcomes

The students will be able to:	
CO1	*understand* basic concepts in python.
CO2	*explore* contents of files, directories and text processing with python
CO3	*develop* program for data structure using built in functions in python.
CO4	*explore* django web framework for developing python-based web application.
CO5	*understand* Multithreading concepts using python.
CO6	*understand* Python's NumPy and Pandas libraries for efficient data manipulation, analysis, and computation.

11.5 Course Contents

S. No.	Detail Content	Hours
1	**Python basics** Data types in python, Operators in python, Input and Output, Control statement, Arrays in python, String and Character in python, Functions, List and Tuples, Dictionaries Exception, Introduction to OOP, Classes, Objects, Interfaces, Inheritance	**5**
2	**Advanced Python** Files in Python, Directories, Building Modules, Packages, Text Processing, Regular expression in python.	**4**
3	**Data Structure in Python** Link List, Stack, Queues, Dequeues	**3**
4	**Python Integration Primer** Graphical User interface, Networking in Python, Python database connectivity, Introduction to Django	**4**
5	**Multithreading** Thread and Process, Starting a thread, Threading module, Synchronizing threads, Multithreaded Priority Queue	**4**

6	**NumPy and Pandas** **6.1** Creating NumPy arrays, Indexing and slicing in NumPy, creating multidimensional arrays, NumPy Data types, Array Attribute, Indexing and Slicing, Creating array views copies, Manipulating array shapes I/O **6.2** Basics of Pandas, Using multilevel series, Series and Data Frames, Grouping, aggregating, Merge Data Frames	**6**

S. No.	Suggested List of Experiments
1	Exploring basics of python like data types (strings, list, array, dictionaries, set, tuples) and control statements.
2	Creating functions, classes and objects using python. Demonstrate exception handling and inheritance.
3	Exploring Files and directories 11.5.1.1.1 Python program to append data to existing file and then display the entire file 11.5.1.1.2 Python program to count number of lines, words and characters in a file. 11.5.1.1.3 Python program to display file available in current directory
4	Creating GUI with python containing widgets such as labels, textbox, radio, checkboxes and custom dialog boxes.
5	Menu driven program for data structure using built in function for link list, stack and queue.
6	Program to demonstrate CRUD (create, read, update and delete) operations on database (SQLite/ MySQL) using python.
7	Creation of simple socket for basic information exchange between server and client.
8	Creating web application using Django web framework to demonstrate functionality of user login and registration (also validating user detail using regular expression).
9	Programs on Threading using python.
10	Exploring basics of NumPy Methods.
11	Program to demonstrate use of NumPy: Array objects.
12	Program to demonstrate Data Series and Data Frames using Pandas.
13	Program to send email and read content of URL.

Note: Lab course shall consist minimum of 08 experiments covering the syllabus of corresponding theory course but not limited to the suggested list.

ISE: Experiments=10 marks, Mini Project=10 marks, Attendance =5 marks

11.6 Suggested Learning Resources:

11.6.1 Textbooks

1. Dr. R. Nageswara Rao, “Core Python Programming”, Dreamtech Press
2. Beginning Python: Using Python 2.6 and Python 3.1. James Payne, Wrox Publication
3. Anurag Gupta, G. P. Biswas, “Python Programming”, McGraw-Hill
4. E. Balagurusamy, “Introduction to computing and problem-solving using python”, McGraw Hill Education

11.6.2 Reference Books

1. Learn Python the Hard Way, 3rd Edition, Zed Shaw's Hard Way Series
2. Laura Cassell, Alan Gauld, “Python Projects”, Wrox Publication

11.6.3 Web Resources

1. "The Python Tutorial", http://docs.python.org/release/3.0.1/tutorial/
2. Beginning Perl, https://www.perl.org/books/beginning-perl/
3. http://spoken-tutorial.org
4. https://starcertification.org/Certifications/Certificate/python

12 Mini Project B (CSM401)

12.1 Teaching Scheme

Course Code	Course Name	Contact Hours	Credits
CSM401	**Mini Project B**	04	02

12.2 Examination and Evaluation Scheme

Evaluation	ISE	PE	OrE	PrE	Total
Marks	25	-	25	-	50

12.3 Course Objectives

S. No.	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 fundamentals to attempt solutions to the problems.
4	To inculcate the process of self-learning and research.

12.4 Course Outcomes

The students will be able to:	
CO1	*identify* problems based on societal /research needs.
CO2	*apply* Knowledge and skill to *solve* societal problems in a group.
CO3	*develop* interpersonal skills to work as member of a group or leader.
CO4	*draw* the proper inferences from available results through theoretical/ experimental/simulations.
CO5	*analyze* the impact of solutions in societal and environmental context for sustainable development.
CO6	*demonstrate* project management principles during project work.

12.5 Course Contents

S. No.	Guidelines for Mini Project
1	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.
2	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.
3	Students shall submit implementation plan in the form of Gantt/PERT/CPM chart, which will cover weekly activity of mini project.
4	A logbook to be prepared by each group, wherein group can record weekly work progress, guide/supervisor can verify and record notes/comments.
5	Faculty supervisor may give inputs to students during mini project activity; however, focus shall be on self-learning.
6	Students in a group shall understand problem effectively, propose multiple solution and select best possible solution in consultation with guide/ supervisor.

7	Students shall convert the best solution into working model using various components of their domain areas and demonstrate.
8	The solution to be validated with proper justification and report to be compiled in standard format of University of Mumbai.
9	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.
10	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.

In-Semester Evaluation: Distribution of marks for In-Semester Evaluation shall be done based on following:

In-Semester Evaluation		
Distribution of In-Semester Evaluation marks for both semesters shall be as below:		**Marks 25**
1	Weekly Log Report and Attendance	10
2	Mid-term Evaluation and End-Semester Evaluation with External Examiner	10
3	Quality of Project report	5

Mid-term Evaluation and End-Semester Evaluation should be done based on the performance indicators mentioned below and scaled down to 5 for each.
The final certification and acceptance of ISE ensures the satisfactory performance on the above aspects.

Oral and Practical: Oral and Practical examination (Final Project Evaluation) of Mini Project should be conducted by the panel of Internal and External examiners approved by CoE at the end of the semester.

12.6 Mini Project Evaluation parameters are as follows:

PERFORMANCE INDICATORS (PI)	Quality and clarity of the problem **GA**	Application of software Engg best practices **GA**	Objective and scope **GA**	Identification of methodologies **GA**	Synopsis and Presentation **IA**
Maximum Marks	5	5	5	5	5

GA – Group Assessment
IA – Individual Assessment

Poor	**Average**	**Good**	**Very Good**	**Excellent**
1	2	3	4	5