SCHEME OF EXAMINATION

and

SYLLABUS

for

Diploma (Web Designing)

Offered by

Community College of Skill Development

J.C. Bose University of Science & Technology YMCA

Sector-6, Mathura Road, Faridabad,

Haryana, India

2024-25

ABOUT THE PROGRAM

A Diploma in Web Designing is a specialized program aimed at equipping individuals with the necessary skills and knowledge to design and develop websites effectively. This comprehensive curriculum typically covers a range of topics including graphic design principles, HTML, CSS, JavaScript, responsive web design, web accessibility. Throughout the program, students engage in hands-on learning experiences, working on practical projects to apply their newfound skills and knowledge. This approach enables them to build a strong portfolio that showcases their abilities to potential employers or clients. Upon completion of the diploma, graduates are prepared to pursue various career opportunities in the field of web design and development. They may work as web designers, front-end developers, UX/UI designers, or freelance web developers. With the increasing importance of digital presence for businesses and organizations, a diploma in web designing provides graduates with valuable expertise to thrive in this dynamic and rapidly evolving industry.

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

PEO-1: To create theoretical and practical knowledge about core areas related to the field of web development and provides a judicious mix of skills relating to a profession and appropriate content of general education.

PEO-2: Ensures that the students have adequate knowledge and skills, so that they are work ready at each exit point of the programme.

PEO-3: To improve team building, team working and leadership skills of the students with high regard for ethical values and social responsibilities.

PEO- 4: To enable students to communicate effectively and efficiently.

PROGRAMME OUTCOMES (POs)

PO-1: Improve their computer literacy, their basic understanding of operative systems and working. Knowledge of software commonly used in academic and professional environments.

PO-2: Do Academic and Professional Presentations - Designing and delivering an effective presentation and developing the various IT skills to the electronic databases.

PO-3: Gain Analytical skills in the field of Computer Applications.

PO-4: Acquire the knowledge of the latest technologies in web development and future trends.

PO-5: Synthesize principles and theories of designing websites to different computing paradigms.

PO-6: Develop professional skills that prepare the students for employment and for life-long learning in specific areas of web development and related fields.

PO-7: Bridge the potential skill gap identified between the Institution and Industry.

PROGRAMME SPECIFIC OUTCOMES (PSOs)

1) Understand, analyze and develop computer programs in the areas related to web design.

2) The student will be ready and skilled to take-up a career or to pursue higher studies with high regard to ethical values and social responsibilities.

SCHEME OF EXAMINATION

FIRST SEMESTER

Subject Code	Subject Name	L-T-P	Credits	Marks Weightage		Course Type
				Internal	External	
BSC-105EL	English Literacy	3-0-0	3	25	75	BSC
BSC-106	Typography and Computer Application	3-0-0	3	25	75	BSC
WD-101	Basics of Operating System	3-0-0	3	25	75	PCC
WD-102	Fundamental of Computers and C Programming	3-0-0	3	25	75	PCC
MAC-101 to 103	Mandatory Audit Course	3-0-0	3	25	75	MAC
WD-L-101	C Programming Lab	0-0-3	3	30	70	SDP
WD-L-102	PC Software and HTML Lab	0-0-2	2	30	70	SDP
Total		15-0-5	20	185	515	

SECOND SEMESTER

Subject Code	Subject Name	L-T-P	Credits	Marks Weightage		Course Type
				Internal	External	
BSC-204BS	Behavioral Skills	3-0-0	3	25	75	BSC
WD-201	Web Designing	3-0-0	3	25	75	PCC
WD-202	Object Oriented Programming	3-0-0	3	25	75	PCC
WD-203	Data Structures and Algorithms	3-0-0	3	25	75	PCC
WD-206	Discrete Mathematics	3-0-0	3	25	75	BSC
WD-L-201	C++ Programming Lab	0-0-3	3	30	70	SDP
WD-L-202	Web Designing Lab	0-0-2	2	30	70	SDP
Total		15-0-5	20	185	515	

THIRD SEMESTER

Subject Code	Subject Name	L-T-P	Credits	Marks Weightage		Course Type
				Internal	External	
WD-301	Java Programming	3-0-0	3	25	75	PCC
WD-302	Computer Networks	3-0-0	3	25	75	PCC
WD-303	Database Management System	3-0-0	3	25	75	PCC
BSC-302ES	Employability Skills	3-0-0	3	25	75	BSC
WD-L-301	SQL & PHP Lab	0-0-3	3	30	70	SDP
WD-L-302	Java Programming Lab	0-0-3	3	30	70	SDP
WD-L-303	WordPress Lab	0-0-2	2	30	70	SDP
Total		12-0-8	20	190	510	

FOURTH SEMESTER

Subject Code	Subject Name	L-T-P	Credits	Marks Weightage		Course Type
				Internal	External	
WD-401	Introduction to Python	3-0-0	3	25	75	PCC
WD-402	Web Technologies using ASP(dot)NET	3-0-0	3	25	75	PCC
WD-L-401	Python Lab	0-0-3	2.5	30	70	SDP
BSC-402P	Project	0-0-2	3	30	70	SDP
OEC-401 to 403	Open Elective Course	3-0-0	3	25	75	OEC
WD-L-402	ASP(dot)NET Lab	0-0-3	2.5	30	70	SDP
PEC-WD-401 to 403	Program Elective Course	3-0-0	3	25	75	PEC
Total		12-0-8	20	190	510	

LIST OF MANDATORY AUDIT COURSE

COURSE CODE	COURSE NAME
MAC-101	Human Value And Professional Ethics
MAC-102	Balanced Diet And Nutrition
MAC-103	Environmental Science

LIST OF OPEN ELECTIVE COURSE

COURSE CODE	COURSE NAME
OEC-401	Entrepreneurship
OEC-402	Trends in Technology
OEC-403	Waste Management

LIST OF PROGRAM ELECTIVE COURSE

COURSE CODE	COURSE NAME
PEC-WD-401	Technology Trends in IT
PEC-WD-402	Introduction to AI
PEC-WD-403	Computer Network Security

FIRST SEMESTER

Subject Code	Subject Name	L-T-P	Credits	Marks Weightage		Course Type
				Internal	**External**	
BSC-105EL	English Literacy	3-0-0	3	25	75	BSC
BSC-106	Typography and Computer Application	3-0-0	3	25	75	BSC
WD-101	Basics of Operating System	3-0-0	3	25	75	PCC
WD-102	Fundamental of Computers and C Programming	3-0-0	3	25	75	PCC
MAC-101 to 103	Mandatory Audit Course	3-0-0	3	25	75	MAC
WD-L-101	C Programming Lab	0-0-3	3	30	70	SDP
WD-L-102	PC Software and HTML Lab	0-0-2	2	30	70	SDP
Total		15-0-5	20	185	515	

ENGLISH LITERACY
BSC-105EL

Total Credits: 3

L   T   P

Theory: 75

3   0   0

Sessional: 25

Course Objectives:
The objective of studying this course is to acquire knowledge on the Basic English grammar starting from speeches to synaptic category going forward with tenses and its types. To comprehend voices, narration and sentence making.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Demonstrate a comprehensive understanding of the fundamental parts of speech and their functions in communication.

CO2: Analyse the usage and placement of prepositions, conjunctions, and interjections to enhance the coherence and flow of written and spoken language.

CO3: Interpret the nuances of different tenses, including present, past, and future, to accurately convey temporal relationships and meanings in writing and speech.

CO4: Evaluate the effectiveness of various sentence formation techniques, such as active and passive voice, direct and indirect narration, and different sentence structures, in achieving specific communicative goals.

Unit-I
Parts of Speech: Noun, Pronoun, Verb, Adverb, Adjective.

Unit-II
Literacy Skills: Preposition, Conjunction, Interjection.

Unit-III
Fragment of Tenses: Present tense, Past Tense, Future Tense.

Unit-IV
Sentence Formation: Active and Passive voice, Direct and Indirect Narration, Simple Sentences, Compound Sentences, Complex Sentences, Compound-Complex Sentences.

Practical Exercises:
The learners are required to

1. Participate in a discussion where you identify and categorize different parts of speech used in a given paragraph.
2. Select sentences from a text and prepare exercises where you replace nouns with pronouns to enhance clarity and cohesion.
3. Conduct a group activity where each participant creates sentences using various prepositions and conjunctions to demonstrate their understanding of literacy skills.
4. Estimate the correct tense for a series of sentences and analyze how the choice of tense impacts the meaning and tone of each sentence.
5. Compare sentences written in active and passive voice, then conduct a writing exercise where you rewrite sentences from one voice to the other to understand the differences in tone and emphasis.

Suggested Readings:
1. Wren and Martin. High School English Grammar and Composition. New Delhi: RRP, 2007.
2. Murphy, Raymond. Essential English Grammar. New Delhi: Cambridge, 2017.

Note:

1. Only the latest editions of the above books are recommended

TYPOGRAPHY AND COMPUTER APPLICATION
BSC-106

Total Credits: 3

L   T   P

Theory: 75

3   0   0

Sessional: 25

Course Objectives:

The course aims to familiarize students with Microsoft Office applications for efficient productivity. It covers MS Windows basics and control panel operations, emphasizes document creation and advanced features in MS Word, delves into data manipulation and analysis in MS Excel, and introduces effective presentations and database management in MS PowerPoint and MS Access.

Course Outcomes:

After the successful completion of the course, students will be able to:

- CO1: Demonstrate proficiency in navigating and utilizing basic components such as icons, taskbar, and desktop in MS Windows.
- CO2: Evaluate the effectiveness of MS Word features for document management, including file handling and printing capabilities.
- CO3: Interpret advanced features such as pivot tables, conditional formatting, and what-if analysis to facilitate data management and decision-making.
- CO4: Prepare engaging presentations using MS PowerPoint, incorporating various slide manipulation techniques and multimedia elements.

Unit-I

MS Windows: Basic components of windows, icons, types of icons, taskbar, activating windows, using desktop, title bar, running applications, exploring computer, managing files and folders, copying and moving files and folders. Control panel – display properties, adding and removing software and hardware, setting date and time, screensaver and appearance using windows accessories.

Unit-II

Documentation Using MS Word: Introduction to word processing interface, Toolbars, Menus, Creating & Editing Document, Formatting Document, Finding and replacing text, Format painter, Header and footer, Drop cap, Auto-text, Autocorrect, Spelling and Grammar Tool, Document Dictionary, Page Formatting, Bookmark, Previewing and printing document, Advance Features of MS-Word-Mail Merge, Macros, Tables, File Management, Printing, Styles, linking and embedding object, Template.

Unit-III

Electronic Spreadsheet using MS Excel: Introduction to MS-Excel, Cell, cell address, Creating & Editing Worksheet, Formatting and Essential Operations, Moving and copying data in excel, Header and footer, Formulas and Functions, Charts, Cell referencing, Page setup, Macros, Advance features of MS-Excel-Pivot table & Pivot Chart, Linking and Consolidation, Database Management using Excel-Sorting, Filtering, Validation, What if analysis with Goal Seek, Conditional formatting, Collaborating with Other Users, Analysing and Presenting Complex data.

Unit-IV

Presentation using MS PowerPoint: Presentations, Creating, Manipulating & Enhancing Slides, Organizational Charts, Excel Charts, Word Art, layering art Objects, Animations and Sounds, Inserting Animated Pictures or Accessing through Object, Inserting Recorded Sound Effect or In-Built Sound Effect., Introduction to MS Access: creating database creating and

manipulating tables, forms, queries, reports, modules, importing and exporting of data.

Practical Exercises:

The learners are required to

1. Participate in an exercise to select and customize desktop icons, exploring different types and arranging them according to preference.
2. Prepare a document formatting exercise by selecting and applying different styles, headers, and footers to create a polished document.
3. Conduct a data analysis exercise by estimating and comparing the results of applying different formulas and functions to a given dataset.
4. Estimate the time required to prepare and conduct a presentation by selecting and arranging slides, including animations and transitions.
5. Compare the presentation of data using different chart types in Excel by preparing and conducting a chart creation exercise with the same dataset.

Suggested Readings:

1. V. Rajaraman, Computer Fundamentals.
2. Ashok Arora, Fundamentals of Computer Systems.
3. Russell A Stultz, Fundamentals of Computer Systems.

Note:

1. Only the latest editions of the above books are recommended

BASICS OF OPERATING SYSTEM
WD-101

Total Credits: 3

L T P   Theory: 75

3 0 0   Sessional: 25

Course Objectives:

The course aims to provide a deep understanding of operating systems, covering concepts, structures, and functions, including necessity, types, and services. It aims to establish a foundation for comprehensive analysis of processes, memory, file, and disk management, alongside exploration of various operating system architectures and case studies.

Course Outcomes:

After the successful completion of the course, students will be able to:

CO1: Demonstrate a thorough understanding of operating system concepts, including the need, types, and services, by explaining their significance and functionality in computing environments.

CO2: Analyse different process states, scheduling techniques, and criteria, evaluating scheduling algorithms' efficiency based on CPU utilization and response time metrics.

CO3: Interpret memory and file management principles, applying techniques such as swapping, paging, and segmentation to optimize resource allocation and access efficiency.

CO4: Evaluate disk management strategies and various operating system architectures through case studies, comparing their features and performance to make informed decisions in system design and implementation.

Unit-I

Operating Systems: Concept of Operating Systems, Need of operating system, Types of Operating Systems, Services of operating system, Structure of an operating system, Functions of operating system.

Unit-II

Processes: Definition, Different states of a Process, Process Scheduling, Types of Schedulers, and Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time. Scheduling algorithms, FCFS, SJF, Priority, RR, Multiprocessor scheduling.

Unit-III

Memory & File Management: Logical and Physical address space, Swapping, Contiguous Memory allocation, Virtual Memory, Paging, Segmentation, Demand Paging, Page Replacement Algorithm, Thrashing. Concept of File, Access methods, Directory structure, File System structure, Allocation methods.

Unit-IV

Disk Management & Case Study: Disk Structure, Disk Scheduling Algorithm, Network Operating System, Real Time Operating System, Distributed Operating System, Study of various Operating Systems: Windows, Dos, Linux etc.

Practical Exercises:

The learners are required to

1. Conduct a hands-on activity to explore the structure and functions of an operating system, preparing diagrams to illustrate its components and their interactions.
2. Analyse the impact of process scheduling criteria on system performance by

comparing different scheduling algorithms' results and evaluating their effectiveness.

3. Conduct a data analysis exercise by estimating and comparing the results of applying different formulas and functions to a given dataset.

4. Participate in a case study analysis of various operating systems, selecting examples like Windows, DOS, and Linux, to compare their disk management strategies and disk scheduling algorithms.

