# NARAYANA ENGINEERING COLLEGE::GUDUR

AUTONOMOUS

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

## B.Tech – CSE - Course Structure, w.e.f AY:2024-25

# DEPARTMENT VISION & MISSION

## VISION OF THE DEPARTMENT

- To produce globally competent software professionals in the field of computer science and engineering to meet the needs of industry and society along with research and consultancy, lifelong learning, leadership qualities and ethics.

## MISSION OF THE DEPARTMENT

- To deliver quality technical education by practicing innovative teaching learning processes making student's self-sufficient individuals
- To inculcate innovative thinking and problem solving skills in learners through training programs and collaborative interaction with industry.
- To develop professional behaviour with strong ethical values, leadership qualitiesand lifelong learning by providing value based education

# PEOs, Pos, PSOs

## PEOs

**PEO 1**: To attain higher position in career by exhibiting expertise in solving real world problems.

**PEO 2**: Fill technical gaps and take leadership roles and achieve substantive results for the development of organization.

**PEO 3**: Adapt to rapidly changing technologies through lifelong learning.

## POs

**1. Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
**2. Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
**3. Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
**4. Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
**5. Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
**6. The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
**7. Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
**8. Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
**9. Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
**10. Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

**11. Project management and finance**: Demonstrate knowledge and understanding of

the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

**12: Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technologicalchange.

## PSOs

**PSO 1:** Software Product Development: Apply the principles and practices of software Engineeringfor developing quality software applications

**PSO 2:** Employment: Get employed in industries through their knowledge attained in Basic and advanced programming languages, specialized software packages or become an entrepreneur.

## SEMESTER –III

| Course Code | Category | Course Title | Contact Periods per week | | | | Credits | Scheme of Examination Max. Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | Total | | Int. Marks | Ext. Marks | Total marks |
| 23A54301 | BS&H | Discrete Mathematics & Graph Theory | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| 23A52301 | BS&H | Universal Human Values Understanding Harmony andEthical human conduct | 2 | 1 | 0 | 3 | 3 | 30 | 70 | 100 |
| 23A30402 | ES | Digital Logic and Computer Organization | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| 23A05302T | PC | Advanced Data Structures & Algorithms Analysis | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| 23A05303T | PC | Object – Oriented Programming Through JAVA | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| 23A05302P | PC | Advanced Data Structures and Algorithms Analysis Lab | 0 | 0 | 3 | 3 | 1.5 | 30 | 70 | 100 |
| 23A05303P | PC | Object – Oriented Programming Through JAVA Lab | 0 | 0 | 3 | 3 | 1.5 | 30 | 70 | 100 |
| 23A05304 | SE | Python programming | 0 | 1 | 2 | 3 | 2 | 30 | 70 | 100 |
| 23A99301 | AC | Environmental Science | 2 | 0 | 0 | 2 | 0 | | 100 | 100 |
| | SC | Career competency Development I | 0 | 0 | 2 | 2 | 0 | 30 | 70 | 100 |
| | SC | Value added course/Certificate course I | 0 | 0 | 0 | 0 | 0 | 30 | 70 | 100 |
| | | Counseling/Mentoring | 0 | 0 | 1 | 1 | 0 | -- | -- | -- |
| | | Sports/Hobby Clubs/Activities | 0 | 0 | 2 | 2 | 0 | -- | -- | -- |
| | | Activity Point Programme | During the Semester | | | | 20 Pts | | | |
| | | | **16** | **2** | **13** | **31** | **20** | **300** | **800** | **1100** |

## SEMESTER –IV

| Course Code | Category | Course Title | Contact Periods per week | | | | Credits | Scheme of Examination Max. Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | Total | | Int. Marks | Ext. Marks | Total marks |
| 23A52402a | MC | Managerial Economics and Financial Analysis | 2 | 0 | 0 | 2 | 2 | 30 | 70 | 100 |
| 23A54401 | ES | Probability & Statistics | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| 23A35401T | PC | Operating Systems | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| 23A05402T | PC | Database Management Systems | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| 23A05403 | PC | Software Engineering | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| 23A35401P | PC | Operating Systems Lab | 0 | 0 | 3 | 3 | 1.5 | 30 | 70 | 100 |
| 23A05402P | PC | Database Management Systems Lab | 0 | 0 | 3 | 3 | 1.5 | 30 | 70 | 100 |
| 23A52401 | SE | FullStackDevelopment-1 | 0 | 1 | 2 | 3 | 2 | 30 | 70 | 100 |
| 23A99401 | BS&H | Design Thinking & Innovation | 1 | 0 | 2 | 3 | 2 | 30 | 70 | 100 |
| | SC | Career competency Development I | 0 | 0 | 2 | 2 | 0 | 30 | 70 | 100 |
| | SC | IndustryOriented Course I | 0 | 0 | 0 | 0 | 0 | 30 | 70 | 100 |
| | | Counseling/Mentoring | 0 | 0 | 1 | 1 | 0 | -- | -- | -- |
| | | Sports/Hobby Clubs/Activities | 0 | 0 | 2 | 2 | 0 | -- | -- | -- |
| | | Activity Point Programme | During the Semester | | | | 20 Pts | | | |
| | | | **15** | **1** | **15** | **31** | **21** | **330** | **770** | **1100** |

**Mandatory: Community Service Project Internship of 08 weeks duration during summer vacation**

## SEMESTER –V

| Course Code | Category | Course Title | Contact Periods per week | | | | Credits | Scheme of Examination Max. Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | Total | | Int. Marks | Ext. Marks | Total marks |
| | PC | MachineLearning | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| | PC | Computer Networks | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| | PC | Formal Languages and Automata Theory | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| PE-I | PE | OOAD / AI / MPMC /DWDM/ 12 Week MOOCs | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| | OE | Open Elective-I | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| | PC | MachineLearning Lab | 0 | 0 | 3 | 3 | 1.5 | 30 | 70 | 100 |
| | PC | Computer Networks Lab | 0 | 0 | 3 | 3 | 1.5 | 30 | 70 | 100 |
| | SE | Full Stack development - II | 0 | 1 | 2 | 3 | 2 | 30 | 70 | 100 |
| | ES | Tinkering Lab | 0 | 0 | 2 | 2 | 1 | 30 | 70 | 100 |
| **Community Service Internship** | Internship | Evaluation of Community Service Internship | - | - | - | - | 2 | -- | -- | -- |
| | SC | Career competency Development III | 0 | 0 | 2 | 2 | 0 | 30 | 70 | 100 |
| | SC | Value added course/Certificate course II | 0 | 0 | 0 | 0 | 0 | 30 | 70 | 100 |
| | | Counseling/Mentoring | 0 | 0 | 1 | 1 | 0 | -- | -- | -- |
| | | Sports/Hobby Clubs/Activities | 0 | 0 | 2 | 2 | 0 | -- | -- | -- |
| | | Activity Point Programme | During the Semester | | | | 20 Pts | | | |
| | | | **15** | **1** | **15** | **31** | **23** | **330** | **770** | **1100** |

## SEMESTER –VI

| Course Code | Category | Course Title | Contact Periods per week | | | | Credits | Scheme of Examination Max. Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | Total | | Int. Marks | Ext. Marks | Total marks |
| | PC | Compiler Design | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| | PC | Cloud Computing | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| | PC | Cryptography and Network Security | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| PE-II | PE | Software TM /Cyber Security /Devops /Embedded Systems/ 12 Week MOOCs | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| PE-III | PE | Software Project Management /Mobile Adhoc Networks /Natural Language Processing / Distributed Operating System / 12 Week MOOCs | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| | OE | Open Elective-II | 3 | 0 | 3 | 3 | 3 | 30 | 70 | 100 |
| | PC | Cloud Computing Lab | 0 | 0 | 3 | 3 | 1.5 | 30 | 70 | 100 |
| | PC | Cryptography and Network Security Lab | 0 | 0 | 3 | 3 | 1.5 | 30 | 70 | 100 |
| | SE | Soft skills (or) IELTS | 0 | 1 | 2 | 3 | 2 | 30 | 70 | 100 |
| | AC | TechnicalPaper Writing & IPR | 2 | 0 | 0 | 2 | 0 | -- | 100 | 100 |
| | SC | Career competency Development IV | 0 | 0 | 2 | 2 | 0 | 30 | 70 | 100 |
| | SC | IndustryOriented Course II | 0 | 0 | 0 | 0 | 0 | 30 | 70 | 100 |
| | | Counseling/Mentoring | 0 | 0 | 1 | 1 | 0 | -- | -- | -- |
| | | Sports/Hobby Clubs/Activities | 0 | 0 | 2 | 2 | 0 | -- | -- | -- |
| | | Activity Point Programme | During the Semester | | | | 20 Pts | | | |
| | | | **20** | **1** | **13** | **34** | **23** | **330** | **870** | **1200** |

