One-sided line papers for lab exp

and 2 sided line papers for assignment and a4 printout papers

| **Course Code** | **Course Name** | **Credits** |
| --- | --- | --- |
| **CSC401** | Engineering Mathematics-IV | **4** |

| **Module** | **Detailed Contents** | | **Hours** |
| --- | --- | --- | --- |
| 1 | **Linear Algebra (Theory of Matrices)** | | 7 |
|  | 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 | The 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** | | 7 |
|  | 2.1 | Line Integral, Cauchy‟s Integral theorem for simple connection 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) |
| 2.4 | **Self-learning Topics:** Application of Residue Theorem to evaluate real integrations. |
| 3 | **Z Transform** | | 5 |
|  | 3.1 | Definition and Region of Convergence, Transform of Standard Functions:  {����}, {�|�|}, {�+��. ��}, {��sin(��+ �)}, {��sinh ��}, {�� cosh ��}. � |  |
| 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** | | 7 |
|  | 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 the 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** | | 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 |
| 5.4 | **Self-learning Topics:** Sensitivity Analysis, Two-Phase Simplex Method, Revised Simplex Method. |
| 6 | **Nonlinear Programming Problems** | | 7 |
|  | 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 |

| **Term Work:** | | |
| --- | --- | --- |
| General Instructions: | | |
| 1 | Batch-wise tutorial have to be conducted. The number of students per batch will be as per the University pattern for practicals. | |
| 2 | Students must be encouraged to write at least 6 class tutorials on the entire syllabus. | |
| 3 | 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 will be considered as a mini project in Engineering Mathematics. This project will be graded out of 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 the entire syllabus | 10 marks |
| 3 | Mini projects | 10 marks |

| **Assessment:** |
| --- |
| **Internal Assessment Test:** |
| The assessment consists of two class tests of 20 marks each. The 1stclass test (Internal Assessment I) has to be conducted when approximately 40% of the syllabus is completed. The 2nd class test has to be conducted (Internal Assessment II) when an additional 35% syllabus is |

| completed. The duration of each test will be for one hour. | |
| --- | --- |
| **End Semester Theory Examination:** | |
| 1 | The question paper will comprise a total of 6 questions, each carrying 20 marks. |
| 2 | Out of the 6 questions, 4 questions have to be attempted. |
| 3 | Question 1, based on the entire syllabus, will have 4sub-questions of 5 marks each and is compulsory. |
| 4 | Question 2 to Question 6 will have 3 sub-questions, each of 6, 6, and 8 marks, respectively. |
| 5 | Each sub-question in (4) will be from different modules of the syllabus. |
| 6 | Weightage of each module will be proportional to the number of lecture hours, as mentioned in the syllabus. |

| **Course Code** | **Course Name** | **Credit** |
| --- | --- | --- |
| **CSC402** | Analysis of Algorithms | **3** |

| **Prerequisite:** Data structure concepts, Discrete structures | |
| --- | --- |
| **Course 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 |
| **Course Outcomes:** At the end of the course learner will be able to | |
| 1 | Analyze the running time and space complexity of algorithms. |
| 2 | Describe, apply and analyze the complexity of divide and conquer strategy. |
| 3 | Describe, apply, and analyze the complexity of greedy strategy. |
| 4 | Describe, apply and analyze the complexity of dynamic programming strategy. |
| 5 | Explain and apply backtracking, branch, and bound. |
| 6 | Explain and apply string matching techniques. |