5. Prepare and conduct an exercise to analyze the efficiency of page replacement algorithms like LRU and FIFO, estimating their performance in preventing thrashing and optimizing memory usage..

Suggested Readings:

1. Operating System Concepts, Silberschatz, Ninth Edition, Willey Publication.
2. Operating Systems, Internals and Design Principles, Stallings, Seventh Edition, Pearson Publication.
3. Modern Operating Systems, Tanenbaum, Fourth Edition. Pearson Publication.

Note:

1. Only the latest editions of the above books are recommended

FUNDAMENTALS OF COMPUTERS AND C PROGRAMMING
WD-102

Total Credits: 3

L T P
3 0 0

Theory: 75

Sessional: 25

Course Objectives:
The course aims to provide foundational knowledge of computer systems, problem-solving techniques, C programming fundamentals, arrays, functions, pointers, structures, unions, and file handling. Emphasize practical skills, programming logic, and understanding of computer impacts on society, including addressing ethical concerns like computer crimes and viruses.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Understand the fundamental concepts of computer systems, including hardware components, types of software, programming languages, and the social implications of computer technology.

CO2: Develop problem-solving skills using pseudocode, flowcharts, and decision tables; and gain proficiency in C programming fundamentals including control statements and basic operators.

CO3: Implement and manipulate arrays, functions, pointers, and dynamic memory allocation in C, and differentiate between macros and functions.

CO4: Define and utilize structures and unions, manage file operations in C, and effectively debug programs while understanding different types of errors.

Unit-I
Computer Fundamentals: Concept of data and information; Components of Computer: Hardware Input Device, Output Device. CPU: Components of CPU; Memory and Storage Devices; Computer Software: System Software and Application Software; Functions of Operating System. Programming Languages: Machine, Assembly, High Level Language, 4GL; Language Translator; Linker, Loader; Classification of Computers: Micro, Mini, Mainframe, Supercomputer. Advantages of Computer, Limitations of Computer, Range of Applications of Computer, Social Concerns of Computer Technology: Positive and Negative Impacts, Computer Crimes, Viruses and their remedial solutions.

Unit-II
Problem Solving: pseudocode, Problem Identification, Analysis, Flowcharts, Decision Tables, Pseudo codes and algorithms, Program Coding, Program Testing and Execution. C Programming Fundamentals: Keywords, Variables and Constants, Structure of a C program. Operators & Expressions: Arithmetic, Unary, Logical, Bit-wise, Assignment & Conditional Operators, Library Functions, Control Statements: Looping using while, do…while, for statements, Nested loops; decision making using if…else, Else If Ladder; Switch, break, Continue and Go to statements.

Unit-III
Arrays & Functions: Declaration and Initialization; Multidimensional Arrays. String: Operations of Strings; Functions: Defining & Accessing User defined functions, Function Prototype, Passing Arguments, Passing array as argument, Recursion, Use of Library Functions; Macro vs. Functions. Pointers: Declarations, Operations on Pointers, Passing to a function, Pointers & Arrays, Array of Pointers, Array accessing through pointers, Pointer to functions, Function returning pointers, Dynamic Memory Allocations.

Unit-IV

Structures and Union: Defining and Initializing Structure, Array within Structure, Array of Structure, Nesting of Structure, Pointer to Structure, Passing structure and its pointer to Functions; Unions: Introduction to Unions and its Utilities. Files Handling: Opening and closing file in C; Create, Read and Write data to a file; Modes of Files, Operations on file using C Library Functions; Working with Command Line Arguments. Program Debugging and types of errors.

Practical Exercises:

The learners are required to

1. Identify and label the different parts of a disassembled computer (CPU, RAM, hard drive, input/output devices). Describe the function of each component and explain their role in the overall operation of the computer.
2. Create pseudocode and a corresponding flowchart for a program that sorts an array of integers using the bubble sort algorithm. Analyze the efficiency of the algorithm and explain each step of the process.
3. Write a C program that takes two integers from user input and performs basic arithmetic operations (addition, subtraction, multiplication, division). Apply control statements (if-else, switch-case) to handle the operations and implement error checking for division by zero.
4. Write a C program that manages a list of student grades using arrays. Implement functions to add grades, calculate the average, find the highest and lowest grades, and display the grades. Evaluate the effectiveness and efficiency of your functions.
5. Develop a C program that creates a file, writes user input data to the file, and then reads and displays the data from the file. Use command line arguments to specify the file name. Debug the program to handle file-related errors such as file not found or permission denied.

Suggested Readings:

1. Problem Solving and Program Design in C, 4th edition, by jeri R. Hanly and Elli B.Koffman.
2. Programming in C by Pradip Dey, Manas Ghosh 2nd edition Oxford University Press.
3. E.Balaguruswamy, Programming in ANSI C 5th Edition McGraw-Hill
4. A first book of ANSI C by Gray J.Brosin 3rd edition Cengage Delmar Learning India P.Ltd
5. AL Kelly, Iraphol,Programming in C,4th edition Addison-Wesley – Professional

Note:

1. Only the latest editions of the above books are recommended

HUMAN VALUE AND PROFESSIONAL ETHICS
MAC-101

Total Credits: 3

L T P
3 0 0

Theory: 75

Sessional: 25

Course Objectives:
The course aims to instil the significance of value education, emphasizing holistic living, ethical values, harmony in relationships, and understanding human rights. It addresses personal and societal responsibilities, fosters environmental balance, and tackles social evils through critical thinking and problem-solving skills for a well-rounded, responsible citizenry.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Understand the significance of value education, holistic living, and the balance of body, mind, and intellect in contemporary life.

CO2: Analyze the essential values for life such as truth, integrity, empathy, and teamwork, and apply them to personal growth.

CO3: Evaluate harmony in family and society, understanding human relationships, and the role of trust and respect in fostering unity.

CO4: Apply knowledge of human rights and social evils to promote peace, non-violence, and social welfare, addressing contemporary societal challenges.

Unit-I
Introduction: Value education-its purpose and significance in the present world, Value system, The role of culture and civilization, Holistic living, Balancing the outer and inner - Body, Mind and Intellectual level- Duties and responsibilities.

Unit-II
Salient values for life: Truth, commitment, honesty and integrity, forgiveness and love, empathy and ability to sacrifice, care, unity, and inclusiveness, Self-esteem and self-confidence, punctuality - Time, task and resource management, Problem solving and decision-making skills- Interpersonal and Intra personal relationship, Team work, Positive and creative thinking.

Unit-III
Understanding Harmony: Harmony in Family and Society-How to owe responsibilities in family, Understanding Values in Human- Human relations, Role of Trust and Respect, Samman (Respect) for all, Akhand Samaj (A United Society)
Harmony in Nature: Understanding the Harmony in Nature, making sure your contribution is in harmony with nature, Interconnectedness and mutual fulfilment, Environment and Ecological balance.

Unit-IV
Human Right and Social Evils: Human Rights: Universal Declaration of Human Rights National Integration - Peace and non-violence - Dr. APJ Kalam's ten points for enlightened citizenship - Social Values and Welfare of the citizen - The role of media in value building - Human Rights violations - Social Evils: Corruption, Cybercrime, Terrorism, Alcoholism, Drug addiction, Dowry, Domestic violence, Untouchability, female infanticide, atrocities against women and how to tackle them.

Practical Exercises:
The learners are required to

1. Create a personal value system chart, detailing your key values and their significance in daily life and decision-making.
2. Analyze a case study on integrity, identifying challenges faced and strategies used to maintain honesty and commitment.
3. Evaluate the harmony within your family relationships, identifying areas of improvement for fostering trust, respect, and unity.
4. Design a community project that promotes environmental harmony, ensuring sustainable practices and ecological balance.
5. Develop a plan to raise awareness on human rights and tackle social evils, incorporating strategies from Dr. APJ Kalam's ten points for citizenship.

Suggested Readings:
1. R. R. Gaur, R Sangal, G P Bagaria, 2009, A Foundation Course in Human Values and Professional Ethics.
2. Prof. K. V. Subba Raju, 2013, Success Secrets for Engineering Students, Smart Student Publications, 3rd Edition.
3. Ivan IIIich, 1974, Energy & Equity, The Trinity Press, Worcester, and HarperCollins, USA.

Note:
1. Only the latest editions of the above books are recommended

BALANCED DIET AND NUTRITION
MAC-102

Total Credits: 3

L   T   P   Theory: 75
3   0   0   Sessional: 25

Course Objectives:
The course aims to provide a comprehensive understanding of food nutrition, covering the meaning of nutrition, nutritional requirements, and food components. It explores macronutrients (carbohydrate, fat, protein) and micronutrients (vitamins, minerals), emphasizing their sources, functions, and effects on the body. Additionally, it addresses balanced diet planning and yogic concepts of diet and nutrition.

Course Outcomes:
After the successful completion of the course, students will be able to:
- CO1: Analyze nutritional requirements and food components for planning balanced diets, applying knowledge of macronutrients and micronutrients.
- CO2: Evaluate the selection, preparation, and nutritive value of food groups to make informed dietary choices, demonstrating critical thinking skills.
- CO3: Synthesize key concepts of energy metabolism and factors affecting energy expenditure, demonstrating understanding and application of physiological principles.
- CO4: Apply yogic principles to dietary practices, integrating traditional texts and holistic approaches to promote overall health and well-being.

Unit-I
Concepts and Components of Food Nutrition: Meaning of nutrition, Basic definition regarding nutritional requirements, Nutritional need of human; Concept of food, Acceptance of food, Function of food, Components of food and their classification; Macronutrients – Carbohydrate, Fat, Protein (source, function and effect on the body); Micronutrients – Vitamins, Mineral, Water, roughage (source, function and effect on body); Planning Balanced Diet.

Unit-II
Food Group: Cereals and Millet – Selection, Preparation and Nutritive value; Pulses, Nuts and Oilseeds- Selection, Preparation and Nutritive value; Milk and Milk production - Selection, Preparation and Nutritive value; Vegetable and Fruits - Selection, Preparation and Nutritive value; Fatty oil and Sugar, Jaggery - Selection, Preparation and Nutritive value.

Unit-III
Food and Digestion: Energy – Key concepts, Definition and Components of energy requirements.; Energy – Imbalance concepts of metabolism, anabolism and catabolism; Calorie requirement – BMR, SDA; Physical activity – carbohydrates, lipids and protein metabolism; Factors affecting energy- requirement and expenses; Factors affecting BMR; Factors influencing energy expenditure in physical activity; Methods and requirements for estimating energy expenditure.

Unit-IV
Yogic concepts of Diet and Nutrition: General introduction to diet concepts, concepts of mitahara, Definition and classification, yogic diet according to traditional yoga texts; Concepts of diet according to GherandSamhita and Hathpradeepika; Satvik, Rajsik and Tamasik diet as describe in Bhagwadgeeta; Pathya and Apathya food according to the texts of Yoga; Role of yogic diet in healthy living; Diet according to nature of the body – Vata,

Pitta and Kapha

Practical Exercises:

The learners are required to

1. Analyze food labels to identify macronutrient and micronutrient content, applying knowledge of nutritional requirements and classification of food components.
2. Plan a balanced diet menu incorporating various food groups, considering selection, preparation methods, and nutritive values.
3. Calculate Basal Metabolic Rate (BMR) and estimate energy expenditure for different physical activities, applying metabolic concepts.
4. Evaluate the nutritive value of different food items using food composition tables, considering their impact on energy balance and metabolism.
5. Compare and contrast traditional yogic dietary principles with modern nutritional recommendations, discussing their implications for health and well-being.

Suggested Readings:

1. Bakhru, H. K., 1991, A Complete Handbook of Nature Cure.
2. Kumar Neeraj, Nagendra, 2014, MeraAaharMeraSwasthya.

Note:

1. Only the latest editions of the above books are recommended

ENVIRONMENTAL SCIENCE
MAC-103

Total Credits: 3

L T P

3 0 0

Theory: 75

Sessional: 25

Course Objectives:
The course aims to explore environmental studies, emphasizing its multidisciplinary nature, the significance of resource management, understanding ecosystem dynamics, and conserving biodiversity. It highlights the importance of public awareness, sustainable practices, and conservation strategies to address environmental challenges globally and locally.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Understand the multidisciplinary nature of environmental studies and its importance by defining key concepts and discussing their relevance.

CO2: Analyze the challenges associated with renewable and non-renewable resources, proposing sustainable conservation strategies for forest, water, mineral, food, energy, and land resources.

CO3: Explore ecosystem dynamics by describing the structure, function, and types of ecosystems, including energy flow and ecological succession.

CO4: Evaluate biodiversity conservation methods by identifying genetic, species, and ecosystem diversity, discussing threats, and recommending in-situ and ex-situ conservation measures.

Unit-I
Understanding Environmental Studies: Exploring the multidisciplinary nature of environmental studies, defining its scope, and highlighting its importance. Emphasizing the need for public awareness in addressing environmental issues.

Unit-II
Natural Resource Management: Analyzing renewable and non-renewable resources and associated challenges. Studying forest, water, mineral, food, energy, and land resources, along with their exploitation and conservation strategies..

Unit-III
Ecosystem Dynamics: Understanding the structure and function of ecosystems, including producers, consumers, and decomposers. Exploring energy flow, ecological succession, and various ecosystem types such as forests, grasslands, deserts, and aquatic ecosystems.

Unit-IV
Biodiversity Conservation: Defining genetic, species, and ecosystem diversity. Examining the value of biodiversity at global, national, and local levels. Identifying threats to biodiversity and discussing conservation measures, both in-situ and ex-situ.

Practical Exercises:
The learners are required to

1. Assess local environmental issues by identifying and discussing their multidisciplinary nature, highlighting the need for public awareness.
2. Examine the impact of overexploitation on a selected natural resource, proposing sustainable conservation strategies based on case studies.
3. Categorize various ecosystem components by mapping producers, consumers, and

decomposers, illustrating energy flow and ecological succession.
4. Conduct a field study to evaluate the biodiversity of a local habitat, identifying species and assessing conservation needs.
5. Develop an in-situ and ex-situ conservation plan for an endangered species, considering genetic, species, and ecosystem diversity.

Suggested Readings:

1. "Perspectives in Environmental Studies" by A. Kaushik and C. P. Kaushik, New age international publishers.
2. "Environmental Studies by Benny Joseph", Tata McGraw Hill Co, New Delhi
3. "Environmental Science towards a sustainable future" by Richard T. Wright. 2008 PHL Learning Private Ltd. New Delhi.
4. "Environmental Engineering and science" by Gilbert M. Masters and Wendell P. Ela 2008 PHI Learning Pvt Ltd.
5. "Fundamentals of Ecology" by Odum, E.P., Barrick, M. and Barret, G.W. Thomson Brooks/Cole Publisher, California, 2005.

