Parvatibai Chowgule College of Arts and Science
(Autonomous)
Accredited by NAAC with Grade 'A+'
Best Affiliated College-Goa University Silver Jubilee Year Award

DEPARTMENT OF COMPUTER SCIENCE

SYLLABUS FOR UNDERGRADUATE PROGRAMME

THREE YEARS BACHALORS DEGREE IN COMPUTER SCIENCE / FOUR YEARS HONOURS IN COMPUTER SCIENCE

PLO’s for BSc Program in Computer Science :

PLO-1	Understand and apply the fundamental knowledge in core areas of Computer Science.
PLO-2	Demonstrate proficiency in different programming paradigms needed for Problem Solving.
PLO-3	Develop verified, tested, efficient and secure computerized solutions to real world problems through the application of principles in Computer Science and current technologies.
PLO-4	Ability to engage and contribute to providing socially responsible design, environment friendly solutions, adapting professional ethics in the broader context of technological change.

DEPARTMENT OF COMPUTER SCIENCE

PROPOSED NEP STRUCTURE
WEF FROM AY 2023-2024
B.Sc IN COMPUTER SCIENCE (WITH HONOURS)
(4-year course structure)

SEM	Major subject/ discipline (3T+1P)	Minor Stream (Disc./Voc) (3T+1P)	(MDC for Others) (2T + 1P)	VAC (2 credits)	SEC (2T+1P)	INTERNSHIP
I	UG-COM-101 Introduction to Programming	UG-COM-101 Introduction to Programming	UG-COM-MDC	UG-COM-VAC	UG-COM-SEC	
			UG-COM-MDC			
II	UG-COM-102 Database Management	UG-COM-102 Database Management	UG-COM-MDC	UG-COM-VAC	UG-COM-SEC	
			UG-COM-MDC			
III	UG-COM-201: Data Structures	UG-COM-207: Foundations of Data Science with Python	UG-COM-MDC		UG-COM-SEC	
	UG-COM-202: Object Oriented Programming					
IV	UG-COM-203: Computer Architecture and Operating System	UG-COM-VOC-1: Mobile Application Development				
	UG-COM-204: Mathematical Foundations for Computer Science					
	UG-COM-205: Web Development with php					
	UG-COM-206:					

	Software Engineering					
V	UG-COM-301: Full Stack Development	UG-COM-VOC -2: Digital Marketing				Internship (4 credits)
	UG-COM-302: IOT					
	UG-COM-303: Computer Networks					
VI	UG-COM-304: Data Science	UG-COM-VOC -3:Software Testing(voc)				
	UG-COM-305: Cloud Computing					
	UG-COM-306: AI					
	UG-COM-PRJ: Project					
VII	UG-COM-401: Design and Analysis of Algorithms	UG-COM-405: Image Processing				
	UG-COM-402: Software Architecture, Design Patterns and Frameworks					
	UG-COM-403: Advanced DBMS					
	UG-COM-404: Machine Learning					
VIII	UG-COM-406: Compiler Design	UG-COM-410: Network Security				
	UG-COM-407: Deep Learning					
	UG-COM-408: NLP					
	UG-COM-409: Educational Technology **Parallel Computing**					

Note : MDC, SEC and VAC will be offered from the pool of approved syllabus.

SEMESTER I

DISCIPLINE SPECIFIC CORES

Course Title: Introduction to Programming
Course Code: UG-COM-101
Marks: 75
Credits: 3
Duration: 45 Hrs
Course Prerequisite: Nil

Course Objectives :

- To understand the concept of basic computer algorithm and flowchart and use the algorithm for various problem solving.
- To implement algorithms using a high level programming language.
- To understand basic principles of structured programming –example C.

Course Learning Outcome: Upon completion of the course students will be able to:

CLO-1: Understand problem solving strategies.
CLO-2: Draw a flowchart and write an algorithm for a given problem.
CLO-3: Recognize and incorporate programming elements such as loops, decision making, functions, arrays, strings.
CLO-4: Recognize and incorporate programming elements such as structures, pointers and files into applications that solve real world problems.

SYLLABUS

UNIT I **[10 HRS]**

Introduction to Computer Problem Solving : Algorithms, Flowchart, The Problem-Solving Aspect, General problem-solving strategies, Top-Down Design, Implementation of Algorithms, Efficiency of Algorithms, Recursive algorithms.

Basic Algorithms : Exchanging the values, Summation of a set of numbers, factorial computation, generation of the Fibonacci series, reversing the digits of an integer, base conversion.

Factoring Methods : Finding divisors of an integer, finding the Greatest Common Divisor of two integers, generating prime numbers, computing prime factors of an integer.

UNIT II **[20 HRS]**

C Language : History, Structure of a C program, Keywords, Identifiers, variables, constants, data types, Arithmetic Operators & Expressions, Logical operators and Relational Operators, Precedence and Associativity rules.

Conditions and Iterations : Conditions and Actions, Condition statement, Simple control statement (if, if-else,switch), Iterative control statements (for, while, do-while).
Functions : What is a function, Advantages of functions, Standard library functions; User define functions – declaration, definition, function call, parameter passing, return keyword. Scope of variables, Storage classes, Recursion.