**Mandatory: Community Service Project Internship of 08 weeks duration during summer vacation**

## SEMESTER –VII

| Course Code | Category | Course Title | Contact Periods per week | | | | Credits | Scheme of Examination Max. Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | Total | | Int. Marks | Ext. Marks | Total marks |
| | PC | Deep Learning | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| | MC | Human Resource Management | 2 | 0 | 0 | 2 | 2 | 30 | 70 | 100 |
| PE-IV | PE | Software Architecture & Design Pattern /Block chain Technology /AR&VR / IoT/12 Week MOOCs | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| PE-V | PE | Agile methodologies / Metaverse /Computer Vision / Cyber Physical Systems / 12 Week MOOCs | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| | OE | Open Elective-III | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| | OE | Open Elective-IV | 3 | 0 | 0 | 3 | 3 | 30 | 70 | 100 |
| | SE | Prompt Engineering | 0 | 1 | 2 | 3 | 2 | 30 | 70 | 100 |
| | AC | Gender Sensitization | 2 | 0 | 0 | 2 | - | | 100 | 100 |
| | IS | Evaluation of IndustryInternship | - | - | - | - | 2 | | | |
| | SC | Career competency Development V | 0 | 0 | 2 | 2 | 0 | 30 | 70 | 100 |
| | SC | Skill Development Training | 0 | 0 | 0 | 0 | 0 | 30 | 70 | 100 |
| | | Counseling/Mentoring | 0 | 0 | 1 | 1 | 0 | -- | -- | -- |
| | | Sports/Hobby Clubs/Activities | 0 | 0 | 2 | 2 | 0 | -- | -- | -- |
| | | Activity Point Programme | During the Semester | | | | 20 Pts | | | |
| | | | **19** | **1** | **7** | **27** | **21** | **270** | **730** | **1000** |

## SEMESTER –VIII

| Course Code | Category | Course Title | Contact Periods per week | | | | Credits | Scheme of Examination Max. Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | Total | | Int. Marks | Ext. Marks | Total marks |
| | PR | Full semester Internship & Project Work | 0 | 0 | 24 | 24 | 12 | 60 | 140 | 200 |
| | | | **0** | **0** | **24** | **24** | **12** | **60** | **140** | **200** |

| NARAYANAENGINEERINGCOLLEGE:GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **III Sem** | **DISCRETE MATHEMATICS & GRAPH THEORY** | | | | | | | R23 |
| Course Code | Hours/Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A54301 | 3 | 0 | 0 | 48 | 3 | 30 | 70 | 100 |

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Apply mathematical logic to solve problems.(BL1, BL3) |
| **CO 2** | Understand the concepts and perform the operations related to sets, relations and functions.<br>Gain the conceptual background needed and identify structures of algebraic nature. (BL3, BL5) |
| **CO 3** | Apply basic counting techniques to solve combinatorial problems. (BL3) |
| **CO 4** | Formulate problems and solve recurrence relations. (BL2, BL3) |
| **CO 5** | Apply Graph Theory in solving computer science problems. (BL3, BL5) |

| COURSECONTENT | | |
|---|---|---|
| **MODULE– 1** | **Mathematical Logic** | **9H** |
| Introduction, Statements and Notation, Connectives, Well-formed formulas, Tautology, Duality law, Equivalence, Implication, Normal Forms, Functionally complete set of connectives, Inference Theory of Statement Calculus, Predicate Calculus, Inference theory of Predicate Calculus. | | |
| **MODULE– 2** | **Set Theory** | **10H** |
| The Principle of Inclusion- Exclusion, Pigeon hole principle and its application, Functions composition of functions, Inverse Functions, Recursive Functions, Lattices and its properties. Algebraic structures: Algebraic systems-Examples and General Properties, Semi groups and Monoids, groups, sub groups, homomorphism, Isomorphism. | | |
| **MODULE– 3** | **Elementary Combinatorics** | **10H** |
| Combinations and Permutations, Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with Repetitions, Enumerating Permutations with Constrained Repetitions, Binomial Coefficients, The Binomial and Multinomial Theorems. | | |
| **MODULE– 4** | **Recurrence Relations** | **10H** |
| Generating Functions of Sequences, Calculating Coefficients of Generating Functions, Recurrence relations, Solving Recurrence Relations by Substitution and Generating functions, The Method of Characteristic roots, Solutions of Inhomogeneous, Recurrence Relations. | | |
| **MODULE– 5** | **Graphs** | **9H** |
| Basic Concepts, Isomorphism and Sub graphs, Trees and their Properties, Spanning Trees, Directed Trees, Binary Trees, Planar Graphs, Euler's Formula, Multi graphs and Euler Circuits, Hamiltonian Graphs. | | |
| | **Total hours:** | **48hours** |

**TEXTBOOK:**

1. J. P. Tremblay and R. Manohar, Discrete Mathematical Structures with Applications to Computer Science, Tata McGraw Hill, 2002.

2. Kenneth H. Rosen, Discrete Mathematics and its Applications with Combinatorics and Graph Theory, 7th Edition, McGraw Hill Education (India) Private Limited.

**REFERENCES:**

1. JoeL.Mott, Abraham Kandel and Theodore P. Baker, Discrete Mathematics for Computer Scientists & Mathematicians, 2nd Edition, Pearson Education.

2. Narsingh Deo, Graph Theory with Applications to Engineering and Computer Science.

Online Learning Resources:

1. http://www.cs.yale.edu/homes/aspnes/classes/202/notes.pdf

| NARAYANAENGINEERINGCOLLEGE:GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| III Sem | DIGITAL LOGIC & COMPUTER ORGANIZATION | | | | | | | R23 |
| Course Code | Hours/Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A30402 | 3 | 0 | 0 | 48 | 3 | 30 | 70 | 100 |

**Course Objectives:**

The main objective of the course is to

- providestudentswithacomprehensiveunderstandingofdigitallogicdesign principles and computer organization fundamentals
- Describe memory hierarchy concepts
- Explain input/output (I/O) systems and their interaction with the CPU, memory, and peripheral devices

**CourseOutcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Differentiate between combinational and sequential circuits based on their characteristics and functionalities. **(BL2)** |
| **CO 2** | Demonstrate an understanding of computer functional units.**(BL2)** |
| **CO 3** | Analyze the design and operation of processors, including instruction execution, pipelining, and control unit mechanisms, to comprehend their role in computer systems.**(BL3)** |
| **CO 4** | Describe memory hierarchy concepts, including cache memory, virtual memory, and secondary storage, and evaluate their impact on system performance and scalability. **(BL3)** |
| **CO 5** | Explain input/output (I/O) systems and their interaction with the CPU, memory, and peripheral devices, including interrupts, DMA, and I/O mapping techniques. **(BL3)** |

| COURSECONTENT | | |
|---|---|---|
| **MODULE– 1** | | **9H** |
| **Data Representation:** Binary Numbers, Fixed Point Representation. Floating Point Representation. Number base conversions, Octal and Hexadecimal Numbers, components, Signed binary numbers, Binary codes<br>**Digital Logic Circuits-I:** Basic Logic Functions, Logic gates, universal logic gates, Minimization of Logic expressions. K-Map Simplification, Combinational Circuits, Decoders, Multiplexers | | |
| **MODULE– 2** | | **10H** |
| **Digital Logic Circuits-II**: Sequential Circuits, Flip-Flops, Binary counters, Registers, Shift Registers, Ripple counters<br>**Basic Structure of Computers:** Computer Types, Functional units, Basic operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers, Computer Generations, Von- Neumann Architecture | | |

| | | |
|---|---|---|
| **MODULE– 3** | | **10H** |
| **Computer Arithmetic** : Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed-operand Multiplication, Fast Multiplication, Integer Division, Floating-Point Numbers and Operations<br>**Processor Organization:** Fundamental Concepts, Execution of a Complete Instruction, Multiple-Bus Organization, Hardwired Control and Multi programmed Control | | |
| **MODULE– 4** | | **10H** |
| **The Memory Organization:** Basic Concepts, Semiconductor RAM Memories, Read-Only Memories, Speed, Size and Cost, Cache Memories, Performance Considerations, Virtual Memories, Memory Management Requirements, Secondary Storage | | |
| **MODULE– 5** | | **9H** |
| **Input /Output Organization:** Accessing I/O Devices, Interrupts, Processor Examples, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces | | |
| | **Totalhours:** | **48hours** |

**Textbooks:**

1. Computer Organization, CarlHamacher, ZvonkoVranesic, SafwatZaky, 6thedition, McGraw Hill, 2023.
2. Digital Design, 6thEdition, M. MorrisMano, PearsonEducation,2018.

ComputerOrganizationandArchitecture,WilliamStallings,11thEdition,Pearson, 2022.

**Reference Books:**

1. Computer Systems Architecture, M. MorisMano,3rdEdition, Pearson, 2017.
2. Computer Organization and Design, David A. Paterson, John L. Hennessy, Elsevier, 2004.
3. Fundamentals of Logic Design, Roth, 5thEdition, Thomson, 2003.

**Online Learning Resources:**

https://nptel.ac.in/courses/106/103/106103068/

| NARAYANAENGINEERINGCOLLEGE:GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **III Sem** | **ADVANCED DATA STRUCTURES & ALGORITHM ANALYSIS** | | | | | | | **R23** |
| Course Code | Hours/Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A05302T | 3 | 0 | 0 | 48 | 3 | 30 | 70 | 100 |

**Course Objectives:** The main objective of the course is to

- Provide knowledge on advance data structures frequently used in Computer Science domain.
- Develop skills in algorithm design techniques popularly used.
- Understand the use of various data structures in the algorithm design.