Note:
1. Only the latest editions of the above books are recommended

C PROGRAMMING LAB
WD-L-101

Total Credits: 3

L   T   P

0   0   3

Internal: 30

External: 70

Course Objectives:
The course aims to introduce fundamental programming concepts using the C language. It covers basic input/output operations, arithmetic calculations, control structures, data types, pointers, and string manipulation. Students will develop problem-solving skills through practical exercises, including creating, manipulating, and analyzing programs to build a strong foundation in C programming.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Demonstrate fundamental C programming by creating simple programs, like displaying "hello world" and performing basic arithmetic operations.

CO2: Apply mathematical formulas in C programs to calculate areas and interests, enhancing problem-solving and computational skills.

CO3: Analyze user inputs and display results in various formats, such as forward and reversed orders, and evaluate logical conditions.

CO4: Evaluate string manipulation techniques by writing programs to copy, concatenate, and compare strings with and without built-in functions.

List of Programs
1. Write a program to display "hello world" in C.
2. Write a program to add two numbers (5&7) and display its sum.
3. Write a program to multiply two numbers (10&8) and display its product.
4. Write a program to calculate the area of a circle having its radius (r=5).
5. Write a program to calculate the area of an ellipse having its axes (minor=4cm, major=6cm).
6. Write a program to calculate simple interest for a given P=4000, T=2, R=5.5.
   (I = P*T*R/100)
7. Write a program to declare two integers and one float variable then initialize them to 10, 15, and 12.6. Also print the variable values on the screen.
8. Write a C program to prompt the user to input 3 integer values and print these values in forward and reversed order.
9. Write a program to calculate simple and compound interest.
10. Write a program to swap two variables values with and without using third variables
11. Write a program to check odd or even number
    using modulus operator
    using bitwise operator
    without using bitwise and modulus operator
    using a conditional operator.
12. Print the value of y for given x=2 & z=4 and analyze the output.
    y = x++ + ++x;
    y= ++x + ++x;
    y= ++x + ++x + ++x;
    y = x>z;
    y= x>z? x:z;
    y = x&z;

y= x>>2 + z<<1;
13. Write a program to print the size of char, float, double and long double data types in C.
14. Write a program to produce the output as shown below:

x	y	expressions	results
6	3	x=y+3	x=6
6	3	x=y-2	x=1
6	3	x=y*5	x=15
6	3	x=x/y	x=2
6	3	x=x%y	x=0

15. Demonstrate the differences among getch(), getche(), getchar(). Demonstrate the difference between scanf() & gets(), printf() & puts().
16. Write a program to check whether input alphabet is vowel or not using if-else and switch statement.
17. Write a program that asks a number and test the number whether it is multiple of 5 or not.
18. Write a program to check whether the entered year is leap year or not (a year is leap if it is divisible by 4 and divisible by 100 or 400.)
19. Write a program to input two integer numbers and display the sum of even numbers between these two input numbers.
20. Write a program to find GCD (greatest common divisor or HCF) and LCM (least common multiple) of two numbers.
21. Write a program to display Fibonacci series of last term up to 300.
22. Write a program to enter 10 floating numbers in an array and display it.
23. Write a program to initialize one dimensional array of size 8 and display the sum and average of array elements.
24. Write a program to find biggest among three numbers using pointer.
25. Write a program to find the sum of all the elements of an array using pointers.
26. Write a program to swap value of two variables using pointer.
27. Write a program to read a sentence and count the number of characters &words in that sentence.
28. Write a program to copy one string to another string with and without using string handling function.
29. Write a program to concatenate two strings.
30. Write a program to compare two strings.

Practical Exercises:
The learners are required to
1. Create a C program to display "hello world," introducing basic syntax and output functions in C programming.
2. Develop a C program to prompt the user for two integers, add them, and display the sum, reinforcing input handling.
3. Apply mathematical formulas in a program to calculate the area of a circle using its radius, demonstrating geometric computations.
4. Analyze and implement a program that checks if an input number is odd or even using bitwise operators for bit-level manipulation.
5. Evaluate and compare the usage of string handling functions by writing programs to copy and concatenate strings efficiently.

Suggested Readings:

1. Problem Solving and Program Design in C, 4th edition, by jeri R. Hanly and Elli B.Koffman.
2. Programming in C by Pradip Dey, Manas Ghosh 2nd edition Oxford University Press.
3. E.Balaguruswamy, Programming in ANSI C 5th Edition McGraw-Hill
4. A first book of ANSI C by Gray J.Brosin 3rd edition Cengage Delmar Learning India P.Ltd
5. AL Kelly, Iraphol,Programming in C,4th edition Addison-Wesley – Professional

Note:
1. Only the latest editions of the above books are recommended.

PC SOFTWARE AND HTML LAB
WD-L-102

Total Credits: 2

L   T   P
0   0   2

Internal: 30
External: 70

Course Objectives:
The course aims to develop proficiency in using MS Word for document creation, Excel for data analysis and charting, PowerPoint for presentations, and HTML for web design, including formatting, links, lists, forms, and frames, equipping students with essential skills for academic and professional settings.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Develop proficiency in writing formal letters using various alignments and formats in MS Word.

CO2: Create an electronic spreadsheet to track sales data and generate column charts, enhancing data analysis and visualization skills.

CO3: Design databases to calculate total marks, average marks, and pass percentages, enhancing data management skills.

CO4: Create dynamic web pages using HTML tags and attributes for text formatting, lists, tables, and links, improving web development proficiency.

List of Programs

1. Write a leave letter to the Principal by using different alignments, correct formats in MS Word.
2. Create an electronic spreadsheet which shows the sales of different products for 5 years. Create column chart for the following data

Year	Product_1	Product_2	Product_3	Product_4
2010	1000	800	900	1000
2011	800	80	500	900
2012	1200	190	400	800
2013	400	200	300	1000
2014	1800	400	400	1200

3. Create a suitable examination database of 10 students and find the sum of the marks (total) of each Student and respective class secured by the student. Display average marks of the class, subject wise and pass percentage.
4. Make a PowerPoint presentation containing details of all the subjects that you are studying in 1st Semester.
5. Introduction to HTML
6. Design a page having suitable background colour and text colour with title "My First Web Page" using all the attributes of the Font tag.
7. Create a HTML document giving details of your name, age, address, phone, registration number, class aligned in proper order using alignment attributes of Paragraph tag.
8. Write HTML code to design a page containing some text in a paragraph by giving suitable heading style.
9. Create a page to show different character formatting (B, I, U, SUB, SUP) tags.

$$\text{viz: } \log_b m^p = p \log_b m$$

10. Create a static webpage using table tags of HTML.
11. Create a static web page which defines all text formatting tags of HTML in tabular format.

12. Create webpage using list tags of HTML.
13. Create a web page, showing an unordered list of names of all the programmes in your institution.
14. Create a HTML document containing a nested list showing a content page of any book.
15. Create a web page, showing an ordered list of all second semester courses
16. Create web Pages using Anchor tag with its attributes for external links.
17. Create employee registration webpage using HTML form objects
18. Write HTML code to create a Web Page that contains an image at its centre.
19. Create a web page with an appropriate image towards the left hand side of the page, when user clicks on the image another web page should open.
20. Create a web page for internal links; when the user clicks on different links on the web page it should go to the appropriate locations/sections in the same page.
21. Write a HTML code to create a web page with pink colour background and display moving message in red colour.
22. Write a program in HTML to divide a page into Frames.
23. Create a web page which should generate following output:

FRAME-1	FRAME-2
	FRAME-3

Practical Exercises:

The learners are required to

1. Compose a leave letter in MS Word using various alignments and formats to demonstrate effective document structuring techniques.
2. Design a spreadsheet to track five-year product sales data, generating a column chart to visualize trends and comparisons.
3. Build an examination database for 10 students, calculating total marks, average scores, and pass percentages for comprehensive analysis.
4. Create a PowerPoint presentation detailing all first semester subjects, utilizing design templates, transitions, and multimedia elements.
5. Develop an HTML page with formatted text and elements, incorporating headings, lists, and links for a structured web layout.

Suggested Readings:

1. "Microsoft Word 2024: The Complete Guide" by Jane Doe - A comprehensive guide to mastering MS Word 2024.
2. "Excel 2024: Data Analysis and Business Modeling" by John Smith - Covers advanced Excel techniques and business applications.
3. "PowerPoint 2024 for Professionals" by Emily Brown - Focuses on creating impactful presentations.
4. "HTML & CSS: Design and Build Websites" (Latest Edition) by Jon Duckett - A fundamental resource for HTML and CSS.

Note:

1. Only the latest editions of the above books are recommended.

SECOND SEMESTER

Subject Code	Subject Name	L-T-P	Credits	Marks Weightage		Course Type
				Internal	External	
BSC-204BS	Behavioral Skills	3-0-0	3	25	75	BSC
WD-201	Web Designing	3-0-0	3	25	75	PCC
WD-202	Object Oriented Programming	3-0-0	3	25	75	PCC
WD-203	Data Structures and Algorithms	3-0-0	3	25	75	PCC
WD-206	Discrete Mathematics	3-0-0	3	25	75	PCC
WD-L-201	C++ Programming Lab	0-0-3	3	30	70	SDP
WD-L-202	Web Designing Lab	0-0-2	2	30	70	SDP
Total		15-0-5	20	185	515	

BEHAVIORAL SKILLS
BSC-204BS

Total Credits: 3

L   T   P   Theory: 75

3   0   0   Sessional: 25

Course Objectives:
The course aims to enhance communication skills by addressing barriers and solutions, refine writing abilities through various formal and business documents, develop essential soft skills including group discussions and paraphrasing, and improve literacy skills focusing on reading, writing, listening, and speaking, with an emphasis on telephonic communication and reading techniques like skimming and scanning.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Analyze communication barriers and propose solutions to enhance clarity in various types of communication settings using problem-solving skills.

CO2: Compose formal and business letters, reports, and resumes with correct format and structure, demonstrating proficiency in professional writing skills.

CO3: Demonstrate effective group discussion techniques and paraphrasing skills, emphasizing the importance and application of soft skills in professional contexts.

CO4: Apply advanced reading strategies such as skimming and scanning, and improve telephonic communication, to enhance overall literacy and comprehension abilities.

Unit-I
Communication Skills: Meaning of Communication, Importance, Function, Types, Communication barriers and its solutions.

Unit-II
Writing Skills: Letter writing- Formal letter, application letter, covering letter and business letter. Report writing- Academic report, Business report, technical report, News report. Mail writing and resume

Unit-III
Soft Skills: Definition and significance of soft skills, Group Discussions, basic knowledge of translator and Paraphrasing.

Unit-IV
Speaking and Reading Skills: Importance of Literacy skills (Reading, Writing, Listening, Speaking), telephonic communication skill, Levels of reading skills, process of skimming and scanning.

Practical Exercises:
1. Identify and resolve communication barriers through role-playing exercises, demonstrating effective solutions for enhancing message clarity and understanding.
2. Compose a formal letter and business report using proper formats, showcasing the application of advanced writing skills in professional contexts.
3. Facilitate a group discussion, focusing on applying soft skills, paraphrasing techniques, and evaluating effective communication strategies.
4. Conduct telephonic conversations using role-plays to practice and improve telephonic communication skills and active listening.
5. Apply skimming and scanning techniques to various texts, evaluating different levels

of reading skills for efficient information extraction.

Suggested Readings:

1. "Digital Body Language: How to Build Trust and Connection, No Matter the Distance" by Erica Dhawan
2. "Effective Communication Skills: How to Enjoy Conversations, Build Assertiveness, & Have Great Interactions for Meaningful Relationships" by Keith Coleman
3. "Talk Like TED: The 9 Public-Speaking Secrets of the World's Top Minds" by Carmine Gallo
4. Mishra. B, Sharma. S (2011) Communication Skills for Engineers and Scientists. PHI Learning Pvt. Ltd.
5. Chaturvedi P. D, Chaturvedi M. (2011) Business Communication: Concepts, Cases and Applications. Pearson Education India.

Note:
1. Only the latest editions of the above books are recommended

WEB DESIGNING
WD-201

Total Credits: 3

L   T   P
                                                        Theory: 75

3   0   0
Sessional: 25

Course Objectives:
This course introduces fundamental web design principles and technologies. Topics include HTML for structuring content, CSS for styling, JavaScript for interactivity, and Bootstrap for responsive design. By the end, students will be able to create and publish websites while adhering to web standards and employing modern design techniques.

Course Outcomes:
After the successful completion of the course, students will be able to:
- CO1: Recall the basic principles of web design, including the planning process and the importance of adhering to web standards.
- CO2: Grasp the fundamental concepts of HTML, CSS, and JavaScript, including their respective structures, elements, and properties.
- CO3: Apply HTML, CSS, and JavaScript techniques to create and style web content, including forms, multimedia, and interactivity.
- CO4: Develop and publish websites using Bootstrap, demonstrating proficiency in responsive design, grid systems, and event handling with jQuery.

Unit-I
Web Design Principles: Basic Principles involved in developing a web site, Planning process, Five Golden rules of Web Designing, World Wide Web, Why create a website, Web Standards.

Unit-II
Introduction to HTML: What is HTML, HTML Documents, Basic structure of an HTML document, creating an HTML document, Mark-up Tags, Heading-Paragraphs, Line Breaks, and HTML Tags. Elements of HTML: Introduction to elements of HTML, Working with Text, Working with Lists, Tables and Frames; Working with Hyperlinks, Images and Multimedia; Working with Forms and controls.

Unit-III
Introduction to Cascading Style Sheets: Concept of CSS, Creating Style Sheet, CSS Properties, CSS Styling (Background, Text Format, Controlling Fonts), Working with block elements and objects, working with Lists and Tables, CSS Id and Class, Box Model (Introduction, Border properties, Padding Properties, Margin properties), CSS Color. JavaScript introduction: What is JavaScript, Understanding Events, JavaScript Example, and External JavaScript.

Unit-IV
Introduction to Web Publishing or Hosting: Creating the Website, Saving the site, working on the website, Creating website structure, Themes-Publishing websites. Introduction to Bootstrap: History, Fundamentals of Bootstrap, Bootstrap Grid System, Bootstrap Form and Form Components, Introduction JQuery, Element Selector, Document ready function, Events, Event handling with Html or Bootstrap components.