| **Module** |  | **Detailed Contents** | **Hours** |
| --- | --- | --- | --- |
| 1 |  | **Introduction** | 8 |
|  | 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** | 6 |
|  | 2.1 | General method, Merge sort, Quicksort, Finding minimum and maximum algorithms and their Analysis, Analysis of Binary search. |  |
| 3 |  | **Greedy Method Approach** | 6 |
|  | 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** | 9 |
|  | 4.1 | General Method, Multistage graphs, Single source shortest path: Bellman-Ford Algorithm  All pair shortest path: Floyd Warshall Algorithm, Assembly-line scheduling Problem0/1 knapsack Problem, Travelling Salesperson problem, Longest common subsequence |  |
| 5 |  | **Backtracking and Branch and bound** | 6 |
|  | 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** | 4 |
|  | 6.1 | The Naïve string-matching algorithm, The Rabin Karp algorithm, The Knuth-Morris-Pratt algorithm |  |

| **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. |

| **References:** | |
| --- | --- |
| 1 | Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, “Algorithms”, Tata McGraw Hill Edition. |
| 2 | S. K. Basu, “Design Methods and Analysis of Algorithm”, PHI |

| **Assessment:** | |
| --- | --- |
| **Internal Assessment:** | |
| Assessment consists of two class tests of 20 marks each. The first-class test is to be conducted when approx. 40% syllabus is completed and the second class test when an additional 40% syllabus is completed. The duration of each test shall be one hour. | |
| **End Semester Theory Examination:** | |
| 1 | Question paper will comprise of total of six questions. |
| 2 | All question carries equal marks |
| 3 | Questions will be mixed in nature (for example supposed Q.2 has part (a) from module 3 then part (b) will be from any module other than module 3) |
| 4 | Only Four question need to be solved. |
| 5 | In question paper weightage of each module will be proportional to a number of respective lecture hours as mentioned in the syllabus. |

| **Useful Links** | |
| --- | --- |
| 1 | https://nptel.ac.in/courses/106/106/106106131/ |
| 2 | https://swayam.gov.in/nd1\_noc19\_cs47/preview |
| 3 | https://www.coursera.org/specializations/algorithms |
| 4 | https://www.mooc-list.com/tags/algorithms |

| **Course Code:** | **Course Title** | **Credit** |
| --- | --- | --- |
| **CSC403** | Database Management System | **3** |

| **Prerequisite: Data Structures** | |
| --- | --- |
| **Course 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. |
| **Course Outcomes:** | |
| 1 | Recognize the need for a database management system |
| 2 | Design ER and EER diagrams for real-life applications |
| 3 | Construct relational model and write relational algebra queries. |
| 4 | Formulate SQL queries |
| 5 | Apply the concept of normalization to relational database design. |
| 6 | Describe the concept of transaction, concurrency, and recovery. |

| **Module** |  | **Content** | **Hrs** |
| --- | --- | --- | --- |
| **1** |  | **Introduction Database Concepts** | **3** |
|  | 1.1 | Introduction, Characteristics of databases, File system v/s Database system, Data abstraction, and data Independence, DBMS system architecture, Database Administrator |  |
| **2** |  | **Entity-Relationship Data Model** | **6** |
|  | 2.1 | The Entity-Relationship (ER) Model: Entity types: Weak and strong entity sets, Entity sets, Types of Attributes, Keys, Relationship constraints: Cardinality and Participation, Extended Entity-Relationship (EER) Model: Generalization, Specialization, and Aggregation |  |
| **3** |  | **Relational Model and Relational Algebra** | **8** |
|  | 3.1 | Introduction to the Relational Model, relational schema and concept of keys. Mapping the ER and EER Model to the Relational Model, Relational Algebra-operators, Relational Algebra Queries. |  |
| **4** |  | **Structured Query Language (SQL)** | **6** |
|  | 4.1 | Overview of SQL, Data Definition Commands, Integrity constraints: key constraints, Domain Constraints, Referential integrity, check constraints, Data Manipulation commands, Data Control commands, Set and string operations, aggregate function-group by, having, Views in SQL, joins, Nested and complex queries, Triggers |  |
| **5** |  | **Relational-Database Design** | **6** |
|  | 5.1 | Pitfalls in Relational-Database designs, Concept of normalization, Function Dependencies, First Normal Form, 2NF, 3NF, BCNF. |  |
| **6** |  | **Transactions Management and Concurrency and Recovery** | **10** |
|  | 6.1 | Transaction concept, Transaction states, ACID properties, Transaction Control Commands, Concurrent Executions, Serializability-Conflict and View, Concurrency Control: Lock-based, Timestamp-based protocols, Recovery System: Log based recovery, Deadlock handling |  |