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Illustrate the working of the advanced tree data structures and their applications.**(BL2)** |
| **CO 2** | Understand the Graph data structure, traversals and apply them in various contexts**. (BL2)** |
| **CO 3** | Use various data structures in the design of algorithms. **(BL3)** |
| **CO 4** | Recommend appropriate data structures based on the problem being solved. **(BL5)** |
| **CO 5** | Analyze algorithms with respect to space and time complexities. **(BL4)** Design new algorithms. **(BL6)** |

| COURSECONTENT | | |
|---|---|---|
| **MODULE– 1** | | **9H** |
| Introduction to Algorithm Analysis, Space and Time Complexity analysis, Asymptotic Notations.<br>AVL Trees–Creation, Insertion, Deletion operations and Applications.<br>Heap Trees (Priority Queues)–Min and Max Heaps, Operations and Applications. | | |
| **MODULE– 2** | | **10H** |
| **Graphs**: Terminology, Representations, Basic Search and Traversals, Connected Components and Biconnected Components, applications<br>**Divide and Conquer:** The General Method, Quick Sort, Merge Sort, Strassen's matrix multiplication, Convex Hull | | |
| **MODULE– 3** | | **10H** |
| **Greedy Method:** General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum cost spanning trees.<br>**Single Source Shortest Paths** –Dijkstra algorithm, General Weights (Bellman Ford Algorithm), Optimal Binary Search Trees.<br>**Dynamic Programming**: General Method, All pairs shortest paths | | |

| MODULE– 4 | | 10H |
|---|---|---|
| **Dynamic programming**: 0/1 Knapsack, String Editing, Travelling Salesperson problem.<br>**Backtracking**: General Method, 8-Queens Problem, Sum of Subsets problem, Graph Coloring, 0/1 Knapsack Problem | | |
| **MODULE– 5** | | **9H** |
| **Branch and Bound**: The General Method, 0/1 Knapsack Problem, Travelling Salesperson problem.<br>**NP Hard and NP Complete Problems**: Basic Concepts, Cook's theorem<br>**NP Hard Graph Problems**: Clique Decision Problem (CDP), Chromatic Number Decision Problem (CNDP), Traveling Salesperson Decision Problem (TSP)<br>**NP Hard Scheduling Problems**: Job Shop Scheduling | | |
| | **Total hours:** | **48hours** |

**Textbooks:**

1. Fundamentals of Data Structures in C, Ellis Horowitz, Sartaj Sahni, Susan Anderson – Freed, 2nd Edition, Universities Press
2. Computer Algorithms, Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, 2nd Edition, Silicon Press

**Reference Books:**

1. Data Structures and program designing C, Robert Kruse, Pearson Education Asia
2. An introduction to Data Structures with applications, Trembley & Sorenson, Mc Graw Hill
3. The Art of Computer Programming, Vol.1: Fundamental Algorithms, Donald E Knuth, Addison-Wesley, 1997.
4. Data Structures using C&C++: Langsam, Augenstein & Tanenbaum, Pearson,1995
5. Algorithms + Data Structures & Programs: N. Wirth, PHI
6. Fundamentals of Data Structures in C++: Horowitz Sahni & Mehta, GalgottiaPub.
7. Data structures in Java: Thomas Standish, Pearson Education Asia

**Online Learning Resources:**

1. https://www.tutorialspoint.com/advanced_data_structures/index.asp
2. http://peterindia.net/Algorithms.html
3. Abdul Bari,1. Introduction to Algorithms(youtube.com)

| NARAYANA ENGINEERING COLLEGE ::GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| III Sem | OBJECT – ORIENTED PROGRAMMING THROUGH JAVA | | | | | | | R23 |
| Course Code | Hours/Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A05303T | 3 | 0 | 0 | 48 | 3 | 30 | 70 | 100 |

**Course Objectives:** The learning objectives of this course are to:

- Identify Java language components and how they work together in applications
- Learn the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
- Learn how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications
- Understand how to design applications with threads in Java
- Understand how to use Java ap is for program development

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Analyze problems, design solutions using OOP principles, and implement them efficiently in Java. (BL4) |
| **CO 2** | Design and implement classes to model real-world entities, with a focus on attributes, behaviors, and relationships between objects (BL4) |
| **CO 3** | Demonstrate an understanding of inheritance hierarchies and polymorphic behaviour, including method overriding and dynamic method dispatch. (BL3) |
| **CO 4** | Apply Competence in handling exceptions and errors to write robust and fault-tolerant code. (BL3) |
| **CO 5** | Perform file input/output operations, including reading from and writing to files using Java I/O classes, graphical user interface (GUI) programming using Java FX. (BL3) |
| **CO6** | Choose appropriate data structure of Java to solve a problem(BL6) |

## COURSECONTENT

| **MODULE– 1** | **9H** |
|---|---|

**Object Oriented Programming:** Basic concepts, Principles, Program Structure in Java: Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments, Programming Style.

**Data Types**, Variables, and Operators :Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output with printf() Method, Static Variables and Methods, Attribute Final, **Introduction to Operators**, Precedence and Associativity of Operators, Assignment Operator ( = ), Basic Arithmetic Operators, Increment (++) and Decrement (- -) Operators, Ternary Operator, Relational Operators, Boolean Logical Operators, Bitwise Logical Operators.

**Control Statements**: Introduction, if Expression, Nested if Expressions, if–else Expressions, Ternary Operator?:, Switch Statement, Iteration Statements, while Expression, do–while Loop, for Loop, Nested for Loop, For–Each for Loop, Break Statement, Continue Statement.

| **MODULE– 2** | **10H** |
|---|---|

**Classes and Objects:** Introduction, Class Declaration and Modifiers, Class Members, Declaration of Class Objects, Assigning One Object to Another, Access Control for Class Members, Accessing Private Members of Class, Constructor Methods for Class, Overloaded Constructor Methods, Nested Classes, Final Class and Methods, Passing Arguments by Value and by Reference, Keyword this.

**Methods:** Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods, Attributes Final and Static.

| **MODULE– 3** | | **10H** |
|---|---|---|

**Arrays:**Introduction,DeclarationandInitializationofArrays,StorageofArrayin ComputerMemory,AccessingElementsofArrays,OperationsonArrayElements,AssigningArray to Another Array, Dynamic Change of Array Size, Sorting of Arrays, Search for Values in Arrays, Class Arrays, Two-dimensional Arrays, Arrays of Varying Lengths, Three-dimensional Arrays, Arrays as Vectors.

**Inheritance:** Introduction, Process of Inheritance, Types of Inheritances, Universal Super Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Inheritance.

**Interfaces:** Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, Default Methods in Interfaces, Static Methods in Interface, Functional Interfaces, Annotations.

| **MODULE– 4** | | **10H** |
|---|---|---|

**Packages and Java Library:** Introduction, Defining Package, Importing Packages and Classes into Programs, Path and Class Path, Access Control, Packages in Java SE, Java.lang Package and its Classes, Class Object, Enumeration, class Math, Wrapper Classes, Auto-boxing and Auto- unboxing, Java util Classes and Interfaces, Formatter Class, Random Class, Time Package, Class Instant (java.time.Instant), Formatting for Date/Time in Java, Temporal Adjusters Class, Temporal Adjusters Class.

**Exception Handling:** Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exceptions.

| **MODULE– 5** | | **9H** |
|---|---|---|

**String Handling in Java:** Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Comparison, Modifying, Searching; Class String Buffer.

**Multithreaded Programming:** Introduction, Need for Multiple Threads Multithreaded Programming for Multi-core Processor, Thread Class, Main Thread-Creation of New Threads, Thread States, Thread Priority-Synchronization, Deadlock and Race Situations, Inter-thread Communication - Suspending, Resuming, and Stopping of Threads.

| | **Totalhours:** | **48hours** |
|---|---|---|

**Text Books:**

1. JAVA on estepahead, AnithaSeth, B.L.Juneja, Oxford.
2. Joy with JAVA, Fundamentals of Object Oriented Programming, DebasisSamanta, MonalisaSarma, Cambridge, 2023.
3. JAVA9for Programmers, PaulDeitel, HarveyDeitel,4thEdition, Pearson.

**References Books:**

1. The complete Reference Java, 11thedition, HerbertSchildt, TMH
2. Introduction to Java programming,7thEdition,YDanielLiang, Pearson

**Online Resources:**

1. https://nptel.ac.in/courses/106/105/106105191/
2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012880464547618816347_shared/overview

| III Sem | ADVANCED DATA STRUCTURES & ALGORITHM ANALYSIS LAB | | | | | | | R23 |
|---|---|---|---|---|---|---|---|---|
| Course Code | Hours/Week | | | Total hrs | Credits | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A05302P | 0 | 0 | 3 | 48 | 1.5 | 30 | 70 | 100 |

**Course Objectives:** The objective of the course is to

- acquire practical skills in constructing and managing Data structures
- apply the popular algorithm design methods in problem-solving scenarios

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Design and develop programs to solve real world problems with the popular algorithm design methods. **(BL5)** |
| **CO 2** | Demonstrate an understanding of Non-Linear data structures by developing implementing the operations on AVL Trees, B-Trees, Heaps and Graphs**.(BL2)** |
| **CO 3** | Critically assess the design choices and implementation strategies of algorithms and data structures in complex applications. (BL5) |
| **CO 4** | Utilize appropriate data structures and algorithms to optimize solutions for specific computational problems. **(BL3)** |
| **CO 5** | Compare the performance of different of algorithm design strategies. **(BL4)**<br>Design algorithms to new real world problems. **(BL6)** |