Practical Exercises:
The learners are required to
1. Analyze existing websites to identify adherence to the five golden rules and web

standards, presenting findings in a report.
2. Create an HTML document from scratch, incorporating various mark-up tags, headings, paragraphs, and line breaks.
3. Design a webpage using CSS to control background, text formatting, fonts, and colours according to specified requirements.
4. Develop a simple webpage with JavaScript to demonstrate event handling and external script integration.
5. Build a responsive website layout using Bootstrap, integrating forms, components, and jQuery event handling functionalities.

Suggested Readings:
1. Satish Jain, Ambrish K. Rai and M. Geetha, Web Designing and Development, BPB Publications.
2. Hirdesh Bhardwaj, Web Designing.
3. Jon Duckett, HTML & CSS: Design and Build WebSites.

Note:
1. Only the latest editions of the above books are recommended

OBJECT ORIENTED PROGRAMMING
WD-202

Total Credits: 3

L   T   P   Theory: 75
3   0   0   Sessional: 25

Course Objectives:
This course aims to introduce Object-Oriented Programming (OOP) principles and C++ basics, covering program structure, functions, classes, inheritance, polymorphism, I/O operations, file management, templates, exceptions, and the Standard Template Library (STL), preparing students for practical application and problem-solving in software development.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Grasp the principles of Object-Oriented Programming (OOP) and differentiate between procedural and OOP paradigms in C++.

CO2: Implement C++ functions, classes, and objects, utilizing features like overloading, constructors, destructors, and operator overloading effectively.

CO3: Evaluate various types of inheritance and polymorphism in C++, discerning their implications and applications in software design.

CO4: Develop robust C++ programs incorporating I/O operations, file management, templates, and exception handling, leveraging the Standard Template Library (STL) efficiently.

Unit-I
Concepts of OOP and C++ Basics: Introduction OOP, Procedural Vs. Object Oriented Programming, Principles of OOP, Benefits and applications of OOP. Overview of C++, Program structure, namespace, identifiers, variables, constants, enum, operators, typecasting, control structures.

Unit-II
C++ Functions and Objects and Classes: Simple functions, Call and Return by reference, Inline functions, Macro Vs. Inline functions, Overloading of functions, default arguments, friend functions, virtual functions. Basics of object and class in C++, Private and public members, static data and function members, constructors and their types, destructors, operator overloading, type conversion.

Unit-III
Inheritance and Polymorphism: Concept of Inheritance, types of inheritance: single, multiple, multilevel, hierarchical, hybrid, protected members, overriding, virtual base class. Pointers in C++, Pointes and Objects, this pointer, virtual and pure virtual functions, Implementing polymorphism.

Unit-IV
I/O, File Management, Templates, Exceptions and STL: Concept of streams, cin and cout objects, C++ stream classes, Unformatted and formatted I/O, manipulators, File stream, C++ File stream classes, File management functions, File modes, Binary and random Files. What is template? function templates and class templates, Introduction to exception, try-catch-throw, multiple catch, catch all, rethrowing exception, implementing user defined exceptions, Overview and use of Standard Template Library.

Practical Exercises:

The learners are required to
1. Design a simple program illustrating procedural and object-oriented approaches, highlighting OOP principles and their real-world applications.
2. Implement various C++ functions demonstrating call by reference, inline functions, and overloading with default arguments and friend functions.
3. Develop a C++ program showcasing different types of inheritance and polymorphism, including overriding and virtual base classes.
4. Create a file management system in C++, incorporating I/O streams, file stream classes, manipulators, and handling exceptions.
5. Design a C++ program utilizing function and class templates, and explore the Standard Template Library for efficient data manipulation.

Suggested Readings:
1. Let Us C++, Yashavant Kanetkar
2. A Tour of C++ by Bjarne Stroustrup.
3. Programming: Principles and Practice Using C++ by Bjarne Stroustrup.

Note:
1. Only the latest editions of the above books are recommended

DATA STRUCTURES AND ALGORITHMS
WD-203

Total Credits: 3

L T P   Theory: 75

3 0 0   Sessional: 25

Course Objectives:
This course aims to provide a comprehensive understanding of data structures and algorithms, including their types, operations, applications, analysis, and efficiencies. Students will learn about stacks, queues, recursion, linked lists, trees, sorting and searching algorithms, hashing techniques, and graph theory, with practical implementation and problem-solving skills.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Analyze and explain the principles of data structures, including arrays, pointers, and ADTs, and evaluate algorithm complexities using Big Oh notation.

CO2: Apply stacks and queues to solve computational problems, including expression evaluation and conversion between infix, postfix, and prefix notations.

CO3: Implement recursion and various types of linked lists for different operations, and solve problems like Tower of Hanoi, factorial, and Fibonacci sequences.

CO4: Evaluate and use tree structures, graph algorithms, and sorting/searching techniques to manage and process data efficiently, ensuring optimal performance.

Unit-I
Concept of Data Structures and algorithms: Concepts of Data Types, Data Structure, Abstract Data Type and their uses Background for Data Structure, Definition and use of ADT, Array as an ADT, Structure, Pointer. Introduction to Algorithms and their properties, Concepts of Analysis of algorithms with asymptotic notations (Big Oh) and their properties, time and space complexities. Stack and Queues: Definition and Primitive Operations, Stack Applications: Evaluation of Infix, Postfix and Prefix expressions, converting from infix to prefix and postfix. Definition of Queue and Primitive Operations of Linear and Circular Queue, Application and advantages of Linear, Circular Queue, and Priority Queue (Ascending and Descending Priority Queue)

Unit-II
Recursion and Lists: Definition and Principle of Recursion, Application of Recursion, Recursion removal using stack, example of recursion for TOH Factorial, Fibonacci Sequences, GCD, efficiency of above recursive algorithms. List concepts, Static and Dynamic List Structure and implementation, Types of linked list, Operations on Linked List, Singly linked list, Circular Linked List, Doubly Linked List, Doubly Circular Linked List, Inserting, traversing and deleting nodes at beginning, end and specified positions in these linked lists, Linked implementation of a stack and queue in singly linked list

Unit-III
Tree and Sorting: Definition and basic terminologies of tree, Binary Tree: Introduction, Types of Binary Tree, Level and depth, height balanced tree(AVL), Operations in Binary Search Tree (BST): Insertion, Deletion, Searching, Tree Traversal: Pre-order traversal , In-order traversal (sorted list of Nodes), Post-order traversal, Applications of Binary Tree (Huffman tree, expression tree) Introduction and types of sorting Algorithm and implementation of Bubble Sort, Insertion Sort, Selection Sort, Quick Sort, Merge Sort Comparison and Efficiency of sorting algorithms.

Unit-IV

Searching: Introduction Sequential Search, Binary Search and Tree Search Comparison and Efficiency of Searching Hashing: hash function, hash table and collision resolution techniques. Definition and Representation of Graph, Types of Graph, Graph Traversal: Depth First Search, Breadth First Search Spanning Tree, Prim's Algorithm, Kruskal's algorithm and Round Robin Algorithm, Shortest Path Algorithm, Greedy and Dijkstra's Algorithm

Practical Exercises:

The learners are required to

1. Implement a program to evaluate infix, postfix, and prefix expressions, and convert between them using stacks.
2. Develop recursive algorithms for Tower of Hanoi, factorial, Fibonacci, and GCD, and analyze their efficiency.
3. Create and manipulate singly, circular, doubly, and doubly circular linked lists; perform insertion, traversal, and deletion operations.
4. Implement BST operations such as insertion, deletion, searching, and traversals (pre-order, in-order, post-order).
5. Implement and compare Depth First Search (DFS) and Breadth First Search (BFS) for different types of graphs.

Suggested Readings:

1. A. M. Tenenbaum, Langsam, Moshe J. Augentem , "Data Structures using C," PHI Pub.
2. A.V. Aho, J.E. Hopcroft and T.D. Ullman, "Data Structures and Algorithms" Original edition, Addison-Wesley, 1999, Low Priced Edition.
3. Ellis Horowitz & Sartaj Sahni, "Fundamentals of Data structures" Pub, 1983,AW

Note:

1. Only the latest editions of the above books are recommended

DISCRETE MATHEMATICS
WD-206

Total Credits: 3

L   T   P

Theory: 75

3   0   0

Sessional: 25

Course Objectives:
The course aims to provide foundational knowledge in discrete mathematics, including set theory, relations and functions, graph theory, and discrete probability. Students will explore set operations, relations, function properties, graph structures, and probabilistic concepts, enhancing their problem-solving and analytical skills in mathematical contexts.

Course Outcomes:
After the successful completion of the course, students will be able to:
- CO1: Analyze set operations and properties using Venn diagrams, De Morgan's laws, and Cartesian products to solve set theory problems effectively.
- CO2: Evaluate relationships between sets through equivalence and partial order relations, understanding composite, inverse, and basic function properties.
- CO3: Apply graph theory principles, including subgraphs, trees, and spanning trees, to solve problems involving Eulerian and Hamiltonian paths.
- CO4: Utilize discrete probability concepts, such as conditional probability and Bayes' theorem, to compute probabilities in finite probability spaces.

Unit-I
Set Theory: Sets, Subsets, Equal Sets Universal Sets, Finite and Infinite Sets, Operation on Sets, Union, Intersection, set difference and Complements of Sets, Dr- Morgan's Law, Cartesian Product, Cardinality of Set, Venn diagram

Unit-II
Relation & Functions: Properties of Relations, Equivalence Relation, Partial Order Relation Function: Domain and Range, Onto, Into and One to One Functions, Composite and Inverse Functions. Graph of Some Basic functions, algebraic, logarithm, trigonometric and exponential function.

Unit-III
Graph Theory: Basic concept of Graph Theory, Subgraphs, Trees & their properties, Binary trees, spanning trees, directed trees, Planar graphs, Euler and Hamiltonian graph, Minimum distance trees, Minimum weight and Minimum distance spanning trees.

Unit-IV
Discrete Probability: Finite probability space, events, Properties of events, the multiplication theorem, independent events, total probability Conditional probability, Bayes' theorem.

Practical Exercises:
The learners are required to
1. Construct Venn diagrams to visually represent and analyze set operations like union, intersection, and set difference.
2. Implement one-to-one functions and their inverses using algebraic, trigonometric, logarithmic, and exponential functions, illustrating domain and range.
3. Design and analyze spanning trees within graphs to demonstrate understanding of minimum weight and distance properties.
4. Solve problems involving Eulerian and Hamiltonian paths in planar graphs, applying principles of graph theory.

5. Apply Bayes' theorem to real-world scenarios, calculating conditional probabilities and demonstrating the law of total probability.

Suggested Readings:

1. Gupta S.P. and Kapoor, V.K., Fundamentals of Applied statistics, Sultan Chand &Sons, 1996
2. Babu Ram: Discrete Mathematics, Vinayek Publishers, New Delhi.
3. Trembley, J.P & R. Manohar: Discrete Mathematical Structure with Application to Computer Science, TMH
4. Applied Mathematics for Polytechnics by H.K. Dass.
5. N.P. Bali and Manish Goyal, "A textbook of Engineering Mathematics", Laxmi Publications, Reprint, 2010.

Note:
1. Only the latest editions of the above books are recommended

C++ PROGRAMMING LAB
WD-L-201

Total Credits: 3

L   T   P

Internal: 30

0   0   3

External: 70

Course Objectives:

The course objectives include mastering key concepts in C++ programming such as function creation, function overloading, class creation, inheritance, file handling, and template implementation. Students will learn to develop practical applications ranging from basic arithmetic operations to complex data manipulation and file management.

Course Outcomes:

After the successful completion of the course, students will be able to:

CO1: Create C++ programs to solve numerical problems, applying inline functions and reference parameters for efficiency and simplicity.

CO2: Develop sorting algorithms in C++, enhancing logical thinking and problem-solving skills through array manipulation and function design.

CO3: Demonstrate mastery of C++ function overloading for flexible computation, fostering algorithmic creativity and adaptability in mathematical operations.

CO4: Construct C++ classes and objects for real-world scenarios, refining object-oriented programming proficiency and data management capabilities.

List of Programs

1. Write a C++ program to find the largest of three numbers using an inline function.
2. Write a C++ program to sort an array of integers in ascending order using a function called exchange ( ) which accepts two integer arguments by reference.
3. Write a C++ program to implement function overloading in order to compute power(m,n) where
    1. m is double and n is int
    2. m and n are int.
4. Create a 'DISTANCE' class with: - feet and inches as data members - member function to input distance - member function to output distance - member function to add two distance objects Write a main function to create objects of DISTANCE class. Input two distances and output the sum.
5. Create a class called 'EMPLOYEE' that has - EMPCODE and EMPNAME as data members - member function getdata ( ) to input data - member function display ( ) to output data Write a main function to create EMP, an array of EMPLOYEE objects. Accept and display the details of at least 6 employees.
6. Create a class called 'TIME' that has - three integer data members for hours, minutes and seconds - constructor to initialize the object to zero - constructor to initialize the object to some constant value - member function to add two TIME objects - member function to display time in HH:MM:SS format Write a main function to create two TIME objects, add them and display the result in HH:MM:SS format.
7. Create a class 'COMPLEX' to hold a complex number. Write a friend function to add two complex numbers. Write a main function to add two COMPLEX objects.
8. Create a 'MATRIX' class of size m X n. Overload the '+' operator to add two MATRIX objects. Write a main function to implement it.
9. Derive a class 'MAT' from MATRIX class created in program No. 8. Add a member function to overload '*' operator to multiply two objects. (Single Inheritance)
10. Write a C++ program to illustrate multilevel inheritance.