| **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 |
| **References:** | |
| 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 |

| **Assessment:** | |
| --- | --- |
| **Internal Assessment:** | |
| Assessment consists of two class tests of 20 marks each. The first-class test is to be conducted when approx. 40% syllabus is completed and the second class test when an additional 40% syllabus is completed. The duration of each test shall be one hour. | |
| **End Semester Theory Examination:** | |
| 1 | Question paper will comprise of total six questions. |
| 2 | All question carries equal marks |
| 3 | Questions will be mixed in nature (for example supposed Q.2 has part (a) from module 3 then part (b) will be from any module other than module 3) |
| 4 | Only Four question need to be solved. |
| 5 | In question paper weightage of each module will be proportional to number of respective lecture hours as mention in the syllabus. |

| **Course Code** | **Course Name** | **Credit** |
| --- | --- | --- |
| **CSC404** | Operating System | **03** |

| **Prerequisites: Data structures and Computer architecture** | |
| --- | --- |
| **Course Objectives:** | |
| 1 | 1. To introduce basic concepts and functions of operating systems. |
| 2 | 2. To understand the concept of process, thread, and resource management. |
| 3 | 3. To understand the concepts of process synchronization and deadlock. |
| 4 | 4. To understand various Memory, I/O, and File management techniques. |
| **Course Outcome:** | |
| 1 | Understand the objectives, functions, and structure of OS |
| 2 | Analyze the concept of process management and evaluate the performance of process scheduling algorithms. |
| 3 | Understand and apply the concepts of synchronization and deadlocks |
| 4 | Evaluate the performance of Memory allocation and replacement policies |
| 5 | Understand the concepts of file management. |
|  | Apply concepts of I/O management and analyze techniques of disk scheduling. |

| **Module** | **Detailed Content** | | **Hours** |
| --- | --- | --- | --- |
| **1** | **Operating system Overview** | | **4** |
|  | 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** | | **9** |
|  | 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** | | **9** |
|  | 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** | | **9** |
|  | 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** | **4** |

|  | 5.1 | Overview, File Organization, and Access, File Directories, File Sharing |  |
| --- | --- | --- | --- |
| **6** |  | **I/O management** | **4** |
|  | 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. |  |

| **Textbooks:** | |
| --- | --- |
| 1 | William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 8thEdition, 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 |
| **References:** | |
| 1 | Achyut Godbole and Atul Kahate, Operating Systems, McGraw Hill Education, 3rdEdition |
| 2 | Andrew Tannenbaum, Operating System Design and Implementation, Pearson, 3rdEdition. |
| 3 | Maurice J. Bach, “Design of UNIX Operating System”, PHI |
| 4 | Sumitabha Das, “UNIX: Concepts and Applications”, McGraw Hill, 4thEdition |

| **Assessment:** | |
| --- | --- |
| **Internal Assessment:** | |
| Assessment consists of two class tests of 20 marks each. The first class test is to be conducted when approx. 40% syllabus is completed and second class test when additional 40% syllabus is completed. Duration of each test shall be one hour. | |
| **End Semester Theory Examination:** | |
| 1 | Question paper will comprise of 6 questions, each carrying 20 marks. |
| 2 | The students need to solve total 4 questions. |
| 3 | Question No.1 will be compulsory and based on entire syllabus. |
| 4 | Remaining question (Q.2 to Q.6) will be selected from all the modules |

| **Useful Links** | |
| --- | --- |
| 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 |