**Experiments covering the Topics:**

- Operations on AVL trees, B-Trees, Heap Trees
- Graph Traversals
- Sorting techniques
- Minimum cost spanning trees
- Shortest path algorithms
- 0/1KnapsackProblem
- Travelling Sales person problem
- Optimal Binary Search Trees
- N-Queens Problem
- Job Sequencing

| COURSE CONTENT | CO |
|---|---|
| **List of Experiments** | |
| **TASK–1** | **CO1** |
| Construct an AVL tree for a given set of elements which are stored in a file. And implement insert and delete operation on the constructed tree. Write contents of tree into a new file using in-order. | |
| **TASK–2** | **CO1** |
| Construct Min and Max Heap using arrays, delete any element and display the content of the Heap. | |
| **TASK–3** | **CO1** |

| | |
|---|---|
| Implement BFT and DFT for given graph, when graph is represented by<br>a) Adjacency Matrix b)Adjacency Lists | |
| **TASK–4** | **CO2** |
| Write a program for finding the bi-connected components in a given graph. | |
| **TASK–5** | **CO2** |
| Implement Quick sort and Merge sort and observe the execution time for various input sizes (Average, Worst and Best cases). | |
| **TASK–6** | **CO2** |
| Compare the performance of Single Source Shortest Paths using Greedy method when the graph is represented by adjacency matrix and adjacency lists. | |
| **TASK–7** | **CO3** |
| Implement Job sequencing with deadlines using Greedy strategy. | |
| **TASK–8** | **CO4** |
| Write a program to solve 0/1 Knapsack problem Using Dynamic Programming. | |
| **TASK–9** | **CO5** |
| Implement N-Queens Problem Using Backtracking. | |
| **TASK–10** | **CO5** |
| Use Backtracking strategy to solve 0/1Knapsack problem. | |
| **TASK–11** | **CO5** |
| Implement Travelling Sales Person problem using Branch and Bound approach. | |

**Reference Books:**


1. Fundamentals of Data Structures in C++, Horowitz Ellis, SahniSartaj, Mehta, Dinesh, 2nd Edition, Universities Press
2. ComputerAlgorithms/C++EllisHorowitz,SartajSahni,SanguthevarRajasekaran, 2nd Edition, University Press
3. DataStructuresandprogramdesigninC,RobertKruse,PearsonEducationAsia
4. An introduction to Data Structures with applications, Trembley & Sorenson, McGraw Hill


**Online Learning Resources:**

1. http://cse01-iiith.vlabs.ac.in/
2. http://peterindia.net/Algorithms.html

| NARAYANAENGINEERINGCOLLEGE ::GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| III Sem. | OBJECT - ORIENTED PROGRAMMING THROUGH JAVA LAB | | | | | | | R23 |
| Course Code | Hours/Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A05303P | 0 | 0 | 3 | 48 | 1.5 | 30 | 70 | 100 |

**Course Objectives:** The aim of this course is to:

- Practice object – oriented programming in the Java programming language
- Implement Classes, Objects, Methods, Inheritance, Exception, Runtime Polymorphism, User defined Exception handling mechanism
- Illustrate inheritance, Exception handling mechanism, JDBC connectivity
- Construct Threads, Event Handling, Implement packages

**Course Outcomes**:After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Demonstrate a solid understanding of Java syntax, including data types, control structures, methods, classes, objects, inheritance, polymorphism, and exception handling. **(BL2)** |
| **CO 2** | Apply fundamental OOP principles such as encapsulation, inheritance, polymorphism, and abstraction to solve programming problems effectively**.(BL3)** |
| **CO 3** | Familiar with commonly used Java libraries and APIs, including the Collections Framework, Java I/O, JDBC, and other utility classes. **(BL2)** |
| **CO 4** | Develop problem-solving skills and algorithmic thinking, applying OOP concepts to design efficient solutions to various programming challenges. **(BL3)** |
| **CO 5** | Proficiently construct Multithreaded programming, IO Files**(BL4)** |
| **CO6** | Develop new programs for solving typical computer science problems**(BL6)** |

**Experiments covering the Topics:**

- Object Oriented Programming fundamentals – data types, control structures
- Classes, methods, objects, Inheritance, polymorphism,
- Exception handling, Threads, Packages, Interfaces
- Files, I/O streams

| COURSE CONTENT | |
|---|---|
| **List of Experiments** | |
| **Week–1** | **CO1** |
| a) Write a JAVA program to display default value of all primitive data type of JAVA<br>b) Write a java program that displaytherootsofaquadraticequation$ax^2+bx=0$. Calculate the discriminate D and basing on value of D, describe the nature of root. | |
| **Week–2** | **CO1** |
| a) Write a JAVA program to search for an element in a given list of elements using binary search mechanism.<br>b) Write a JAVA program to sort for an element in a given list of elements using bubble sort | |
| **Week–3** | **CO2** |
| a) Write a JAVA program to implement class mechanism. Create a class, methods and invoke them inside main method.<br>b) Write a JAVA program implement method overloading. | |
| **Week–4** | **CO2** |
| a) Write a JAVA program to implement constructor.<br>b) Write a JAVA program to implement constructor overloading. | |
| **Week-5** | **CO2** |
| a) Write a JAVA program that implements Runtime polymorphism<br>b)Write a JAVA program implement method overriding. | |
| **Week-6** | **CO3** |
| a) Write a JAVA program for abstract class to find areas of different shapes<br>b)Write Java program(s) on ways of implementinginterface. | |
| **Week–7** | **CO3** |
| a) Write a JAVA program give example for "super" keyword.<br>b) Write a JAVA program to implement Interface. What kind of Inheritance can be achieved? | |
| **Week–8** | **CO3** |
| a)Write a JAVA program to Implement Single Inheritance.<br>b)Write a JAVA program to implement multi level Inheritance | |
| **Week–9** | **CO4** |
| b) Write a JAVA program that describes exception handling mechanism<br>c) Write a JAVA program Illustrating Multiple catch clauses | |
| **Week–10** | **CO4** |
| a) Write a JAVA program for creation of Java Built-in Exceptions.<br>b) Write a JAVA program for creation of User Defined Exception. | |
| **Week–11** | **CO5** |
| a) Write a JAVA program that creates threads by extending Thread class. First thread display "Good Morning "every 1 sec, the second thread displays "Hello "every 2 seconds and the third display "Welcome" every 3 seconds, (Repeat the same by implementing Runnable)<br>b) Write a program illustrating **isAlive()** and **join ()** | |
| **Week-12** | **CO5** |
| a) Write a JAVA program illustrative Daemon Threads | |

b) Write a JAVA program Producer-Consumer Problem

**Additional Experiments**

a) Design a simple calculator which performs all arithmetic operations. The interface should look like the calculator application of the operating system. Handle the exceptions ifany.
b) Write a Java Program that demonstrates Inner class.
c) Write a Java Program that demonstrate Command Line Arguments.

**Textbooks:**

1. JAVA one step ahead, AnithaSeth, B.L.Juneja, Oxford.
2. JoywithJAVA, Fundamentals of Object Oriented Programming, DebasisSamanta, MonalisaSarma, Cambridge, 2023.
3. JAVA9forProgrammers,PaulDeitel,HarveyDeitel,4th Edition,Pearson

**References Books:**

1. The completeReferenceJava,11th edition,HerbertSchildt,TMH
2. IntroductiontoJavaprogramming,7th Edition,YDanielLiang,Pearson

**Online Resources:**

1. https://nptel.ac.in/courses/106/105/106105191/
2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01288046454761881634 7

| NARAYANAENGINEERINGCOLLEGE:GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| III Sem | PYTHON PROGRAMMING | | | | | | | R23 |
| Course Code | Hours/Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A05304 | 0 | 1 | 2 | 48 | 2 | 30 | 70 | 100 |

**The main objectives of the course are to**

- Introduce core programming concepts of Python programming language.
- Demonstrate about Python data structures like Lists, Tuples, Sets and dictionaries
- Implement Functions, Modules and Regular Expressions in Python Programming and to create practical and contemporary applications using these

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Classify data structures of Python(BL4) |
| **CO 2** | Apply Python programming concepts to solve a variety of computational problems (BL3) |
| **CO 3** | Understand the principles of object-oriented programming (OOP) in Python, including classes, objects, inheritance, polymorphism, and encapsulation, and apply them to design and implement Python programs (BL3) |
| **CO 4** | Become proficient in using commonly used Python libraries and frameworks such as JSON, XML, NumPy, pandas (BL2) |
| **CO 5** | Exhibit competence in implementing and manipulating fundamental data structures such as lists, tuples, sets, dictionaries (BL3) |
| **CO6** | Propose new solutions to computational problems(BL6) |

| COURSECONTENT | | |
|---|---|---|
| **MODULE– 1** | | **9H** |

History of Python Programming Language, Thrust Areas of Python, Installing Anaconda Python Distribution, Installing and Using Jupyter Notebook.

**Parts of Python Programming Language:** Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, the type () Function and Is Operator, Dynamic and Strongly Typed Language.

**Control Flow Statements:** if statement, if-else statement, if...elif…else, Nested if statement, while Loop, for Loop, continue and break Statements, Catching Exceptions Using try and except Statement.

**Sample Experiments:**

1. Write a program to find the largest element among three Numbers.
2. Write a Program to display all prime numbers with in an interval
3. Write a program to swap two numbers without using a temporary variable.
4. Demonstrate the following Operators in Python with suitable examples.

i) Arithmetic Operators ii) Relational Operators iii) Assignment Operators iv) Logical Operators v)Bitwise Operators vi)Ternary Operator vii) Membership Operators
viii) Identity Operators

5. Write a program to add and multiply complex numbers
6. Write a program to print multiplication table of a given number.

| **MODULE– 2** | | **10H** |
|---|---|---|

**Functions:** Built-In Functions, Commonly Used Modules, Function Definition and Calling the function, return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments.

**Strings:** Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings.
**Lists**: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, del Statement.

**Sample Experiments:**

1. Write a program to define a function with multiple return values.
2. Write a program to define a function using default arguments.
3. Writeaprogramtofindthelengthofthestringwithoutusinganylibraryfunctions.
4. Write a program to check if the sub string is present in a given string or not.
5. Write a program to perform the given operations on a list:
   i. Addition ii. Insertion iii. slicing
6. Writeaprogramtoperformany5 built-in functions by taking any list.

| **MODULE– 3** | | **10H** |
|---|---|---|

**Dictionaries:** Creating Dictionary, Accessing and Modifying key: value Pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary Methods, del Statement.
**Tuples and Sets:** Creating Tuples, Basic Tuple Operations, tuple() Function, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Using zip() Function, Sets, Set Methods, Frozen set.