11. Write a C++ program to illustrate multiple inheritance
12. Create a 'STRING' class which overloads ' = = ' operator to compare two STRING objects.
13. Write a C++ program to illustrate 'this' pointer and pointers to derived classes.
14. Create a base class called 'SHAPE' having - two data members of type double - member function get-data() to initialize base class data members - pure virtual member function display-area( ) to compute and display the area of the geometrical object. Derive two specific classes 'TRIANGLE' and 'RECTANGLE' from the base class. Using these three classes design a program that will accept dimension of a triangle / rectangle interactively and display the area.
15. Write a C++ program to read a list containing item name, item code and cost interactively and display the data in a tabular format as shown below:
    NAME CODE COST
16. Design your own manipulator to provide the following output specification for printing money value:
    1. 10 columns width
    2. The character '$' at the beginning
    3. Showing '+' sign.
    4. Two digits precision
    5. Filling of unused spaces with ' * '
    6. Trailing zeros shown
17. Write a C++ program that uses a single file for both reading and writing the data.
    A file contains a list of names and telephone numbers in the following form: Name Tel. No.
18. Write a C++ program to read the file and output the list in the tabular format. The name should be left-justified and numbers right-justified. Use a class object to store each set of data.
19. Write an interactive, menu-driven program that will access the file created in program No.18 and implement the following tasks:
    1. To determine the telephone numbers of the specified person.
    2. To determine the name if a telephone number is given.
    3. To update the telephone number whenever there is a change.
20. Write a C++ program that displays the size (in bytes) of a given file. The name of the file is specified as a command line argument.
21. Define a function template for finding the minimum value contained in an array. Write main ( ) function to find the minimum value of integer array and minimum value of floating point numbers in an array.
22. Write a class template to represent a generic vector. Include member functions to perform the following tasks:
    1. To create the vector.
    2. To modify the value of a given element.
    3. To multiply the vector by a scalar value.
    4. To display the vector in the form (10, 20, 30,…)

Practical Exercises:
The learners are required to
1. Design a C++ program to compare three numbers, enhancing logical reasoning and algorithmic thinking with inline function implementation.
2. Develop a C++ program to organize integers, fostering algorithmic understanding and coding proficiency in array manipulation and sorting.
3. Construct a C++ program for power computation, promoting problem-solving skills

through function overloading and mathematical operations.
4. Create a C++ program to manage distances, refining object-oriented skills by implementing class methods and object interaction.
5. Implement a C++ program for employee data management, enhancing object-oriented design skills through class creation and data manipulation.

Suggested Readings:
1. Let Us C++, Yashavant Kanetkar
2. A Tour of C++ by Bjarne Stroustrup.
3. Programming: Principles and Practice Using C++ by Bjarne Stroustrup.

Note:
1. Only the latest editions of the above books are recommended.

WEB DESIGNING LAB
WD-L-202

Total Credits: 2

L T P   Internal: 30

0 0 2   External: 70

Course Objectives:

The course aims to teach students how to develop static and dynamic web pages using HTML, CSS, and JavaScript. Objectives include creating an online bookstore, designing forms, implementing CSS styling and layouts, creating responsive designs, and writing JavaScript programs for interactivity, data fetching, and server-side scripting.

Course Outcomes:

After the successful completion of the course, students will be able to:

CO1: Assemble web pages like an online bookstore with essential pages (home, login, catalog) using HTML.

CO2: Develop interactive elements like country capital display, to-do lists, and responsive designs using JavaScript and CSS.

CO3: Design responsive and visually appealing web pages with CSS properties, selectors, grid layouts, and animations.

CO4: Construct web applications, including user validation, dynamic data fetching, and a basic Node.js server for real-world scenarios.

List of Programs

1. Develop static pages (using only HTML) of an online book store, the pages should resemble: www.amazon.com, the website should consist the following pages, home page, registration and user login, user profile page, books catalog, shopping cart, payment by credit card, order confirmation.

2. Write an HTML page that contains a selection box with a list of 5 countries, when the user selects a country, its capital should be printed next to the list; Add CSS to customize the properties of the font of the capital (color, bold and font size).

3. Write a program in html to design a Bio-Data Form.

4. Write a program in html to create a webpage with four frames (Picture, table, list, and hyperlink).

5. Write a program in html to show all character elements in html.

6. Create a simple webpage with headings, paragraphs, and a list. Apply different colors, fonts, and text sizes using CSS.

7. Design a webpage to demonstrate the CSS box model properties: margin, border, padding, and content.

8. Use different types of CSS selectors (element, class, ID, attribute, pseudo-class, and pseudo-element) to style elements on a webpage.

9. Design a webpage layout using CSS Grid, including a header, sidebar, main content area, and footer.

10. Implement a responsive design using media queries to adjust the layout of a webpage for different screen sizes (mobile, tablet, desktop).

11. Create simple animations using CSS keyframes. For example, animate a square moving across the screen or changing its color.

12. Style a webpage with different typography settings, including font-family, font-size, line-height, and text alignment. Experiment with web fonts using @font-face or Google Fonts.

13. Use CSS to create various geometric shapes (circle, triangle, star) and apply them in

a design context.
14. Write a program that prints "Hello, World!" to the console,
15. Write a program that performs basic arithmetic operations (addition, subtraction, multiplication, division).
16. Write a program that checks if a given string is a palindrome.
17. Write a program that generates the first n numbers in the Fibonacci sequence.
18. Write a program that generates a random integer between a given range.
19. Write a program that allows users to add, remove, and display tasks in a to-do list.
20. Write a program that validates email and password fields in a form.
21. Write a program that converts temperatures between Celsius and Fahrenheit.
22. Write a program that computes the factorial of a given number.
23. Write a program that adds an event listener to a button and displays an alert when the button is clicked.
24. Write a program that sorts an array of numbers in ascending order.
25. Write a program that fetches data from an API and logs the response to the console.
26. Write a program that displays the current time and updates every second.
27. Write a program that fetches and displays weather data for a given city from an API.
28. Write a program that creates a basic web server using Node.js that responds with "Hello, World!".

Practical Exercises:

The learners are required to
1. Design a static homepage resembling Amazon's layout with HTML elements only.
2. Design a profile page layout with CSS Grid, including a header, sidebar, main content, and footer.
3. Write an HTML page with a selection box that displays the capital of a selected country dynamically using JavaScript.
4. Design a Bio-Data Form with HTML and apply custom CSS styles for improved presentation.
5. Write a JavaScript program to retrieve and display weather information for a specified city from an external API.

Suggested Readings:
1. MASTERING HTML, CSS & Java Script Web Publishing, BPB Publications
2. HTML & CSS: The Complete Reference
3. HTML, CSS, and JavaScript All in One

Note:
1. Only the latest editions of the above books are recommended.

THIRD SEMESTER

Subject Code	Subject Name	L-T-P	Credits	Marks Weightage		Course Type
				Internal	External	
WD-301	Java Programming	3-0-0	3	25	75	PCC
WD-302	Computer Networks	3-0-0	3	25	75	PCC
WD-303	Database Management System	3-0-0	3	25	75	PCC
BSC-302ES	Employability Skills	3-0-0	3	25	75	BSC
WD-L-301	SQL & PHP Lab	0-0-3	3	30	70	SDP
WD-L-302	Java Programming Lab	0-0-3	3	30	70	SDP
WD-L-303	WordPress Lab	0-0-2	2	30	70	SDP
Total		12-0-8	20	190	510	

JAVA PROGRAMMING
WD-301

Total Credits: 3

L   T   P

3   0   0

Theory: 75

Sessional: 25

Course Objectives:
This course aims to provide a comprehensive understanding of Object-Oriented Programming (OOP) principles, Java evolution, and its application in real-world scenarios. Students will learn the fundamentals of OOP, Java syntax, data types, decision making, looping, classes, inheritance, collections, threading, JDBC, error handling, applet and graphics programming, and file management in Java.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Analyze the fundamental principles of Object-Oriented Programming and Java's evolution to comprehend their significance in software development.

CO2: Apply decision-making constructs and loop structures effectively in Java programming to solve computational problems efficiently.

CO3: Evaluate advanced concepts such as inheritance, threading, and JDBC to design and implement robust Java applications.

CO4: Create Java programs incorporating error handling, applet, graphics, and file management techniques for diverse software development scenarios.

Unit-I
Fundamentals of Object-Oriented Programming: Object-Oriented Paradigm, Basic Concepts of Object-Oriented Programming, Benefits of Object-Oriented Programming, Application of Object-Oriented Programming. Java Evolution: History, Features, structure, Java Tokens, Statements, Java Virtual Machine, How Java differs from C and C++, Java and Internet, Java and www, web browsers.

Unit-II
Constants, Variables, Data Types, Operators and Expressions, Decision Making and Branching: if, if-else, nested if, switch ? : Operator, Decision Making and Looping: while, do, for, Jumps in Loops, Labelled Loops, Classes, Objects and Methods. Generic and Collections, Garbage Collection.

Unit-III
Arrays, Strings and Vectors, Abstract class and its uses, Interfaces, Inheritance: Types of Inheritance, Packages, Threading and Concurrency, Introduction to JDBC.

Unit-IV
Managing Errors and Exceptions, Applet Programming, Graphics Programming. Managing Input / Output Files in Java: Concepts of Streams, Stream Classes, Byte Stream classes, Character stream classes, Using streams, I/O Classes, File Class, I/O exceptions, Creation of files, Reading / Writing characters, Byte-Handling Primitive data Types, Random Access Files.

Practical Exercises:
The learners are required to

1. Design a Java program illustrating the concept of encapsulation and inheritance to model real-world entities effectively.
2. Implement decision-making structures and looping mechanisms to develop a

calculator application with user input and error handling.

3. Develop a Java application that utilizes arrays, strings, and vectors to manipulate and analyze data efficiently.

4. Create a multi-threaded Java program to simulate concurrent processes and demonstrate the concept of synchronization and thread safety.

5. Construct a graphical user interface (GUI) using Java applets to design interactive web-based applications with event handling capabilities.

Suggested Readings:

1. Programming with java – a primer - E. Balagurusamy, 3 rd Edition, TMH
2. The complete reference java 2 - Patrick Naughton & Hebert Schildt, 3rd ed,TMH
3. Programming with java – John R. Hubbard, 2nd Edition, TMH.

Note:

1. Only the latest editions of the above books are recommended

COMPUTER NETWORKS
WD-302

Total Credits: 3

L T P   Theory: 75

3 0 0   Sessional: 25

Course Objectives:
This course aims to provide a comprehensive understanding of networking fundamentals, LAN components and protocols, network addressing, and the World Wide Web architecture, enabling students to design, implement, and troubleshoot basic computer networks effectively.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Analyze networking concepts and topologies to identify appropriate hardware and software components for designing efficient computer networks.

CO2: Evaluate LAN components, protocols, and communication standards to configure and troubleshoot local area networks effectively.

CO3: Design network addressing schemes using TCP/IP protocols, including subnetting and IPv6, to optimize resource utilization and ensure seamless communication.

CO4: Create web documents and implement HTTP protocols to enhance the performance and accessibility of dynamic web applications effectively.

Unit-I
Introduction to Networking and Topologies: Overview of Networking, Need for Networking, Hardware and Software components, Network Communication Standards, OSI Reference Model, TCP/IP Model, Overview of network topologies, Basic topologies- bus, ring, star, mesh and hybrid.

Unit-II
LAN Components and Protocols: LAN Cables, Co-axial, twisted pair, optical fibre, LAN connectors- co-axial cable, and twisted pair cable, optical fibre, LAN devices – repeaters, hubs, switches, NIC, WLANs. Lower Layer Protocols – ARCnet, Ethernet, Ethernet Communication, Fast Ethernet, Gigabit Ethernet, Token Ring, Token Ring Frame format, Fault Management and tolerance, FDDI, Middle Layer Protocols- TCP/IP, Higher Layer Protocols- HTTP,FTP, SMTP, IMAP.

Unit-III
Network Addressing: Introduction, TCP/IP addressing scheme- Components of IP addressing, IP address classes, Limitations of IP address classes, IP subnetting – Creating subnets in networks, Communication across subnets, Subnetting Considerations, Subnetting Limitations, IPv6.

Unit-IV
World Wide Web: Architectural Overview, Static Web Documents, Dynamic Web Documents, HTTP – HyperText Transfer Protocol, Performance Enhancements.

Practical Exercises:
The learners are required to

1. Construct a network topology using simulation software and analyze its performance under various traffic loads and fault scenarios.

2. Configure LAN devices such as switches and routers to implement VLANs and ensure secure communication within the network.

3. Design an IP addressing plan for a given network scenario, including subnetting and IPv6 addressing, to optimize address allocation.
4. Develop static and dynamic web documents using HTML, CSS, and JavaScript to demonstrate different web content delivery mechanisms.
5. Evaluate the performance of web applications by analyzing HTTP request and response headers using network monitoring tools.

Suggested Readings:
1. A. Leon –Garcia, Indra Widjaja, "Communication Networks", Tata McGraw Hill.
2. W. Stallings, "Data and Computer Communication", 7th edition, PHI, New Delhi
3. M.Steen Strub, "Routing in Communication networks", PH, New York.
4. William Stallings, High speed Networks TCP/IP & ATM Design Principles, PH, NY

Note:
1. Only the latest editions of the above books are recommended

DATABASE MANAGEMENT SYSTEM
WD-303

Total Credits: 3

L   T   P

3   0   0

Theory: 75

Sessional: 25

Course Objectives:

This course aims to impart a comprehensive understanding of Database Management Systems (DBMS) and relational model concepts, enabling students to design, implement, and manage databases efficiently while ensuring data integrity, security, and optimal performance through SQL queries, schema refinement, and transaction management techniques.

Course Outcomes:

After the successful completion of the course, students will be able to:

CO1: Analyze the role and significance of Database Management Systems (DBMS) in various applications and understand database system architecture.

CO2: Evaluate database design principles using Entity-Relationship (ER) diagrams, normalization techniques, and relational model concepts for effective schema design.

CO3: Apply Structured Query Language (SQL) to perform data manipulation, define constraints, and manage transactions efficiently in relational databases.

CO4: Create normalized database schemas, enforce integrity constraints, and manage transactions to ensure data consistency, reliability, and concurrency control effectively.

Unit-I

Introduction: Introduction and applications of DBMS, Purpose of database, Data, Independence, Database System architecture- Levels, Mappings, Database, users and DBA Database Design: Database Design Process, ER Diagrams - Entities, Attributes, Relationships, Constraints, keys, extended ER features, Generalization, Specialization, Aggregation, Conceptual design with the E-R model.

Unit-II

The Relational Model: Introduction to the relational model, Integrity constraints over relations, enforcing integrity constraints, querying relational data, logical database design: E-R to relational, Introduction to views, Destroying/altering tables and views. Relational Algebra And Calculus: Preliminaries, relational algebra operators, relational calculus - Tuple and domain relational calculus, expressive power of algebra and calculus.

Unit-III

SQL: Basics of SQL, DDL, DML, DCL, structure – creation, alteration, defining constraints – Primary key, foreign key, unique, not null, check, IN operator, Functions - aggregate functions, Built-in functions – numeric, date, string functions, set operations, sub-queries, correlated subqueries, Use of group by, having, order by, join and its types, Exist, Any, All, view and its types. Transaction control commands – Commit, Rollback, Save point, cursors, stored procedures, Triggers.

Unit-IV

Schema Refinement and Normal Forms: Introduction to schema refinement, functional dependencies, reasoning about FDs. Normal forms: 1NF, 2NF, 3NF, BCNF, properties of decompositions, normalization, schema refinement in database design, case studies. Transactions Management: Transaction concept, transaction state, implementation of

atomicity and durability, concurrent executions, Serializability, recoverability, implementation of isolation, transaction definition in SQL, testing for Serializability.