| **Course Code** | **Course Name** | **Credits** |
| --- | --- | --- |
| **CSC405** | Microprocessor | **3** |

| **Module** | **Detailed Contents** | | **Hours** |
| --- | --- | --- | --- |
| 1 | **The Intel Microprocessors 8086 Architecture** | | 8 |
|  | 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** | | **6** |
|  | 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** | | **8** |
|  | **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** | | **7** |
|  | **4.1** | Architecture of 80386 microprocessor |  |
| **4.2** | 80386 registers–General-purpose Registers, EFLAGS and Control |

|  |  | register |  |
| --- | --- | --- | --- |
| **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** | | **6** |
|  | **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** | | **4** |
|  | **6.1** | Comparative study of 8086, 80386, Pentium I, Pentium II and Pentium III |  |
| **6.2** | Pentium 4: Net burst microarchitecture. |
| **6.3** | Instruction translation lookaside buffer and branch prediction |
| **6.4** | Hyper-threading technology and its use in Pentium 4 |

| **Assessment:** | |
| --- | --- |
| **Internal Assessment Test:** | |
| Assessment consists of two class tests of 20 marks each. The first class test is to be conducted when approx. 40% syllabus is completed and second class test when additional 40% syllabus is completed. Duration of each test shall be one hour. | |
| **End Semester Theory Examination:** | |
| 1 | Question paper will comprise of 6 questions, each carrying 20 marks. |
| 2 | The students need to solve total 4 questions. |
| 3 | Question No.1 will be compulsory and based on entire syllabus. |
| 4 | Remaining question (Q.2 to Q.6) will be selected from all the modules. |

| **Course Name** | **Lab Name** | **Credit** |
| --- | --- | --- |
| CSL401 | Analysis of Algorithms Lab | **1** |

| **Prerequisite: Basic knowledge of programming and data structure** | |
| --- | --- |
| **Lab Objectives:** | |
| 1 | To introduce the methods of designing and analyzing algorithms |
| 2 | Design and implement efficient algorithms for a specified application |
| 3 | Strengthen the ability to identify and apply the suitable algorithm for the given real-world problem. |
| 4 | Analyze worst-case running time of algorithms and understand fundamental algorithmic problems. |
| **Lab Outcomes:** At the end of the course, the students will be able to | |
| 1 | Implement the algorithms using different approaches. |
| 2 | Analyze the complexities of various algorithms. |
| 3 | Compare the complexity of the algorithms for specific problem. |

| **Description** | | |
| --- | --- | --- |
| Implementation can be in any language. | | |
| **Suggested Practical List:** | | |
| **Sr No** |  | **Suggested Experiment List** |
| **1** |  | **Introduction** |
|  | **1.1** | Selection sort, Insertion sort |
| **2** |  | **Divide and Conquer Approach** |
|  | **2.1** | Finding Minimum and Maximum, Merge sort, Quick sort, Binary search |
| **3** |  | **Greedy Method Approach** |
|  | **3.1** | Single source shortest path- Dijkstra  Fractional Knapsack problem  Job sequencing with deadlines  Minimum cost spanning trees-Kruskal and Prim‟s algorithm |
| **4** |  | **Dynamic Programming Approach** |
|  | **4.1** | Single source shortest path- Bellman Ford  All pair shortest path- Floyd Warshall  0/1 knapsack  Travelling salesperson problem  Longest common subsequence |
| **5** |  | **Backtracking and Branch and bound** |
|  | **5.1** | N-queen problem  Sum of subsets  Graph coloring |
| **6** |  | **String Matching Algorithms** |
|  | **6.1** | The Naïve string-matching Algorithms  The Rabin Karp algorithm  The Knuth-Morris-Pratt algorithm |