**Sample Experiments:**

1. Write a program to create tuples (name, age, address, college) for at least two members and concatenate the tuples and print the concatenated tuples.
2. Write a program to count the number of vowels in a string (No control flow allowed).
3. Write a program to check if a given key exists in a dictionary or not.
4. Write a program to add a new key-value pair to an existing dictionary.
5. Write a program to sum all the items in a given dictionary.

| **MODULE– 4** | | **10H** |
|---|---|---|

**Files**: Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, Pickle Module, Reading and Writing CSV Files, Python os and os.path Modules.
**Object-Oriented Programming:** Classes and Objects, Creating Classes in Python, Creating Objects in Python, Constructor Method, Classes with Multiple Objects, Class Attributes Vs Data Attributes, Encapsulation, Inheritance, Polymorphism.

1. Write a program to sort words in a file and put them in another file. The output file should have only lower-case words, so any upper-case words from source must be lowered.
2. Python program to print each line of a file in reverse order.
3. Python program to compute the number of characters, words and lines in a file.
4. Write a program to create, display, append, insert and reverse the order of the

items in the array.

5. Write a program to add, transpose and multiply two matrices.

6. Write a Python program to create a class that represents a shape. Include methods to calculate its area and perimeter. Implement subclasses for different shapes like circle, triangle, and square.

| **MODULE– 5** | | **9H** |
|---|---|---|

**Introduction to Data Science:** Functional Programming, JSON and XML in Python, NumPy with Python, Pandas.

**Sample Experiments:**

1. Python program to check whether a JSON string contains complex object or not.
2. Python Program to demonstrate NumPy arrays creation using array() function.
3. Python program to demonstrate use of ndim, shape, size, dtype.
4. Python program to demonstrate basic slicing, integer and Boolean indexing.
5. Python program to find min, max, sum, cumulative sum of array
6. Create a dictionary with at least five keys and each key represent value as a list where this list contains at least ten values and convert this dictionary as a pandas data frame and explore the data through the data frame as follows:

a) Apply head() function to the pandas data frame
b) Perform various data selection operations on Data Frame

**7.** Select any two columns from the above data frame, and observe the change in one attribute with respect to other attribute with scatter and plot operations in matplotlib

| **Total hours:** | **48hours** |
|---|---|

**Reference Books:**

1. Gowri shankarS, VeenaA., Introduction to Python Programming, CRC Press.
2. Python Programming, S Sridhar, J Indumathi, V M Hariharan, 2nd Edition, Pearson, 2024
3. Introduction to Programming Using Python, Y.DanielLiang, Pearson.

**Online Learning Resources /Virtual Labs:**

1. https://www.coursera.org/learn/python-for-applied-data-science-ai
2. https://www.coursera.org/learn/python?specialization=python#syllabus

| NARAYANA ENGINEERING COLLEGE:GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| IV Sem | MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS | | | | | | | R23 |
| Course Code | Hours / Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| **23A52402a** | 2 | 0 | 0 | 48 | 2 | 30 | 70 | 100 |

**Course Objectives:**

- To inculcate the basic knowledge of micro economics and financial accounting
- To make the students learn how demand is estimated for different products, input- output relationship for optimizing production and cost
- To Know the Various types of market structure and pricing methods and strategy
- To give an overview on investment appraisal method stop romote the students to learn how to plan long-term investment decisions.
- To provide fundamental skills on accounting and to explain the process of preparing financial statements.

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Define the concepts related to Managerial Economics, financial accounting and management**(BL2)** |
| **CO 2** | Understand the fundamentals of Economics viz., Demand, Production, cost, revenue and markets **(BL2)** |
| **CO 3** | Apply the Concept of Production cost and revenues for effective Business decision **(BL3)** |
| **CO 4** | Analyze how to invest their capital and maximize returns**(BL4)** |
| **CO 5** | Evaluate the capital budgeting techniques.**(BL5)** |
| **CO 6** | Developtheaccountingstatementsandevaluatethefinancialperformanceofbusiness entity **(BL5)** |

| COURSE CONTENT | | |
|---|---|---|
| **MODULE – 1** | **Managerial Economics** | **9H** |
| Introduction – Nature, meaning, significance, functions, and advantages. Demand-Concept, Function, Law of Demand- Demand Elasticity- Types– Measurement. Demand Forecasting-Factors governing Forecasting, Methods. Managerial Economics and Financial Accounting and Management. | | |
| **MODULE – 2** | **Production and Cost Analysis** | **10H** |
| Introduction – Nature, meaning, significance, functions and advantages. Production Function– Least- cost combination– Short run and long run Production Function- Isoquants and Is costs, Cost & Break-Even Analysis - Cost concepts and Cost behaviour- Break-Even Analysis (BEA) - Determination of Break-Even Point (Simple Problems). | | |
| **MODULE – 3** | **Business Organizations and Markets** | **10H** |
| Introduction – Forms of Business Organizations- Sole Proprietary - Partnership - Joint Stock Companies - Public Sector Enterprises. Types of Markets - Perfect and Imperfect Competition - Features of Perfect Competition Monopoly- Monopolistic Competition– Oligopoly-Price-Output Determination - Pricing Methods and Strategies. | | |
| **MODULE – 4** | **Capital Budgeting** | **10H** |
| Introduction – Nature, meaning, significance. Types of Working Capital, Components, Sources of Short-term and Long-term Capital, Estimating Working capital requirements. CapitalBudgeting–Features,Proposals,MethodsandEvaluation.Projects–PayBack Method, Accounting Rate of Return (ARR) Net Present Value (NPV) Internal Rate Return (IRR) Method (sample problems) | | |
| **MODULE – 5** | **Financial Accounting and Analysis** | **9H** |
| Introduction–Concepts and Conventions- Double-Entry Bookkeeping, Journal, Ledger, Trial Balance- Final Accounts (Trading Account, Profit and Loss Account and Balance Sheet with simple adjustments). Introduction to Financial Analysis - Analysis and Interpretation of Liquidity Ratios, Activity Ratios, and Capital structure Ratios and Profitability. | | |
| | **Total hours:** | **48 hours** |

**Text books:**


1. Varshney & Maheswari: Managerial Economics, Sultan Chand.

Aryasri: BusinessEconomicsandFinancialAnalysis,4/e,MGH.


**Reference Books:**


1. AhujaHl Managerial economics Schand.
2. S.A.SiddiquiandA.S.Siddiqui: Managerial Economics and Financial Analysis, New Age International.
3. Joseph G. Nellis and David Parker: Principles of Business Economics, Pearson, 2/e, New Delhi.
4. DomnickSalvatore: Managerial Economics in a Global Economy, Cengage.


**Online Learning Resources:**

https://www.slideshare.net/123ps/managerial-economics-ppt
https://www.slideshare.net/rossanz/production-and-cost-45827016
https://www.slideshare.net/darkyla/business-organizations-19917607
https://www.slideshare.net/balarajbl/market-and-classification-of-market
https://www.slideshare.net/ruchi101/capital-budgeting-ppt-59565396
https://www.slideshare.net/ashu1983/financial-accounting

| NARAYANA ENGINEERING COLLEGE:GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| IV Sem | PROBABILITY & STATISTICS | | | | | | | R23 |
| Course Code | Hours / Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A54401 | 3 | 0 | 0 | 48 | 3 | 30 | 70 | 100 |

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Acquire knowledge in finding the analysis of categorically and various statistical elementary tools.**(BL2,BL3)** |
| **CO 2** | Develop skills in designing mathematical models involving probability, random variables and the critical thinking in the theory of probability and its applications in real life problems. **(BL3,BL5)** |
| **CO 3** | Apply the theoretical probability distributions like binomial, Poisson, and Normal in the relevant application areas. **(BL3)** |
| **CO 4** | Analyze to test various hypotheses included in theory and types of errors for large samples. **(BL2,BL3)** |
| **CO 5** | Apply the different testing tools like t-test, F-test, chi-square test to analyze the relevant real life problems. **(BL3,BL5)** |

| COURSE CONTENT | | |
|---|---|---|
| **MODULE – 1** | **Descriptive statistics** | **9H** |
| Statistics Introduction, Population vs Sample, Collection of data, primary and secondary data, Measures of Central tendency, Measures of Variability (spread or variance) Skewness, Kurtosis, correlation, correlation coefficient, rank correlation, regression coefficients, method of least squares, regression lines. | | |
| **MODULE – 2** | **Probability** | **10H** |
| Probability, probability axioms, addition law and multiplicative law of probability, conditional probability, Baye's theorem, random variables (discrete and continuous), probability density functions, properties, mathematical expectation. | | |
| **MODULE – 3** | **Probability distributions** | **10H** |
| Probability distributions: Binomial, Poisson and Normal-their properties (Chebyshevs inequality). Approximation of the binomial distribution to normal distribution. | | |
| **MODULE – 4** | **Estimation and Testing of hypothesis, large sample tests** | **10H** |
| Estimation-parameters, statistics, sampling distribution, point estimation, Formulation of null hypothesis, alternative hypothesis, the critical and acceptance regions, level of significance, two types of errors and power of the test. Large Sample Tests: Test for single proportion, difference of proportions, test for single mean and difference of means. Confidence interval for parameters in one sample and two sample problems | | |
| **MODULE – 5** | **Small sample tests** | **9H** |
| Student t-distribution (test for single mean, two means and pair edt-test), testing of equality of variances (F-test), $\chi 2$ - test for goodness of fit, $\chi 2$ - test for independence of attributes | | |
| | **Total hours:** | **48 hours** |

**Textbooks:**

1. Millerand Freunds, Probability and Statistics for Engineers,7/e,Pearson,2008.
2. S.C. Gupta and V.K. Kapoor, Fundamentals of Mathematical Statistics, 11/e, Sultan Chand & Sons Publications, 2012.