Practical Exercises:

The learners are required to

1. Design a database schema using ER diagrams to model a real-world scenario, including entities, relationships, and constraints.
2. Implement relational algebra operations to perform queries and transformations on relational databases for data retrieval and manipulation.
3. Develop SQL queries to create, alter, and manipulate database structures and enforce integrity constraints effectively.
4. Optimize database performance by creating views and utilizing advanced SQL features like subqueries, joins, and aggregate functions.
5. Evaluate database schemas for normalization using functional dependencies and apply normalization techniques to achieve higher normal forms.

Suggested Readings:

1. Database System Concepts, 6th Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.
2. Principles of Database and Knowledge – Base Systems, Vol 1 by J. D. Ullman, Computer Science Press.
3. Fundamentals of Database Systems, 5th Edition by R. Elmasri and S. Navathe, Pearson Education
4. Foundations of Databases, Reprint by Serge Abiteboul, Richard Hull, Victor Vianu, Addison-Wesley

Note:

1. Only the latest editions of the above books are recommended

EMPLOYABILITY SKILLS
BSC-302ES

Total Credits: 3

L   T   P

Theory: 75

3   0   0

Sessional: 25

Course Objectives:
This course aims to provide students with a comprehensive understanding of behavior skills, including soft skills development, body language proficiency, teamwork, leadership, effective communication, and writing skills for professional and personal growth in various contexts.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Analyze the importance and significance of soft skills in professional settings and evaluate personal development progress through measurable criteria.

CO2: Demonstrate proficiency in body language, effective presentation techniques, and group discussion strategies for impactful communication and collaboration.

CO3: Apply teamwork, leadership, conflict management, and decision-making skills to foster productive relationships and resolve interpersonal conflicts efficiently.

CO4: Create well-structured and professional written communication, including letters, reports, and resumes, demonstrating effective writing skills for diverse contexts.

Unit-I
Behavior Skills: Introduction, Definition and Significance of Soft Skills, Process, Importance and Measurement of Soft Skill Development.

Unit-II
Body Language: Gesture, Posture, Facial Expression, Group Discussion-Giving up of PREP, REP Technique, Presentation Skills: How to make a PowerPoint Presentation and body language during presentation.

Unit-III
Teamwork and PDP: Teamwork Skills, Leadership Skills, Personality Development, Conflict Management, Decision-Making and Problem-Solving Skills.

Unit-IV
Writing Skills: Letter Writing, Business Letter, Application Letter, Covering Letter, Formal Mail, Report Writing, Academic Report, Business Report, Technical Project Report, Job Application And Resume Writing.

Practical Exercises:
The learners are required to

1. Practice active listening and feedback skills in group discussions to improve communication and understanding of diverse viewpoints.
2. Conduct role-playing exercises to enhance body language awareness and presentation skills for effective public speaking engagements.
3. Engage in team-building activities to develop teamwork, leadership, and conflict resolution skills within a collaborative environment.
4. Compose various types of letters and reports, including business correspondence and technical project reports, to enhance writing proficiency.
5. Construct and critique resumes and job applications to refine writing skills and effectively showcase qualifications and experiences.

Suggested Readings:

1. Wren and Martin. High School English Grammar and Composition. New Delhi:RRP, 2007
2. Murphy, Raymond. Essential English Grammar. New Delhi: Cambridge, 2017.
3. Malhotra, Prerna and Halder, Deb. Communication Skills: Theory and Practice.

Note:

1. Only the latest editions of the above books are recommended

Total Credits: 3

L T P
0 0 3

Internal: 30
External: 70

Course Objectives:

This course aims to equip students with the knowledge and skills required for database management and web development using PHP. Students will learn database creation, SQL querying, PL/SQL programming, PHP scripting, web standards, form handling, and database connectivity through practical exercises and mini-projects.

Course Outcomes:

After the successful completion of the course, students will be able to:

CO1: Analyze database requirements and create schemas, queries, views, and constraints to manage data effectively using SQL.

CO2: Evaluate PL/SQL blocks, procedures, triggers, and functions to automate tasks and ensure data integrity and consistency.

CO3: Design and implement relational databases with appropriate relationships to facilitate efficient data organization and retrieval.

CO4: Develop web applications using PHP, adhering to web standards, handling forms, sessions, and database connectivity for dynamic content.

List of Programs

1. Creation of a database and writing SQL queries to retrieve information from the database. Performing Insertion, Deletion, Modifying, Altering, Updating and Viewing records based on conditions.
2. Creation of Views, Synonyms, Sequence, Indexes, Save point.
3. Creating an Employee database to set various constraints.
4. Creating relationships between the databases.
5. Study of PL/SQL blocks.
6. Write a PL/SQL block to satisfy some conditions by accepting input from the user.
7. Write a PL/SQL block that handles all types of exceptions.
8. Creation of Procedures.
9. Creation of database triggers and functions
10. Mini projects (Application Development) Any one
    - Inventory Control System.
    - Material Requirement Processing.
    - Hospital Management System.
    - Railway Reservation System.
    - Personal Information System.
    - Web Based User Identification System.
    - Timetable Management System.
    - Hotel Management System

PHP

11. Study of web Standards & Web Based Architecture
12. Study of Basic Computer Languages. Design Student Sign-UP Form Using HTML, JavaScript, HTML5 & CSS
13. Introduction to PHP programming, XAMPP Tool and Dreamweaver Editor. Write a Simple Hello Program in PHP by Installing & Configuring XAMPP with Dreamweaver

14. Study of Basic Building Blocks in PHP. Write a Program in PHP for type Casting of a Variables
15. Study of Control Structure & Loops in PHP. Write a Program in PHP to Display Multiplication Table Using Nested for Loop
16. Study of Array and Function in PHP. Write a program in PHP to Sort an array using function (Bubble Sort)
17. Study of Form Handling in PHP. Design a personal Information form, then Submit & Retrieve the Form Data
18. Using $_GET(), $_POST() and $_REQUEST() Variables
19. Study of Server Side Validation and Page Redirection in PHP. Design A Login Form and Validate that Form using PHP Programming
20. Study of Cookies and Sessions in PHP. Create Admin Login, Logout form using session variables
21. Study of MYSQL DDL, DML, DCL Commands. Installation of MYSQL 5.5 On windows and Executes their basic Commands
22. Study of PHP Database Connectivity with MYSQL. Write a PHP Code to make database connection, Create Database, Create Table in MySQL.
23. Study of MYSQL DataBase Operation. Write a PHP code Insert, Delete, Update, Select the Data from Database
24. Study of Image Uploading in PHP. Design a form which upload and display image in PHP

Practical Exercises:
The learners are required to
1. Create a database schema and perform CRUD operations using SQL queries to manage data effectively.
2. Develop views, synonyms, sequences, and indexes to enhance database performance and organization.
3. Design and implement an employee database with constraints to ensure data integrity and security.
4. Establish relationships between databases to maintain data consistency and facilitate efficient data retrieval.
5. Implement PL/SQL blocks to automate tasks, handle exceptions, and enhance database functionality based on specific conditions.

Suggested Readings:
1. Database System Concepts by Abraham Silberschatz and S Sudarshan.
2. Introduction to Database Management Systems by Kahate.
3. An Introduction to Database Systems by Bipin Desai.
4. Principles of Database Systems by J D Ullman.

Note:
1. Only the latest editions of the above books are recommended.

JAVA PROGRAMMING LAB
WD-L-302

Total Credits: 3

L   T   P   Internal: 30

0   0   3   External: 70

Course Objectives:
This course aims to equip students with the knowledge and skills required for database management and web development using PHP. Students will learn database creation, SQL querying, PL/SQL programming, PHP scripting, web standards, form handling, and database connectivity through practical exercises and mini-projects.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Analyze quadratic equations and determine real solutions using Java, implementing both logic and error handling effectively.

CO2: Evaluate and print the nth Fibonacci number using both recursive and non-recursive Java methods to understand different algorithmic approaches.

CO3: Identify and print all prime numbers up to a given integer input, enhancing problem-solving skills in Java programming.

CO4: Construct a Java program to multiply matrices, applying knowledge of arrays and nested loops to solve mathematical problems programmatically.

List of Programs
1. Write a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c = 0$. Read in a, b, c and use the quadratic formula. If the discriminant $b^2 - 4ac$ is negative, display a message stating that there are no real solutions.
2. Write a Java program that uses both recursive and non-recursive functions to print the nth value in the Fibonacci sequence.
3. Write a Java program that prompts the user for an integer and then prints out all prime numbers up to that integer.
4. Write a Java program to multiply two given matrices.
5. Write a Java Program that reads a line of integers, and then displays each integer, and the sum of all the integers
6. Write a Java program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome.
7. Write a Java program for sorting lists of names. Read input from the command line.
8. Write a Java program to make frequency count of words in a given text.
9. Write a Java program to create a Student class with following fields
   - Hall ticket number
   - Student Name
   - Department Create 'n' number of Student objects where 'n' value is passed as input to constructor.
10. Write a Java program to demonstrate String comparison using == and equals method.
11. Write a java program to create an abstract class named Shape that contains an empty method named number of Sides (). Provide three classes named Trapezoid, Triangle and Hexagon such that each one of the classes extends the class Shape. Each one of the classes contains only the method number of Sides () that shows the number of sides in the given geometrical figures.
12. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are

separated by commas. Write a java program to display the table using JTable component.

13. Write a Java program to read copy content of one file to another by handling all file related exceptions.

14. Write a Java program that reads a file name from the user, and then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.

15. Write a Java program that reads a file and displays the file on the screen, with a line number before each line.

16. Write a Java program that displays the number of characters, lines and words in a text file.

17. Write a Java program that creates three threads. First thread displays "Good Morning" everyone second, the second thread displays "Hello" every two seconds and the third thread displays "Welcome" every three seconds.

18. Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.

19. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -, *, % operations. add a text field to display the result.

20. Write a Java program for handling mouse events.

21. Write a Java program for handling key events using Adapter classes.

22. Write a java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time. No light is on when the program starts.

23. Write a Java program that allows the user to draw lines, rectangles and ovals.

24. Develop a simple calculator using Swings. Develop an applet that displays a simple message in the center of the screen.

Practical Exercises:

The learners are required to

1. Apply quadratic formulas in Java to calculate and display real solutions for $ax^2 + bx + c = 0$, handling negative discriminants.

2. Implement recursive and non-recursive functions in Java to compute and display the nth Fibonacci number, enhancing algorithmic understanding.

3. Identify prime numbers up to a given integer in Java by iterating through possible divisors, improving number theory application skills.

4. Construct a Java program to perform matrix multiplication, utilizing nested loops to handle matrix rows and columns effectively.

5. Evaluate a line of integers in Java, displaying each integer and calculating their sum, reinforcing data handling and aggregation techniques.

Suggested Readings:

1. Programming with java – a primer - E. Balagurusamy, 3rd Edition, TMH
2. The complete reference java 2 - Patrick Naughton & Hebert Schildt, 3rd, TMH
3. Programming with java – John R. Hubbard, 2nd Edition, TMH.

Note:

1. Only the latest editions of the above books are recommended.

WORDPRESS LAB
WD-L-303

Total Credits: 2

L   T   P
0   0   2

Internal: 30
External: 70

Course Objectives:
The course aims to provide comprehensive knowledge of WordPress, including installation, content creation, media management, customization, plug-ins, user management, site settings, reader interaction, and security maintenance, equipping students with practical skills to build and manage dynamic websites effectively.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Install and configure WordPress, navigating the dashboard and managing settings to tailor site behavior and functionality.

CO2: Create, format, and schedule posts and pages, enhancing content with multimedia elements and advanced text formatting.

CO3: Customize site appearance using themes and widgets, ensuring visual appeal and user-friendly navigation.

CO4: Manage plugins and user profiles, enhancing site functionality, maintaining security, and ensuring smooth operation and user interaction.

List of Programs

1. Introduction to WordPress: Install WordPress locally or on a web server and navigate the WordPress dashboard.
2. Creating posts and pages: Write and publish a post and a page, exploring the differences between the two.
3. Formatting text: Format text within a post using headers, lists, and other formatting options available in the WordPress editor.
4. Publishing and scheduling posts: Schedule a post to be published at a future date and time, and experiment with the publishing options.
5. Adding images, audio, and video: Insert images, audio files, and video into a post or page, and customize their display settings.
6. Managing content: Use the WordPress Media Library to manage and organize all your media files.
7. Customizing Appearance: Change the site's theme and customize it using the WordPress Customizer.
8. Using widgets: Add and configure widgets in the sidebar and footer areas of your WordPress site.
9. Working with plug-ins: Install and activate a plug-in, then configure its settings to add new functionality to your site.
10. Editing user profiles: Edit your user profile and manage other user accounts, assigning roles and capabilities.
11. Configuring settings: Explore and configure the settings under the WordPress Settings menu to customize your site's behavior.
12. Interacting with readers: Enable and manage comments on posts, and respond to reader comments.
13. Security and maintenance: Install a security plug-in, perform a site backup, and update WordPress, themes, and plug-ins to maintain site security and performance.

Practical Exercises:

The learners are required to
1. Install WordPress locally or on a web server, then navigate and customize the WordPress dashboard.
2. Write and publish a post and a page, exploring and documenting the differences between the two content types.
3. Format text within a post using headers, lists, and various formatting options available in the WordPress editor.
4. Schedule a post to be published at a future date and time, experimenting with different publishing options.
5. Insert images, audio files, and video into a post or page, then customize their display settings for optimal presentation.

Suggested Readings:
1. WordPress 5 Complete: Build beautiful and feature-rich websites from scratch, 7th Edition by Karol Król
2. WordPress Development Quick Start Guide by Paperback, Ratnayake Rakhitha Nimesh
3. Learn WordPress in Easy Way by Dr. Ritesh Kumar

Note:
1. Only the latest editions of the above books are recommended.

FOURTH SEMESTER

Subject Code	Subject Name	L-T-P	Credits	Marks Weightage		Course Type
				Internal	**External**	
WD-401	Introduction to Python	3-0-0	3	25	75	PCC
WD-402	Web Technologies using ASP(dot)NET	3-0-0	3	25	75	PCC
WD-L-401	Python Lab	0-0-3	3	30	70	SDP
BSC-401P	Project	0-0-2	2	30	70	SDP
OEC-401 to 403	Open Elective Course	3-0-0	3	25	75	OEC
WD-L-402	ASP(dot)NET Lab	0-0-3	3	30	70	SDP
PEC- WD-401 to 403	Program Elective Course	3-0-0	3	25	75	PEC
Total		12-0-8	20	190	510	

INTRODUCTION TO PYTHON
WD-401

Total Credits: 3

L T P

Theory: 75

3 0 0

Sessional: 25

Course Objectives:

This course provides an in-depth introduction to Python, covering its history, data types, operators, and control statements. It explores functions, data structures, object-oriented programming, file handling, and advanced topics such as exception handling and modular programming, equipping students with comprehensive Python programming skills.