| **Term Work:** | |
| --- | --- |
| 1 | Term work should consist of 10 experiments. |
| 2 | Journal must include at least 2 assignments on content of theory and practical of “Analysis of Algorithms” |
| 3 | The final certification and acceptance of term work ensures that satisfactory performance of laboratory work and minimum passing marks in term work. |
| 4 | Total 25 Marks (Experiments: 15-marks, Attendance Theory& Practical: 05-marks, Assignments: 05-marks) |
| **Oral & Practical exam** | |
|  | Based on the entire syllabus of CSC402: Analysis of Algorithms |

| **Lab Code** | **Lab Name** | **Credit** |
| --- | --- | --- |
| **CSL402** | Database Management System Lab | **1** |

| **Prerequisite:** Discrete Structures | |
| --- | --- |
| **Lab Objectives:** | |
| 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 |
| **Lab Outcomes:** At the end of the course, the students will be able to | |
| 1 | Design ER /EER diagram and convert to relational model for the realworld application. |
| 2 | Apply DDL, DML, DCL and TCL commands |
| 3 | Write simple and complex queries |
|  | 4 UsePL / SQL Constructs. |
| 5 | Demonstrate the concept of concurrent transactions execution and frontend-backend connectivity |

| **Suggested List of Experiments** | |
| --- | --- |
| **Sr. No.** | **Title of Experiment** |
| 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. |

| **Term Work:** | |
| --- | --- |
| 1 | Term work should consist of 10 experiments. |
| 2 | Journal must include at least 2 assignments on the content of theory and practical of “Database Management System” |
| 3 | The final certification and acceptance of term work ensures that satisfactory performance of laboratory work and minimum passing marks in term work. |
| 4 | Total 25 Marks (Experiments: 15-marks, Attendance Theory& Practical: 05-marks, Assignments: 05-marks) |
| **Oral & Practical exam:** Based on the entire syllabus of CSC403: Database Management System | |

| **Course Code** | **Course Name** | **Credit** |
| --- | --- | --- |
| **CSL403** | Operating System Lab | **01** |

| **Prerequisite:** Knowledge on Operating system principles | |
| --- | --- |
| **Lab Objectives:** | |
| 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 |
| **Lab Outcomes:** At the end of the course, the students will be able to | |
| 1 | Demonstrate basic Operating system Commands, Shell scripts, System Calls and API wrt Linux |
| 2 | Implement various process scheduling algorithms and evaluate their performance. |
| 3 | Implement and analyze concepts of synchronization and deadlocks. |
| 4 | Implement various Memory Management techniques and evaluate their performance. |
| 5 | Implement and analyze concepts of virtual memory. |
| 6 | Demonstrate and analyze concepts of file management and I/O management techniques. |

| **Suggested List of Experiments** | | |
| --- | --- | --- |
| **Sr.**  **No.** |  | **Content** |
| **1** |  | **Explore Linux Commands** |
| 1.1 | **Explore the 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** |
| 2.1 | 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 the 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** |
|  | 3.1 | Implement basic commands of Linux like ls, cp, mv, and others using kernel APIs. |
| **4** |  | **Linux- Process** |
|  | 4.1 | 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** |

|  | 5.1 | 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** |
|  | 6.1 | **a.** Write a C program to implement solution of Producer consumer problem through Semaphore |
| **7** |  | **Process Management: Deadlock** |
|  | 7.1 | 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** |
|  | 8.1 | 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** |
|  | 9.1 | a. Write a program to demonstrate the concept of demand paging for simulation of Virtual Memory implementation  b. Write a program in C demonstrating the concept of page replacement policies for handling page faults eg: FIFO, LRU etc. |
| **10** |  | **File Management & I/O Management** |
|  | 10.1 | 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 |

| **Term Work:** | |
| --- | --- |
| 1 | Term work should consist of 10 experiments covering all modules. |
| 2 | Journal must include at least 2 assignments on the content of theory and practical of “Database Management System” |
| 3 | The final certification and acceptance of term work ensures that satisfactory performance of laboratory work and minimum passing marks in term work. |
| 4 | Total 25 Marks (Experiments: 15-marks, Attendance Theory& Practical: 05-marks, Assignments: 05-marks) |
| **Oral & Practical exam** | |
|  | Based on the entire syllabus of CSC405: Operating System. |