**Reference Books:**

1. S.Ross, First Course in Probability, Pearson Education India,2002.
2. W.Feller, an Introduction to Probability Theory and its Applications,1/e,Wiley, 1968.
3. B.V.Ramana, Higher Engineering Mathematics, McGraw Hill Education.

Online Learning Resources:

1. https://onlinecourses.nptel.ac.in/noc21_ma74/preview
2. https://onlinecourses.nptel.ac.in/noc22_mg31/preview

| NARAYANA ENGINEERING COLLEGE:GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| IV Sem | OPERATING SYSTEMS | | | | | | | R23 |
| Course Code | Hours / Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A35401T | 3 | 0 | 0 | 48 | 3 | 30 | 70 | 100 |

**Course Objectives:** The main objectives of the course is to make student

- Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection
- Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
- Illustrate different conditions for deadlock and their possible solutions.

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Describe the basics of the operating systems, mechanisms of OS to handle processes, threads, and their communication. (L1) |
| **CO 2** | Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection. (L2) |
| **CO 3** | Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system. (L3) |
| **CO 4** | Illustrate different conditions for deadlock and their possible solutions.(L2) |
| **CO 5** | Analyze the memory management and its allocation policies.(L4) |

| COURSE CONTENT | | |
|---|---|---|
| **MODULE – 1** | | **9H** |
| **Operating Systems Overview:** Introduction, Operating system functions, Operating systems operations, Computing environments, Free and Open-Source Operating Systems<br>**System Structures:** Operating System Services, User and Operating-System Interface, system calls, Types of System Calls, system programs, Operating system Design and Implementation, Operating system structure, Building and Booting an Operating System, Operating system debugging | | |
| **MODULE – 2** | | **10H** |
| **Processes:** Process Concept, Process scheduling, Operations on processes, Inter-process communication.<br>**Threads and Concurrency:** Multithreading models, Thread libraries, Threading issues. **CPU Scheduling:** Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling. | | |
| **MODULE – 3** | | **10H** |
| **Synchronization Tools:** The Critical Section Problem, Peterson's Solution, Mutex Locks, Semaphores, Monitors, Classic problems of Synchronization.<br>**Deadlocks:** system Model, Deadlock characterization, Methods for handling Deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from Deadlock. | | |
| **MODULE – 4** | | **10H** |
| **Memory-Management Strategies:** Introduction, Contiguous memory allocation, Paging, Structure of the Page Table, Swapping.<br>**Virtual Memory Management:** Introduction, Demand paging, Copy-on-write, Page replacement, Allocation of frames, Thrashing.<br>**Storage Management:** Overview of Mass Storage Structure, HDD Scheduling. | | |
| **MODULE – 5** | | **9H** |
| **File System:** File System Interface: File concept, Access methods, Directory Structure; File system Implementation: File-system structure, File-system Operations, Directory implementation, Allocation method, Free space management; File-System Internals: File-System Mounting, Partitions and Mounting, File Sharing.<br>**Protection:** Goals of protection, Principles of protection, Protection Rings, Domain of protection, Access matrix. | | |
| | **Total hours:** | **48 hours** |

**Textbooks:**

1. Operating System Concepts, Silberschatz A, GalvinPB, GagneG,10th Edition, Wiley, 2018.
2. ModernOperatingSystems,TanenbaumAS,4th Edition,Pearson,2016

**Reference Books:**

1. Operating Systems -Internals and Design Principles, Stallings W, 9th edition, Pearson, 2018
2. Operating Systems :A Concept Based Approach, D.MDhamdhere, 3rd Edition, McGraw- Hill, 2013

**Online Learning Resources:**

1. https://nptel.ac.in/courses/106/106/106106144/
2. http://peterindia.net/OperatingSystems.html

| NARAYANA ENGINEERING COLLEGE:GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| IV Sem | DATABASE MANAGEMENT SYSTEMS | | | | | | | R23 |
| Course Code | Hours / Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A05402T | 3 | 0 | 0 | 48 | 3 | 30 | 70 | 100 |

**Course Objectives: T**he main objective of the course is to

- Introduce database management systems and to give a good formal foundation on the relational model of data and usage of Relational Algebra
- Introduce the concepts of basic SQL as a universal Database language
- Demonstrate the principles behind systematic database design approaches by covering conceptual design, logical design through normalization
- Provide an overview of physical design of a database system, by discussing Database indexing techniques and storage techniques

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Understand the basic concepts of database management systems**(BL2)** |
| **CO 2** | Analyze a given database application scenario to use ER model for conceptual design of the database **(BL4)** |
| **CO 3** | Utilize SQL proficiently to address diverse query challenges **(BL3).** |
| **CO 4** | Employ normalization methods to enhance database structure**(BL3)** |
| **CO 5** | Assess and implement transaction processing, concurrency control and database recovery protocols in databases. **(BL4)** |

| COURSE CONTENT | | |
|---|---|---|
| **MODULE – 1** | | **9H** |
| **Introduction:** Database system, Characteristics (Database Vs File System), Database Users, Advantages of Database systems, Database applications. Brief introduction of different Data Models; Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure, environment, Centralized and Client Server architecture for the database.<br>**Entity Relationship Model:** Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams. | | |
| **MODULE – 2** | | **10H** |
| **Relational Model:** Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance, Relational Algebra, Relational Calculus. BASIC SQL: Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update). | | |
| **MODULE – 3** | | **10H** |
| **SQL:** Basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions(Date and Time, Numeric, String conversion).Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, view(updatable and non-updatable), relational set operations. | | |
| **MODULE – 4** | | **10H** |
| **Schema Refinement (Normalization):**Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependency Loss less join and dependency preserving decomposition, (1NF, 2NF and 3 NF), concept of surrogate key, Boyce - Codd normal form(BCNF), MVD, Fourth normal form(4NF), Fifth Normal Form (5NF). | | |
| **MODULE – 5** | | **9H** |
| **Transaction Concept:** Transaction State, ACID properties, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability, lock based, time stamp based, optimistic, concurrency protocols, Deadlocks, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.<br>**Introduction to Indexing Techniques:** B+ Trees, operations on B+Trees, Hash Based Indexing: | | |
| | **Total hours:** | **48 hours** |

**Textbooks:**

1. Database Management Systems, 3rd edition, Raghurama Krishnan, Johannes Gehrke, TMH (For Chapters 2, 3, 4)
2. DatabaseSystemConcepts,5thedition,Silberschatz,Korth,Sudarsan,TMH( For Chapter 1 and Chapter 5)

**ReferenceBooks:**

1. IntroductiontoDatabaseSystems,8thedition,CJDate,Pearson.
2. DatabaseManagementSystem,6thedition,RamezElmasri,ShamkantB.Navathe, Pearson
3. DatabasePrinciplesFundamentalsofDesignImplementationandManagement, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning.

**Web-Resources:**

1. https://nptel.ac.in/courses/106/105/106105175/
2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01275806667282022456_shared/overview

| NARAYANA ENGINEERING COLLEGE:GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| IV Sem | SOFTWARE ENGINEERING | | | | | | | R23 |
| Course Code | Hours / Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A05403 | 3 | 0 | 0 | 48 | 3 | 30 | 70 | 100 |

**Course Objectives:** The objectives of this course are to introduce

- Software life cycle models, Software requirements and SRS document.
- Project Planning, quality control and ensuring good quality software.
- Software Testing strategies, use of CASE tools, Implementation issues, validation & verification procedures.

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | PerformvariouslifecycleactivitieslikeAnalysis,Design,Implementation,Testing and Maintenance **(BL3)** |
| **CO 2** | Analyse various software engineering models and apply methods for design and development of software projects **(BL4)** |
| **CO 3** | Develop system designs using appropriate techniques **(BL3)** |
| **CO 4** | Understand various testing techniques for a software project **(BL2)** |
| **CO 5** | Apply standards, CASE tools and techniques for engineering software projects **(BL3)** |

| COURSE CONTENT | | |
|---|---|---|
| **MODULE – 1** | | **9H** |
| **Introduction:** Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, No table changes in software development practices, Computer system engineering.<br>**Software Life Cycle Models:** Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model. | | |
| **MODULE – 2** | | **10H** |
| **Software Project Management:** Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, Halstead's software science, risk management.<br>**Requirements Analysis And Specification:** Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL. | | |
| **MODULE – 3** | | **10H** |
| **Software Design:** Overview of the design process, How to characterize a good software design? Layered arrangement of modules, Cohesion and Coupling, approaches to software design.<br>**Agility:** Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, Tool Set for the Agile Process (Text Book 2)<br>**Function-Oriented Software Design:** Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review.<br>**User Interface Design:** Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology. | | |
| **MODULE – 4** | | **10H** |
| **Coding And Testing:** Coding, Code review, Software documentation, Testing, Black-box testing, White-Box testing, Debugging, Program analysis tools, Integration testing**,** Testing object-oriented programs, Smoke testing, and Some general issues associated with testing.<br>**Software Reliability And Quality Management:** Software reliability. Statistical testing, Software quality, Software quality management system, ISO 9000. SEI Capability maturity model. Few other important quality standards, and Six Sigma**.** | | |
| **MODULE – 5** | | **9H** |
| **Computer-Aided Software Engineering (Case):** CASE and its scope, CASE environment, CASE support in the software lifecycle, other characteristics of CASE tools, Towards second generation CASE Tool, and Architecture of a CASE Environment.<br>**Software Maintenance:** Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.<br>**Software Reuse:** reuse- definition, introduction, reason behind no reuse so far, Basic issues in any reuse program, A reuse approach, and Reuse at organization level. | | |
| | **Total hours:** | **48 hours** |

**Text Books:**

1. Fundamentals of Software Engineering, RajibMall, 5th Edition, PHI.
2. Software Engineering Apractitioner'sApproach, RogerS.Pressman, 9th Edition, Mc- Graw Hill International Edition.