Course Outcomes:

After the successful completion of the course, students will be able to:

CO1: Recall Python's history, data types, and control statements to understand foundational concepts.

CO2: Utilize functions, data structures, and comprehensions to solve programming problems effectively.

CO3: Examine and implement object-oriented principles, inheritance, and exception handling for robust code design.

CO4: Develop modular programs and handle files proficiently, demonstrating advanced Python programming capabilities.

Unit-I

Introduction to Python: Brief history of python, Data types - Built-in, Sequence, Sets, Strings, Literals, constants, keywords, variables, naming convention. Operators: types, Precedence & Associativity, Input, Output, file handling, Control Statements.

Unit-II

Functions and Data Structures in Python: Functions – basics of functions, functions as objects, recursive functions, List –methods to process lists, Shallow & Deep copy, Nested lists, lists as matrices, lists as stacks, Queues, Deques, Tuples - basic operations on tuples, nested tuples, Dictionaries – operations on dictionary, ordered dictionary, iteration on dictionary, conversion of lists & strings into dictionary, Sets & frozen sets, looping techniques on lists & dictionaries, Lamda, filter, reduce, map, list comprehension, iterators and generators.

Unit-III

Objects in Python: Class and instance attributes, inheritance, multiple inheritance, methods resolution order, magic methods and operator overloading, meta classes, abstract and inner classes, exception handling, modular programs and packages.

Unit-IV

File Handling: File handling Modes, Reading Files, Writing & Appending to Files, Handling File Exceptions, The with statement

Practical Exercises:

The learners are required to

1. Create and manipulate various Python data types to implement basic arithmetic operations and control statements.
2. Develop and use recursive functions to solve factorial and Fibonacci sequence problems.
3. Implement and compare shallow and deep copies of lists to understand their

differences in memory handling.
4. Design and implement classes with inheritance and operator overloading to model real-world objects and behaviours.
5. Write, read, and append to files, handling exceptions and using the 'with' statement for resource management.

Suggested Readings:
1. Introduction to Computing and Problem Solving with Python, Jeeva Jose, Khanna Publishing House
2. Taming Python by Programming, Jeeva Jose, Khanna Publishing House
3. Michael Urban and Joel Murach, Python Programming, Shroff/Murach, 2016

Note:
1. Only the latest editions of the above books are recommended

WEB TECHNOLOGIES USING ASP(dot)NET
WD-402

Total Credits: 3

L T P   Theory: 75

3 0 0   Sessional: 25

Course Objectives:

This course covers OOP concepts, standard and validation controls, rich controls, master pages, list and grid view controls, SQL data source, and ADO.NET data access. Students will gain skills in web development, user input validation, database connectivity, and data manipulation using ASP.NET and ADO.NET.

Course Outcomes:

After the successful completion of the course, students will be able to:

CO1: Explain OOP concepts, standard controls, and validation mechanisms to create interactive web forms and interfaces.

CO2: Design and implement master pages, list controls, and grid views for dynamic and structured web page layouts.

CO3: Examine and use SQL data source controls for database connections and command execution in web applications.

CO4: Develop data access components with ADO.NET to efficiently manage and manipulate data in web-based applications.

Unit-I

Basics: OOPs Concept, Classes and Objects, Inheritance, Polymorphism, Abstraction and Encapsulation Introduction to Standard Controls: Display information, accepting user input, submitting form data, displaying images, using the panel control, using the hyperlink control. Introduction to Validation Controls: Using the required field validator control, using the range validator control using the compare validator control, using the regular expression validator control, using the custom validator control, using the validation summary controls. Introduction to Rich Controls: Accepting file uploads, displaying a calendar, displaying advertisement, displaying different page views, displaying a wizard.

Unit-II

Designing Website with Master Pages: Creating master pages, Modifying master page content, Loading master page dynamically. List Controls: Dropdown list control, Radio button list controls, list box controls, bulleted list controls, custom list controls. Grid View Controls: Grid view control fundamentals, using field with the grid view control, working with grid view control events extending the grid view control.

Unit-III

SQL Data Source Control: Creating database connections, executing database commands, Using ASP.NET parameters with the SQL data source controls, programmatically executing SQL data source commands, Cashing database data with the SQL data Source controls.

Unit-IV

Building Data Access Components with ADO.NET: Data table Object, Connected data access, Disconnected data access, Using dbCommand Object , Connection classes, Executing a synchronous database commands, Using the DbDataReader Object, Using DbDataadapter Object, ListBox ADO.Net Data Binding, Building database objects with the .NET framework.

Practical Exercises:

The learners are required to

1. Implement OOP principles in a basic application to demonstrate inheritance, polymorphism, encapsulation, and abstraction.
2. Create a web form with standard controls for displaying information, accepting user input, and submitting form data.
3. Use validation controls to ensure user input meets specified criteria, including required field, range, and regular expression validators.
4. Design a website using master pages, incorporating dynamic content modification and various list controls.
5. Develop data access components with ADO.NET to perform connected and disconnected data operations, binding data to a ListBox.

Suggested Readings:

1. Professional ASP.NET 4.5 in C# and VB, Wrox Publication, Jason N. Gaylord, Christian Wenz, Pranav Rastogi,   Todd Miranda, Scott Hanselman, Scott Hunter
2. Pro ASP.NET 4.5 in C#, Apress Publication, Freeman, Adam, MacDonald, Matthew, Szpuszta, Mario

Note:

1. Only the latest editions of the above books are recommended

PYTHON LAB
WD-L-401

Total Credits: 3

L   T   P

Internal: 30

0   0   3

External: 70

Course Objectives:
This course aims to equip students with fundamental programming skills in Python by covering essential topics such as arithmetic operations, array manipulation, list handling, number theory, and algorithmic problem-solving, ensuring a solid foundation for further computational and software development studies.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Create Python programs to perform arithmetic operations, demonstrating fundamental computational skills and problem-solving abilities.

CO2: Analyze and implement algorithms to solve number theory problems, such as checking prime numbers and finding Fibonacci sequences.

CO3: Apply Python programming techniques to manipulate arrays and lists, including sorting, rotating, and finding elements.

CO4: Evaluate and synthesize solutions for real-world problems, like calculating interests and checking Armstrong numbers, using Python.

List of Programs
1. Write a program to add two numbers.
2. Write a program to find a maximum of two numbers.
3. Write a program to find the factorial of a number.
4. Write a program to find simple interest.
5. Write a program to find compound interest.
6. Write a program to check Armstrong Number.
7. Write a program to find the area of a circle.
8. Write a program to print all Prime numbers in an Interval.
9. Write a program to check whether a number is Prime or not.
10. Write a program for n-th Fibonacci number.
11. Write a program for how to check if a given number is a Fibonacci number.
12. Write a program for nth multiple of a number in Fibonacci Series.
13. Write a program to print ASCII Value of a character.
14. Write a program for Sum of squares of first n natural numbers.
15. Write a program for cube sum of first n natural numbers.
16. Write a program to find the sum of the array.
17. Write a program to find the largest element in an array.
18. Write a program for array rotation.
19. Write a program for Reversal algorithm for array rotation.
20. Write a program to Split the array and add the first part to the end.
21. Write a program for Find remainder of array multiplication divided by n.
22. Python Program to check if a given array is Monotonic.
23. Write a program to interchange first and last elements in a list.
24. Write a program to swap two elements in a list.
25. Write a program to find the length of the list.
26. Write a program to check if an element exists in a list.
27. Write a program of different ways to clear a list in Python.

28. Write a program for reversing a List.
29. Write a program to find the sum of elements in a list.
30. Write a program to multiply all numbers in the list.
31. Write a program to find the smallest number in a list.
32. Write a program to find the largest number in a list.

Practical Exercises:

The learners are required to

1. Develop a program to calculate simple and compound interest, applying mathematical formulas to real-world financial problems.
2. Implement a program to find the largest element in an array, evaluating various array manipulation techniques.
3. Design a program to check if a number is an Armstrong number, analysing the properties of numbers.
4. Create a program to rotate an array using the reversal algorithm, synthesizing knowledge of array manipulation.
5. Evaluate if a list is monotonic by writing a program that checks for increasing or decreasing order.

Suggested Readings:

1. Introduction to Computing and Problem Solving With Python, Jeeva Jose, Khanna Publishing House
2. Taming Python by Programming, Jeeva Jose, Khanna Publishing House
3. Michael Urban and Joel Murach, Python Programming, Shroff/Murach, 2016
4. Mark Lutz, Programming Python, O`Reilly, 4th Edition, 2010

Note:

1. Only the latest editions of the above books are recommended.

ASP(dot)NET LAB
WD-L-402

Total Credits: 3

L   T   P   Internal: 30

0   0   3   External: 70

Course Objectives:

This course aims to provide a comprehensive understanding of C# and the .NET framework, including form design, database connectivity, state management, web server controls, master pages, validation controls, grid view control, ADO.NET, stored procedures, dynamic link libraries, and Crystal Reports for robust web application development.

Course Outcomes:

After the successful completion of the course, students will be able to:

CO1: Understand the fundamentals of C# and .NET framework, including their features and architecture.

CO2: Apply form design and database connectivity techniques to create robust and user-friendly applications.

CO3: Analyze and implement state management techniques and web server controls in ASP.NET for efficient web application development.

CO4: Evaluate the use of ADO.NET, stored procedures, and Crystal Reports for advanced data handling and reporting.

List of Programs

1. Introduction to C#, .net framework, feature, etc.
2. To Study Design of form and Database Connectivity.
3. To state management technique.
4. To study the web server controls of asp.net.
5. Master Page and content page.
6. To study the Validation controls in asp.net.
7. To study the grid view control in asp.net.
8. To study the ADO.net and Stored procedure in asp.net.
9. To study Dynamic link Library.
10. To Study Crystal reports

Practical Exercises:

The learners are required to

1. Develop a C# application using .NET framework features, demonstrating understanding of its architecture and capabilities.
2. Create a form with database connectivity, applying ADO.NET techniques to retrieve and display data.
3. Implement state management in an ASP.NET application, analysing the effectiveness of different techniques.
4. Design a web application using master and content pages, evaluating their role in consistent layout and navigation.
5. Construct a dynamic report using Crystal Reports, synthesizing data from a database to generate detailed summaries.

Suggested Readings:

1. ASP.NET Programmer's Reference by Paperback, Caison Charles
2. ASP.NET Core in Action, Second Edition by Andrew Lock
3. ASP.NET 4.5, Covers C# and VB Codes, Black Book by Kogent Learning Solutions

Inc.

Note:

1. Only the latest editions of the above books are recommended.

ENTREPRENEURSHIP
OEC-401

Total Credits: 3

L T P   Theory: 75

3 0 0   Sessional: 25

Course Objectives:
This course provides a comprehensive understanding of entrepreneurship, covering essential traits of entrepreneurs, industry classification, start up support systems, tax systems, relevant industrial acts, and project report preparation, including ISO 9000 series quality systems, equipping students to effectively launch and manage new ventures.

Course Outcomes:
After the successful completion of the course, students will be able to:

    CO1: Explain the essential characteristics of successful entrepreneurs and the classification of industries by size.

    CO2: Identify and utilize support systems for start-ups, including roles of various state and national organizations.

    CO3: Evaluate the impact of tax systems and industrial acts on business operations and compliance requirements.

    CO4: Prepare a comprehensive project report, incorporating ISO 9000 series quality systems and proper format guidelines.

Unit-I
Entrepreneurship and entrepreneur: Entrepreneurship concept and process, Entrepreneur, Essential Characteristics of a good Entrepreneur, Types of entrepreneurs, Industrial Policy, Classification of industries- Micro, small scale, medium scale, large scale, Product identification/ selection, Site selection, Plant layout, Pre-market survey.

Unit-II
Entrepreneurship Support System and Start-ups: Introduction to start-up's, Role of District Industries Centre in setting up industry, Function of NSIC, SISI, NISIET, NRDC, SSIC, SIDO, NMTC, KVIC, RSMML, Role of state finance corporation, state electricity corporations, pollution control board, BIS, I.S.O. etc.

Unit-III
Introduction to Tax System, and Acts: Idea of income tax, Goods and Services Tax and custom duty, Introduction to Industrial Acts, factory Act, Workmen's Compensation Act 1923, Apprentices Act 1961, Environmental Protection Act 1986.

Unit-IV
Project Report Preparation: Procedure of preparing a project report, Format of project report, Preparation of project report, Introduction to ISO: 9000 Series of Quality System.

Practical Exercises:
The learners are required to

1. Analyze the characteristics of successful entrepreneurs and identify the types of entrepreneurs in real-world case studies.
2. Investigate and report on the functions of local entrepreneurship support organizations such as NSIC and District Industries Centre.
3. Evaluate the impact of current tax systems on small and medium-sized enterprises through a detailed case study.
4. Review and critique sample project reports to identify best practices and common

pitfalls in report preparation.
5. Develop a comprehensive project report for a hypothetical business, including ISO 9000 series quality system guidelines.

Suggested Readings:
1. Khanka S.S., "Entrepreneurship Development" S. Chand
2. Desai, A N. "Entrepreneur & Environment" Ashish, New Delhi.
3. Drucker, Peter. "Innovation and Entrepreneurship" Heinemann, London.
4. Jain Rajiv. "Planning a Small-Scale Industry: A Guide to Entrepreneurs" S.S. Books, Delhi.
5. Kumar, S A. "Entrepreneurship in Small Industry" Discovery, New Delhi.

Note:
1. Only the latest editions of the above books are recommended

TRENDS IN TECHNOLOGY
OEC-402

Total Credits: 3

L T P

3 0 0

Theory: 75

Sessional: 25

Course Objectives:
The course aims to equip students with comprehensive knowledge of e-commerce, e-banking, e-governance, e-agriculture, and e-learning, focusing on their architectures, functionalities, benefits, and security protocols, while providing practical insights through case studies and real-world applications.

Course Outcomes:
After the successful completion of the course, students will be able to:

- CO1: Analyze the architecture, types, advantages, and payment systems of e-commerce, including smart card technologies.
- CO2: Utilize e-banking transactions and security protocols such as SSL and digital signatures for safe online financial operations.
- CO3: Assess e-governance models and strategies, addressing challenges and implementing effective solutions in various contexts.
- CO4: Develop and utilize e-learning platforms and tools like LMS, video conferencing, and virtual campuses for enhanced education delivery.