| **Lab Code** | **Lab Name** | **Credits** |
| --- | --- | --- |
| CSL404 | Microprocessor Lab | **1** |

| **Prerequisite:** Basic knowledge digital integrated circuits | |
| --- | --- |
| **Lab Objectives:** | |
| 1 | To emphasize on use of Assembly language program. |
| 2 | To prepare students for advanced subjects like embedded system and IOT. |
| **Lab Outcomes:** At the end of the course, the students will be able to | |
| 1 | Use appropriate instructions to program microprocessor to perform various task |
| 2 | Develop the program in assembly/ mixed language for Intel 8086 processor |
| 3 | Demonstrate the execution and debugging of assembly/ mixed language program |

| **Suggested List of Experiments:** | |
| --- | --- |
| **Sr.**  **No.** | **Title 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 a different color with blinking |
| 12 | Assembly program using the procedure. |
| 13 | Assembly program using macro. |
| 14 | Program and interfacing using 8255. |
| 15 | Program and interfacing of ADC/ DAC/ Stepper motor. |

| **Term Work:** | |
| --- | --- |
| 1 | Term work should consist of 10 experiments, out of theses at least one experiment on hardware interfacing. |
| 2 | Journal must include at least 2 assignments on content of theory and practical of “Microprocessor” |
| 3 | The final certification and acceptance of term work ensures that satisfactory performance of laboratory work and minimum passing marks in term work. |
| 4 | Total 25 Marks (Experiments: 15-marks, Attendance Theory& Practical: 05-marks, Assignments: 05-marks) |

| **Lab Code** | **Lab Name** | **Credit** |
| --- | --- | --- |
| **CSL405** | Skill Base Lab Course: Python Programming | **2** |

| **Prerequisite:** Knowledge of some programming languages like C, Java | |
| --- | --- |
| **Lab Objectives:** | |
| 1 | Basic of Python programming |
| 2 | Decision Making, Data structure, and Functions in Python |
| 3 | Object-Oriented Programming using Python |
| 4 | Learning Web frameworks for web development. |
| **Lab Outcomes:** At the end of the course, the students will be able to | |
| 1 | To understand basic concepts in python. |
| 2 | To explore contents of files, directories, and text processing with python |
| 3 | To develop a program for data structure using built-in functions in python. |
| 4 | To explore the Django web framework for developing python-based web applications. |
| 5 | To understand Multithreading concepts using python. |

| **Module** |  | **Detailed Content** | **Hours** |
| --- | --- | --- | --- |
| **1** |  | **Python basics** | **5** |
|  | 1.1 | 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 |  |
| **2** |  | **Advanced Python** | **4** |
|  | 2.1 | Files in Python, Directories, Building Modules, Packages, Text Processing, Regular expression in python. |  |
| **3** |  | **Data Structure in Python** | **3** |
|  | 3.1 | Link List, Stack, Queues, Dequeues |  |
| **4** |  | **Python Integration Primer** | **4** |
|  | 4.1 | GUI - Graphical User interface, Networking in Python, Python database connectivity, Introduction to Django |  |
| **5** |  | **Multithreading** | **4** |
|  | 5.1 | Thread and Process, Starting a thread, Threading module, Synchronizing threads, Multithreaded Priority Queue |  |
| **6** |  | **NumPy and Pandas** | **6** |
|  | 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 |  |

| **Suggested experiments using Python:** | |
| --- | --- |
| **Sr.**  **No.** | **Title 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  a. Python program to append data to an existing file and then display the entire file b. Python program to count a number of lines, words, and characters in a file. c. Python program to display files available in the 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 the 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 a simple socket for basic information exchange between server and client. |
| **8** | Creating web application using Django web framework to demonstrate the 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 the use of NumPy: Array objects. |
| **12** | Program to demonstrate Data Series and Data Frames using Pandas. |
| **13** | Program to send an email and read the content of the URL. |