**Reference Books:**

1. Software Engineering, IanSommerville,10th Edition,Pearson.
2. Software Engineering, Principles and Practices, DeepakJain, Oxford University Press.

**e- Resources:**

1) https://nptel.ac.in/courses/106/105/106105182/
2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_shared/overview
3) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01338269041100390473 5_shared/overview

| NARAYANA ENGINEERING COLLEGE:GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| IV Sem | OPERATING SYSTEMS LAB | | | | | | | R23 |
| Course Code | Hours / Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A35401P | 0 | 0 | 3 | 48 | 1.5 | 30 | 70 | 100 |

**Course Objectives:** The main objectives of the course are to

- Provide insights into system calls, file systems, semaphores,
- Develop and debug CPU Scheduling algorithms, page replacement algorithms, thread implementation
- Implement Bankers Algorithms to Avoid the Dead Lock

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Trace different CPU Scheduling algorithms **(BL2)** |
| **CO 2** | Implement Bankers Algorithms to Avoid the Dead Lock **(BL3)** |
| **CO 3** | Evaluate Page replacement algorithms **(BL5)** |
| **CO 4** | Illustrate the file organization techniques **(BL4).** |
| **CO 5** | Illustrate Interprocess Communication and concurrent execution of threads **(BL4)** |

**Experiments covering the Topics:**

- UNIX fundamentals, commands & system calls
- CPU Scheduling algorithms, thread processing
- IPC, semaphores, monitors, deadlocks
- Page replacement algorithms, file allocation strategies
- Memory allocation strategies

| COURSE CONTENT | CO |
|---|---|
| **List of Experiments** | |
| **TASK–1** | **CO1** |
| 1. Practicing of Basic UNIX Commands. | |
| **TASK–2** | |
| 2. Write programs using the following UNIX operating system calls fork, exec, getpid, exit, wait, close, stat, opendir and readdir | |
| **TASK–3** | **CO1** |
| 3. Simulate UNIX commands like cp, ls, grep etc., | |
| **TASK–4** | **CO1** |
| 4. Simulate the following CPU scheduling algorithms<br>a)FCFS b) SJF c) Priority d) Round Robin | |
| **TASK–5** | **CO2** |
| 5. Control the number of ports opened by the operating system with<br>a) Semaphore b) Monitors. | |
| **TASK–6** | **CO2** |
| 6. Write a program to illustrate concurrent execution of threads using p threads library. | |
| **TASK–7** | **CO2** |
| 7. Write a program to solve producer – consumer problem using Semaphores. | |
| **TASK–8** | **CO3** |
| 8. Implement the following memory allocation methods for fixed partition<br>a) Firstfit b)Worstfit c) Bestfit | |
| **TASK–9** | **CO4** |
| 9. Simulate the following page replacement algorithms<br>**a)** FIFO b) LRU c) LFU | |
| **TASK–10** | **CO4** |
| 10. Simulate Paging Technique of memory management. | |
| **TASK–11** | **CO4** |
| 11. Implement Bankers Algorithm for Dead Lock avoidance and prevention | |
| **TASK–12** | **CO4** |
| 12. Simulate the following file allocation strategies<br>a) Sequential b) Indexed c) Linked | |

**Reference Books:**

1. Operating System Concepts, SilberschatzA, GalvinPB,GagneG,10th Edition,Wiley, 2018.
2. ModernOperatingSystems,TanenbaumAS,4th Edition,Pearson,2016
3. Operating Systems –Internals and Design Principles, StallingsW, 9th edition, Pearson, 2018
4. OperatingSystems:AConceptBasedApproach,D.MDhamdhere,3rd Edition, McGraw-Hill, 2013

**Online Learning Resources:**

1. https://www.cse.iitb.ac.in/~mythili/os/
2. http://peterindia.net/OperatingSystems.html

| IV Sem | **DATABASE MANAGEMENT SYSTEMS LAB** | | | | | | | R23 |
|---|---|---|---|---|---|---|---|---|
| Course Code | Hours / Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A05402P | 0 | 0 | 3 | 48 | 1.5 | 30 | 70 | 100 |

**Course Objectives:** This Course will enable students to

- Populate and query a database using SQL DDL/DML Commands
- Declare and enforce integrity constraints on a database
- Writing Queries using advanced concepts of SQL
- Programming PL/SQL including procedures, functions, cursors and triggers.

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Utilizing Data Definition Language (DDL), Data Manipulation Language (DML), and Data Control Language (DCL) commands effectively within a database environment **(BL3)** |
| **CO 2** | Constructing and execute queries to manipulate and retrieve data from databases **(BL3)** |
| **CO 3** | Develop application programs using PL/SQL **(BL3)** |
| **CO 4** | Analyze requirements and design custom Procedures, Functions, Cursors, and Triggers, leveraging their capabilities to automate tasks and optimize database functionality **(BL4)** |
| **CO 5** | Establish database connectivity through JDBC(Java Database Connectivity) **(BL3)** |

**Experiments covering the topics:**

- DDL, DML, DCL commands
- Queries, nested queries, built-in functions,
- PL/SQL programming-control structures
- Procedures, Functions, Cursors, Triggers,
- Database connectivity-ODBC/JDBC

| **COURSE CONTENT** | **CO** |
|---|---|
| **List of Experiments** | |
| **TASK–1** | **CO1** |
| 1. Creation, altering and droping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command. | |
| **TASK–2** | |
| 1. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints. Example :- Select the roll number and name of the student who secured fourth rank in the class. | |
| **TASK–3** | **CO1** |
| 1. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views. | |
| **TASK–4** | **CO1** |
| 1. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr andinstr),datefunctions(Sysdate,next_day,add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date) | |
| **TASK–5** | **CO2** |
| 1.<br>i. Create a simple PL/SQL program which includes declaration section, executable section and exception–Handling section(Ex. Student mark scan be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)<br>ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block. | |
| **TASK–6** | **CO2** |
| 1. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions. | |
| **TASK–7** | **CO2** |
| 1. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE-APPLICATION ERROR. | |
| **TASK–8** | **CO3** |

| | |
|---|---|
| 1. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES. | |
| **TASK–9** | **CO4** |
| 1. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions. | |
| **TASK–10** | **CO4** |
| 1. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables. | |
| **TASK–11** | **CO4** |
| 1. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers | |
| **TASK–12** | **CO4** |
| 1. Create a table and perform the search operation on table using indexing and non-indexing techniques. | |
| **TASK–13** | **CO4** |
| 1. Write a Java program that connects to a database using JDBC | |
| **TASK–14** | **CO4** |
| 1. Write a Java program to connect to a data base using JDBC and insert values into it | |
| **TASK–15** | **CO4** |
| 1. WriteaJavaprogramtoconnecttoadatabaseusingJDBCanddeletevaluesfromit | |
| | |

**Text Books / Suggested Reading:**


1. Oracle: The Complete Reference by Oracle Press
2. NileshShah, "DatabaseSystemsUsingOracle",PHI,2007
3. RickFVanderLans,"IntroductiontoSQL",FourthEdition,PearsonEducation,2007

| IV Sem | FULLSTACK DEVELOPMENT–1 | | | | | | | R23 |
|---|---|---|---|---|---|---|---|---|
| Course Code | Hours / Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A52401 | 0 | 1 | 2 | 48 | 1.5 | 30 | 70 | 100 |

**Course Objectives:** The main objectives of the course are to

- Make use of HTML elements and their attributes for designing static WebPages
- Build a webpage by applying appropriate CSS styles to HTML elements
- Experiment with Java Script to develop dynamic web pages and validate forms

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Design Websites**.(BL6)** |
| **CO 2** | Apply Styling to WebPages. **(BL4)** |
| **CO 3** | Make WebPages interactive**.(BL6)** |
| **CO 4** | Design Forms for applications**.(BL6)** |
| **CO 5** | Choose Control Structure based on the logic to be implemented**.(BL3)** |
| **CO6** | Understand HTML tags, Attributes and CSS properties**(BL2)** |

**Experiments covering the Topics:**

- Lists, Links and Images
- HTML Tables, Forms and Frames
- HTML5 and Cascading Style Sheets, Types of CSS
- Select or forms
- CSS with Color, Background, Font, Text and CSS Box Model
- Applying JavaScript-internal and external, I/O, Type Conversion
- JavaScript Conditional Statements and Loops, Pre-defined and User-defined Objects
- JavaScript Functions and Events
- Node.js

| COURSE CONTENT | CO |
|---|---|
| **List of Experiments** | |
| **TASK–1** | **CO1** |

**1. Lists, Links and Images**

a. Write a HTML program, to explain the working of lists.
Note: It should have an ordered list, un ordered list, nested lists and ordered list in an unordered list and definition lists.

b. Write a HTML program, to explain the working of hyperlinks using <a> tag and href, target attributes.

c. Create a HTML document that has your image and your friend's image with a specific height and width. Also when clicked on the images it should navigate to their respective profiles.