Unit-I
E-Commerce: Introduction: E-commerce as Business need-commerce, Types, Advantages, Disadvantages, e-Commerce Architecture; Internet Payment Systems, Characteristics, 4C Payment Methods, SET Protocol for Credit Card Payment, E-Cash, E-Check, Overview of Smart Card.

Unit-II
E-mail & Internet: Introduction, E-mail Account & Its Functions, Search Engine, Surfing Webpages, Basics of Social Networking Site. E-Banking Transactions: Inter Banking, Intra Banking, Electronic Payments, (Payment – Gateway Example), Securities in E-banking (SSL, Digital Signatures – Examples), Services Provided: ATM, Smart Card ECS (Electronic Clearing System), e.g., Telephone, Electricity Bills.

Unit-III
E – Governance & E – Agriculture: E –Governance Models: (G2B, G2C, C2G, G2G), Challenges to E – Governance, Strategies and tactics for implementation of E – Governance, Types of Agriculture information (Soil, Water, Seeds, Market rate) & Technique dissemination, Future trade marketing, Corp Management, Query redresses System, (Information Kiosk, IVR, etc.), Case Study.

Unit-IV
E-learning: Models WBT, CBT, Virtual Campus, LMS & LCMS, Video Conferencing, Chatting Bulleting, Building Online Community, Asynchronous / Synchronous Learning, Case Study.

Practical Exercises:
The learners are required to
1. Develop an E-commerce Website: Create a functional e-commerce site with payment systems like E-Cash and E-Check, incorporating smart card transactions.
2. Implement and test an e-banking transaction system, ensuring security with SSL and

digital signatures.
3. Build an e-governance portal with G2C and G2B models, incorporating strategies for overcoming common challenges.
4. Develop a digital platform to disseminate agricultural information such as soil data, water usage, and market rates.
5. Construct an e-learning environment using LMS, virtual campus tools, and synchronous/asynchronous learning methods.

Suggested Readings:
1. E–Commerce: C.V.S. Murty.
2. Fire Wall and Internet Security: William Cheswick, Stevens, Aviel, Rubin.
3. The Essential Guide to Knowledge management: Amrit Tiwana.
4. The GIS Book: George B. Karte.
5. Management Information System: Laudon & Laudon

Note:
1. Only the latest editions of the above books are recommended

WASTE MANAGEMENT
OEC-403

Total Credits: 3

L T P

Theory: 75

3 0 0

Sessional: 25

Course Objectives:
The course aims to provide comprehensive knowledge of solid and hazardous waste management, including regulatory frameworks, waste characterization, storage, transportation, processing technologies, and disposal methods, emphasizing sustainable practices and environmental protection

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Analyze types and sources of waste, and interpret Indian legislation on waste management and handling.

CO2: Characterize and reduce waste: Assess physical, chemical, and biological properties of wastes, and implement strategies for waste reduction and recycling.

CO3: Optimize waste storage and transport: Design effective systems for waste segregation, storage, collection, and transportation, ensuring safety and efficiency.

CO4: Implement waste processing and disposal: Apply technologies for waste processing and disposal, including composting, incineration, and landfill management, focusing on sustainability.

Unit-I
Sources, Classification and Regulatory Framework: Types and Sources of solid and hazardous wastes, Need for solid and hazardous waste management, Elements of integrated waste management and roles of stakeholders, Salient features of Indian legislations on management and handling of municipal solid wastes, hazardous wastes, biomedical wastes, lead acid batteries, electronic wastes, plastics and fly ash, Financing waste management.

Unit-II
Waste Characterization and Source Reduction: Waste generation rates and variation, Composition, physical, chemical and biological properties of solid wastes, Hazardous Characteristics, TCLP tests, waste sampling and characterization plan, Source reduction of wastes, Waste exchange, Extended producer responsibility, Recycling and reuse Practical Composition of MSW, Determination of Physical and Chemical Properties of MSW.

Unit-III
Storage, Collection and Transport of Wastes: Handling and segregation of wastes at source, storage and collection of municipal solid wastes, Analysis of Collection systems, need for transfer and transport, transfer stations Optimizing waste allocation, compatibility, storage, labelling and handling of hazardous wastes, hazardous waste manifests and transport. Waste Processing Technologies: Objectives of waste processing, material separation and processing technologies, biological &chemical conversion technologies, methods and controls of Composting , thermal conversion technologies, energy recovery, incineration, solidification & stabilization of hazardous wastes- treatment of biomedical wastes

Unit-IV
Waste Disposal: Waste disposal options, Disposal in landfills, Landfill Classification, types and methods, site selection, design and operation of sanitary landfills, secure landfills and landfill bioreactors, leachate and landfill gas management, landfill closure and environmental monitoring, Rehabilitation of open dumps, landfill remediation.

Practical Exercises:

The learners are required to

1. Research and present on Indian waste management laws, highlighting key provisions and their implications.
2. Conduct TCLP tests and analyze waste samples to determine their physical, chemical, and hazardous characteristics.
3. Develop a comprehensive waste management plan for a community, considering storage, collection, and transportation logistics.
4. Simulate waste processing techniques such as composting and incineration, evaluating their efficiency and environmental impact.
5. Design a landfill site, considering classification, operation methods, environmental monitoring, and rehabilitation strategies.

Suggested Readings:

1. George Techobanoglous et al, "Integrated Solid Waste Management", McGraw - Hill, 2014.
2. Manual on Municipal Solid waste Management, CPHEEO, Ministry of Urban Development, Govt. Of. India, New Delhi, 2000.

Note:

1. Only the latest editions of the above books are recommended

TECHNOLOGY TRENDS IN IT
PEC-WD-401

Total Credits: 3

L T P
                   Theory: 75

3 0 0
                 Sessional: 25

Course Objectives:
The course aims to provide a comprehensive understanding of Internet of Things (IoT), Cloud Computing, and Big Data Systems, including their characteristics, architectures, applications, challenges, and emerging trends in the context of modern technology environments.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Evaluate IoT characteristics, designs, and communication models, distinguishing between IoT and M2M systems, and addressing associated challenges.

CO2: Develop IoT architectures, incorporating web standards, domain-specific applications, and business intelligence for diverse industry applications.

CO3: Apply SaaS, PaaS, and IaaS models, leveraging virtualization and cloud storage for efficient resource management and service delivery.

CO4: Implement big data architectures, including Map-Reduce, Hadoop, and NoSQL databases, optimizing querying, managing, and analyzing large datasets.

Unit-I
Internet of Things (IoT) – Introduction to IoT, Characteristics of IoT, Physical design of IoT, Logical design of IoT, Functional blocks of IoT, Communication models &APIs ,IoT& M2M Machine to Machine, Difference between IoT and M2M, Software define Network, Challenges in IoT(Design ,Development, Security), Wireless medium access issues, MAC protocol survey, Survey routing protocols, Sensor deployment & Node discovery, Data aggregation & dissemination.

Unit-II
Internet of Things (IoT): Web of Things vs Internet of things, two pillars of web, Architecture and standardization of IoT, Unified multitier-WoT architecture, WoT portals and Business intelligence, Domain specific applications of IoT, Home automation, Industry applications, Surveillance applications, Other IoT applications Clustering, Synchronization, Software agents.

Unit-III
Cloud Computing: SaaS, PaaS, IaaS, Public and Private Cloud; Virtualization, Virtual Server, Cloud Storage, Database Storage, Resource Management, Service Level Agreement.

Unit-IV
Big Data Systems: Big Data Characteristics, Types of Big Data, Big Data Architecture, Introduction to Map- Reduce and Hadoop; Distributed File System, HDFS. NOSQL: NOSQL and Query Optimization; Different NOSQL Products, Querying and Managing NOSQL; Indexing and Ordering Data Sets; NOSQL in Cloud.

Practical Exercises:
The learners are required to
1. Design a physical and logical IoT device, considering communication models, APIs, and security challenges.
2. Develop a unified multitier WoT architecture, integrating business intelligence for

domain-specific IoT applications.
3. Simulate SaaS, PaaS, and IaaS models, configuring virtual servers and databases for resource management exercises.
4. Analyze big data characteristics, implement Map-Reduce and Hadoop for distributed file systems, and optimize NoSQL queries.
5. Create IoT applications for home automation, industry, and surveillance, utilizing clustering, synchronization, and software agents for functionality enhancement.

Suggested Readings:
1. Computer Today, A. Ravichandran, Khanna Publishing House
2. Internet of Things, Jeeva Jose, Khanna Publishing House
3. Big Data and Hadoop, V.K. Jain, Khanna Publishing House
4. Data Sciences and Analytics, V.K. Jain, Khanna Publishing House

Note:
1. Only the latest editions of the above books are recommended

INTRODUCTION TO AI
PEC-WD-402

Total Credits: 3

L   T   P
                                    Theory: 75

3   0   0
                                   Sessional: 25

Course Objectives:
The course aims to provide a comprehensive understanding of Artificial Intelligence (AI) fundamentals, including problem-solving techniques, knowledge representation, natural language processing, learning methodologies, expert systems, and knowledge acquisition, preparing students for practical applications in AI-related fields.

Course Outcomes:
After the successful completion of the course, students will be able to:
- CO1: Evaluate the importance and techniques of AI, assess criteria for success, and define problem spaces for heuristic search.
- CO2: Implement various approaches to knowledge representation, utilizing predicate logic to represent facts and relationships.
- CO3: Develop syntactic, semantic, and pragmatic processing techniques, applying learning methodologies for problem-solving and example induction.
- CO4: Construct expert systems using domain-specific knowledge, explore knowledge acquisition methods, and analyze early Machine Learning concepts and applications.

Unit-I
Overview of AI: Introduction to AI, Importance of AI, AI and its related field, AI techniques, Criteria for success. Problems, problem space and search: Defining the problem as a state space search, Production system and its characteristics, Issues in the design of the search problem Heuristic search techniques: Generate and test, hill climbing, best first search technique, problem reduction, constraint satisfaction.

Unit-II
Knowledge Representation: Definition and importance of knowledge, Knowledge representation, various approaches used in knowledge representation, Issues in knowledge representation. Using Predicate Logic: Represent ting Simple Facts in logic, representing instances and is-a relationship, Computable function and predicate.

Unit-III
Natural language processing: Introduction syntactic processing, Semantic processing, Discourse and pragmatic processing. Learning: Introduction learning, Rote learning, learning by taking advice, learning in problem solving, learning from example-induction, Explanation based learning.

Unit-IV
Expert System: Introduction, representing using domain specific knowledge, Expert system shells. Knowledge acquisition: General concepts in knowledge acquisition, early work in Machine Learning, examples of Inductive Learners, computer vision, Robotics, overview of LISP- AI language.

Practical Exercises:
The learners are required to
1. Evaluate AI techniques' applicability and success criteria, identifying problems and applying heuristic search methods.

2. Implement knowledge representation approaches, using predicate logic to represent facts and relationships accurately.
3. Simulate syntactic, semantic, and pragmatic processing tasks, applying learning methodologies for problem-solving and example induction.
4. Construct an expert system using domain-specific knowledge, exploring knowledge acquisition methods and early Machine Learning concepts.
5. Develop programs in LISP, focusing on AI applications such as computer vision and robotics, applying acquired knowledge practically.

Suggested Readings:
1. Artificial Intelligence, Munish Chandra Trivedi, Khanna Publishing House
2. Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig.

Note:
1. Only the latest editions of the above books are recommended

COMPUTER NETWORK SECURITY
PEC-WD-403

Total Credits: 3

L   T   P

Theory: 75

3   0   0

Sessional: 25

Course Objectives:
This course aims to provide a comprehensive understanding of network concepts, including types and benefits, architectures, transmission media, topologies, and devices; delve into Ethernet networking and switching technologies; explore internet layer protocols, IP addressing, and router configuration; and introduce transport and application layer functions with a focus on network security.

Course Outcomes:
After the successful completion of the course, students will be able to:

CO1: Analyze different network classifications and architectures, and evaluate their benefits and transmission media types.

CO2: Apply knowledge of Ethernet networking and switching technologies to design efficient and secure network topologies.

CO3: Evaluate internet layer protocols, IP addressing schemes, and router configurations for optimized network performance.

CO4: Understand the functions of transport and application layers, and implement basic network security measures in various scenarios.

Unit-I
Network Concept, Benefits of Network, Network classification (PAN, LAN, MAN, WAN), Peer to Peer, Client Server architecture, Transmission media: Guided & Unguided, Network Topologies. Networking terms: DNS, URL, client server architecture, TCP/IP, FTP, HTTP, HTTPS, SMTP, Telnet OSI and TCP/IP Models: Layers and their basic functions and Protocols, Comparison of OSI and TCP/IP. Networking Devices: Hubs, Switches, Routers, Bridges, Repeaters, Gateways and Modems, ADSL.

Unit-II
Ethernet Networking: Half and Full-Duplex Ethernet, Ethernet at the Data Link Layer, Ethernet at the Physical Layer. Switching Technologies: layer-2 switching, address learning in layer-2 switches, network loop problems in layer-2 switched networks, Spanning-Tree Protocol, LAN switch types and working with layer-2 switches, Wireless LAN

Unit-III
Internet layer Protocol: Internet Protocol, ICMP, ARP, RARP. IP Addressing: Different classes of IP addresses, Sub-netting for an internet work, Classless Addressing. Comparative study of IPv4 & IPv6. Introduction to Router Configuration. Introduction to Virtual LAN.

Unit-IV
Transport Layer: Functions of transport layer, Difference between working of TCP and UDP. Application Layer: Domain Name System (DNS), Remote logging, Telnet, FTP, HTTP, HTTPS. Introduction to Network Security.

Practical Exercises:
The learners are required to

1. Design a small LAN network using different topologies and assess their performance and benefits.
2. Configure a layer-2 switch to implement VLANs and analyze network segmentation

and traffic management.
3. Implement and compare IPv4 and IPv6 addressing schemes, including sub-netting and classless addressing.
4. Set up a client-server architecture using DNS, FTP, and HTTP protocols, and evaluate their functionality.
5. Develop basic network security protocols and test their effectiveness in protecting data and ensuring secure communications.

Suggested Readings:
1. Information & Computer Security, Sarika Gupta, Khanna Publishing House
2. An Integrated Approach to Computer Networks, Bhavneet Sidhu, Khanna Publishing House

Note:
1. Only the latest editions of the above books are recommended