| **Term Work:** | |
| --- | --- |
| 1 | Term work should consist of 12 experiments. |
| 2 | Journal must include at least 2 assignments |
| 3 | Mini Project based on the content of the syllabus (Group of 2-3 students) |
| 4 | The final certification and acceptance of term work ensure satisfactory performance of laboratory work and minimum passing marks in term work. |
| 5 | Total 25 Marks (Journal: 10-marks, Attendance: 05-marks, and Mini Project: 10-marks) |

| **Course Code** | **Course Name** | **Credits** |
| --- | --- | --- |
| **CSM401** | Mini Project B | **02** |

| **Objectives** | |
| --- | --- |
| 1 | To acquaint yourself 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 | acquaint yourself with the process of applying basic engineering fundamentals to attempt solutions to the problems. |
| 4 | To inculcate the process of self-learning and research. |
| **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 a member of a group or leader. |
| 4 | Draw the proper inferences from available results through theoretical/ experimental/simulations. |
| 5 | Analyze the impact of solutions in a 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 lifelong learning. |
| 9 | Demonstrate project management principles during project work. |
| **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. |

| **Term Work** | | | |
| --- | --- | --- | --- |
| The review/ progress monitoring committee shall be constituted by the head of departments of each institute. The progress of the mini-project is to be evaluated on a continuous basis, minimum of two reviews in each semester. | | | |
| In continuous assessment focus shall also be on each individual student, assessment based on individual‟s contribution in a group activity, their understanding and response to questions. | | | |
| **Distribution of Term work marks for both semesters shall be as below:** | | | **Marks** |
| 1 | Marks awarded by guide/supervisor based on logbook | | 10 |
| 2 | Marks awarded by review committee | | 10 |
| 3 | 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:** | | | |
| 1 | | In the first semester the entire theoretical solution shall be ready, including components/system selection and cost analysis. Two reviews will be conducted based on the presentation given by the student's group.  ∙ First shall be for finalization of the problem  ∙ Second shall be on finalization of the proposed solution of the problem. | |
| 2 | | In the second semester expected work shall be the procurement of components/systems, the building of a working prototype, testing and validation of results based on work completed in an earlier semester.  ∙ The first review is based on the readiness of building a working prototype to be conducted.  ∙ The second review shall be based on poster presentation cum demonstration of a working model in the last month of the said semester. | |
| **Half-year project:** | | | |
| 1 | | In this case in one semester students‟ group shall complete the project in all aspects including, ∙ Identification of need/problem  ∙ Proposed final solution  ∙ Procurement of components/systems  ∙ Building prototype and testing | |
| 2 | | Two reviews will be conducted for continuous assessment,  ∙ First shall be for finalization of the problem and proposed solution  ∙ Second shall be for implementation and testing of solutions. | |
| **Assessment criteria of Mini Project.** | | | |
| **Mini Project** shall be assessed based on the following criteria; | | | |
| 1 | | Quality of survey/needs 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 impacts | |
| 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 the performance of students in mini-project. |
|  | In the case of a **half-year project** all criteria‟s in generic may be considered for evaluation of the performance of students in mini-project. |
| **Guidelines for Assessment of Mini Project Practical/Oral Examination:** | |
| 1 | Report should be prepared as per the guidelines issued by the University of Mumbai. |
| 2 | Mini Project shall be assessed through a presentation and demonstration of the working model by the student project group to a panel of Internal and External Examiners preferably from industry or research organizations having experience of more than five years approved by the head of Institution. |
| 3 | Students shall be motivated to publish a paper based on the work in Conferences/students competitions. |
| **Mini Project** shall be assessed based on the 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 |