d. Write a HTML program, in such a way that, rather than placing large images on a page, the preferred technique is to use thumbnails by setting the height and width parameters to something like to 100*100 pixels. Each thumbnail image is also a link to a full sized version of the image. Create an image gallery using this technique

| TASK–2 | |
|---|---|

**2. HTML Tables, Forms and Frames**

a. Write a HTML program, to explain the working of tables.
(usetags:<table>,<tr>,<th>,
<td>and attributes: border, rowspan, colspan)

b. Write a HTML program, to explain the working of tables by preparing a timetable. (Note: Use <caption> tag to set the caption to the table & also use cell spacing, cell padding, border, rowspan, colspan etc.).

c. Write a HTML program, to explain the working of forms by designing Registration form. (Note: Include text field, password field, number field, date of birth field, checkboxes, radio buttons, list boxes using <select>&<option> tags, <text area> and two buttons ie: submit and reset. Use tables to provide a better view).

d. Write a HTML program, to explain the working of frames, such that page is to be divided into 3 parts on either direction. (Note: first frame image, second frame paragraph, third frame  hyperlink. And also make sure of using "no frame" attribute such that frames to be fixed).

| TASK–3 | CO1 |
|---|---|

**3. HTML5andCascadingStyleSheets,TypesofCSS**

a. WriteaHTMLprogram,thatmakesuseof<article>,<aside>,<figure>,<figcaption>, <footer>,<header>,<main>,<nav>,<section>, ,tags.
b. Write a HTML program, to embed audio and video into HTML webpage.
c. Write a program to apply different types (or levels of styles or style specification formats) -inline,internal,externalstylestoHTMLelements.(identifyselector,propertyand value).

| TASK–4 | CO1 |
|---|---|

**4. Selector forms**

a. Write a program to apply different types of selector forms
   - Simples elector (element, id, class, group, universal)
   - Combinator selector (descendant, child, adjacent sibling, general sibling)
   - Pseudo-class selector
   - Pseudo-element selector
   - Attribute selector

| TASK–5 | CO2 |
|---|---|

**5. CSS with Color, Background, Font, Text and CSS Box Model**

a. Write a program to demonstrate the various ways you can reference a color in CSS.
b. Write a CSS rule that places a background image halfway down the page, tilting It horizontally. The image should remain in place when the user scrolls up or down.
c. Write a program using the following terms related to CSS font and text:
   i. font-size ii. font-weight iii. font-style
   iv. text-decoration v. text-transformation vi. text-alignment
d. Writeaprogram,toexplaintheimportanceofCSSBoxmodelusing
   i. Content ii. Border iii. Margin iv. padding

| TASK–6 | CO2 |
|---|---|

**6. Applying JavaScript-internal and external, I/O, Type Conversion**

a. Write a program to embed internal and external JavaScript in a webpage.
b. Write a program to explain the different ways for displaying output.
c. Write a program to explain the different ways for taking input.
d. Create a webpage which uses prompt dialogue box to ask a voter for his name and age. Display the information in table format along with either the voter can vote or not

| TASK–7 | CO2 |
|---|---|

**7. Java Script Pre-defined and User-defined Objects**

a. Write a program using document object properties and methods.
b. Write a program using window object properties and methods.
c. Write a program using array object properties and methods.
d. Write a program using math object properties and methods.
e. Write a program using string object properties and methods.
f. Write a program using regex object properties and methods.
g. Write a program using date object properties and methods.
h. Write a program to explain user-defined object by using properties, methods, accessors, constructors and display.

| TASK–8 | CO3 |
|---|---|

**8. Java Script Conditional Statements and Loops**

a. Write a program which asks the user to enter three integers, obtains the numbers from the user and outputs HTML text that displays the larger number followed by the words

"LARGER NUMBER" in an information message dialog. If the numbers are equal, output HTML text as "EQUAL NUMBERS".

b. Write a program to display week days using switch case.
c. Write a program to print 1to10 numbers using for, while and do-while loops.
d. Writeaprogramtoprintdatainobjectusingfor-in,for-eachandfor-ofloops
e. Develop a program to determine whether a given number is an 'ARMSTRONG NUMBER' or not.[Eg:153 is an Armstrong number, since sum of the cube of the digits is equal to the number i.e.,13 + 53+ 33 = 153]
f. Write a program to display the denomination of the amount deposited in the bank in terms of 100's, 50's, 20's, 10's, 5's, 2's & 1's. (Eg: If deposited amount is Rs.163, the output should be 1-100's, 1-50's, 1- 10's, 1-2's & 1-1's)

| **TASK–9** | **CO4** |
|---|---|

**9. Java script Functions and Events**

a. Design a appropriate function should be called to display
   - Factorial of that number
   - Fibonacci series upto that number
   - Prime numbers upto that number
   - Is it palindrome or not

b. Design a HTML having a text box and four buttons named Factorial, Fibonacci, Prime, and Palindrome. When a button is pressed an appropriate function should be called to display
   1. Factorial of that number
   2. Fibonacci series upto that number
   3. Prime numbers upto that number
   4. Is it palindrome or not

c. Write a program to validate the following fields in a registration page
   i. Name(start with alphabet and followed by alphanumeric and the length should not be less than 6 characters)
   ii. Mobile(only numbers and length10 digits)
   iii. E-mail(should contain form at like xxxxxxx@xxxxxx.xxx)

**Textbooks:**


1. Programming the World Wide Web, 7thEdition, RobetWSebesta, Pearson,2013.
2. WebProgrammingwithHTML5,CSS and JavaScript, JohnDean, Jones&Bartlett Learning, 2019 (Chapters 1-11).
3. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, 2nd edition, APress, O'Reilly.


**Web Links:**

1. https://www.w3schools.com/html
2. https://www.w3schools.com/css
3. https://www.w3schools.com/js/
4. https://www.w3schools.com/nodejs

https://www.w3schools.com/typescript

| NARAYANA ENGINEERING COLLEGE :: GUDUR | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| IV Sem | DESIGN THINKING FOR INNOVATION | | | | | | | R23 |
| Course Code | Hours / Week | | | Total hrs | Credit | Max Marks | | |
| | L | T | P | | C | CIE | SEE | TOTAL |
| 23A99401 | 1 | 0 | 2 | 48 | 2 | 30 | 70 | 100 |

**Course Objectives:**

The objective of this course is to familiarize students with design thinking process as a tool for breakthrough innovation. It aims to equip students with design thinking skills and ignite the minds to create innovative ideas, develop solutions for real-time problems.

**Course Outcomes**: After successful completion of this course, the students should be able to:

| | |
|---|---|
| **CO 1** | Define the concepts related to design thinking.**(BL1, BL2)** |
| **CO 2** | Explain the fundamentals of Design Thinking and innovation **(BL1, BL2)** |
| **CO 3** | Apply the design thinking techniques for solving problems in various sectors. **(BL3)** |
| **CO 4** | Analyse to work in a multi disciplinary environment **(BL4)** |
| **CO 5** | Evaluate the value of creativity **(BL5)** |
| **CO 6** | Formulate specific problem statements of real time issues **(BL3, BL6)** |

| COURSE CONTENT | | |
|---|---|---|
| **MODULE – 1** | **Introduction to Design Thinking** | **9H** |
| Introduction to elements and principles of Design, basics of design-dot, line, shape, form as fundamental design components. Principles of design. Introduction to design thinking, history of Design Thinking, New materials in Industry. | | |
| **MODULE – 2** | **Design Thinking Process** | **10H** |
| Design thinking process (empathize, analyze, idea &prototype), implementing the process in driving inventions, design thinking in social innovations. Tools of design thinking - person, costumer, journey map, brainstorming, product development<br>**Activity:** Every student presents their idea in three minutes, Every student can present design process in the form of flow diagram or flow chart etc. Every student should explain about product development. | | |
| **MODULE – 3** | **Innovation** | **10H** |
| Art of innovation, Difference between innovation and creativity, role of creativity and innovation in organizations- Creativity to Innovation- Teams for innovation- Measuring the impact and value of creativity.<br>**Activity:** Debate on innovation and creativity, Flow and planning from idea to innovation, Debate on value-based innovation. | | |
| **MODULE – 4** | **Product Design** | **10H** |
| Problem formation, introduction to product design, Product strategies, Product value, Product planning, product specifications- Innovation towards product design- Case studies<br>**Activity:** Importance of modelling, how to set specifications, Explaining their own product design. | | |
| **MODULE – 5** | **Design Thinking in Business Processes** | **9H** |
| Design Thinking applied in Business &Strategic Innovation, Design Thinking principles that redefine business – Business challenges: Growth, Predictability, Change, Maintaining Relevance, Extreme competition, Standardization. Design thinking to meet corporate needs- DesignthinkingforStartups-DefiningandtestingBusinessModelsandBusinessCases- Developing & testing prototypes.<br>**Activity:** How to market our own product, About maintenance, Reliability and plan for startup. | | |
| | **Total hours:** | **48 hours** |

**Textbooks:**


1. TimBrown, Change by design, Harper Bollins(2009)
2. IdrisMootee, Design Thinking for Strategic Innovation, 2013, John Wiley&Sons.

**Reference Books:**


1. DavidLee, Design Thinking in the Classroom, Ulyssespress
2. ShrutinNShetty, Design the Future, NortonPress
3. WilliamLidwell, Universal Principles of Design-Kritinaholden,JillButter.

Chesbrough.H,TheEraofOpenInnovation– 2013


**Online Learning Resources:** https://nptel.ac.in/courses/110/106/110106124/
https://nptel.ac.in/courses/109/104/109104109/
https://swayam.gov.in/nd1_noc19_mg60/preview