Arrays : One- and Two-dimensional arrays: Array declaration, initialization, accessing the values, passing arrays to functions.

Pointers : Pointer declaration, initialization, Pointer arithmetic, Pointer to Pointer, Arrays and Pointers, Functions and Pointers – passing pointers to functions, function returning pointer, dynamic memory allocation.

UNIT III **[15 HRS]**

Strings : Declaration and initialization, standard library string functions, strings and pointers, array of strings.

Structure and Union : Creating structures, accessing structure members, array of structures, passing structure to functions, nested structure, pointers and structures, union, difference between structures and unions.

File Handling : FILE variable, file access modes, operations on files, random access to files, command line arguments.

Pre-processing : Format of Preprocessor directive, File Inclusion directive, Macro substitution, conditional compilation.

REFERENCES:

MANDATORY:

1. Dromey, R. G. (1982). How to Solve it by Computer. Prentice-Hall, Inc..
2. Kanetkar,Y. (2012). Let us C, BPB Publications,
3. Forouzan, B. A., & Gilberg, R. F. (2000). Computer Science: A structured programming approach using C. Brooks/Cole Publishing Company..

SUPPLEMENTARY:

1. Horowith,E.,Sahni ,S.,Sanguthevar,R. (2008). Fundamentals of Computer algorithm, Orient Longman.
2. Gottfried,B. (2010). Programming with C, Tata McGraw Hill.

WED BASED:

1. GNU GCC (GNU Compiler Collection) @http://gcc.gnu.org, with source codes.

2. Bjarne Stroustrup's C++11 FAQ @http://www.stroustrup.com/C++11FAQ.html.
3. https://www.tutorialspoint.com/cprogramming
4. https://www.javatpoint.com/c-programming-language-tutorial
5. https://www.w3schools.in/c-tutorial/
6. https://www.guru99.com/c-programming-tutorial.html
7. https://www.geeksforgeeks.org/c-programming-language/
8. E Book - https://www.edutechlearners.com/download/books/Let%20Us%20C%20by%20Yashavant%20Kanetkar%20PDF.pdf
9. E Book - http://www2.cs.uregina.ca/~hilder/cs833/Other%20Reference%20Materials/The%20C%20Programming%20Language.pdf
10. E Book - http://www.kciti.edu/wp-content/uploads/2017/07/cprogramming_tutorial.pdf

Practical: Introduction to Programming

Credit: 1
Marks: 25
Duration: 30 Hrs

Programs using C language that covers the following concepts:

1. Conditions **(1P)**
   - if..else
   - nested if

2. Iterative Control Statements **(2P)**
   - for
   - while
   - do…while

3. Functions. **(2P)**
   - Standard Library functions
   - Call by Value
   - Call by reference
   - Recursive functions

4. Arrays. **(2P)**
   - One Dimensional Arrays
   - Two Dimensional Arrays

5. Pointers. **(2P)**
   - Arrays and Pointers
   - Function returning pointers
   - Dynamic memory allocation

6. Strings. **(2P)**
   - Standard Library string functions
   - Strings and Pointers
   - Array of Strings

7. Structure and Union **(2P)**
   - Array of structures
   - Passing Structure to functions
   - Nested structure
   - Structure and Pointer
   - Union

8. File Handling. **(2P)**
   - Text file
   - Binary file
   - Random Access to a file
   - Command Line arguments

****

SEMESTER II

Course Title: Database Management Systems - I
Course Code: UG-COM-102
Marks: 75
Credits: 3
Duration: 45 Hrs
Course Prerequisite: Nil

Course Objectives:

- To provide basic knowledge of a database management system.
- To understand the importance of Entity Relationship diagram.
- To formulate queries in Relational Algebra and SQL for Database manipulation.
- Familiarity with any RDBMS during practical sessions.

Course learning outcomes:

Upon completion of the course students will be able to:

CLO-1: Explain the fundamental concepts of database systems, including data models, database languages, and database design
CLO-2: Design and model a database using Entity-Relationship diagrams.
CLO-3: Formulate queries in Relational Algebra, and SQL to create and manipulate the databases.
CLO-4: Normalize databases to eliminate data redundancy and ensure data integrity.

SYLLABUS

UNIT I: Overview of DBMS, Design and ER model. [15 HRS]

Overview of database management : Data, information, database, database management system; Managing data; File systems versus a DBMS, advantages of a DBMS; Data abstraction, instances and schemas, data models; Data manipulation language, data definition language; Architecture of a DBMS; Users of a DBMS, database administrator.

Database design and the ER model: Design phases – conceptual design, logical design, physical design; ER model – entities, attributes, and relationships, mapping cardinalities, keys; ER diagrams – strong entities, weak entities, generalization, specialization, aggregation; Converting ER diagram to relational schemes.

Unit II : Functional dependency and normalization [10 HRS]

Atomic domain, nested relation; Key, super key, primary key, candidate key; Functional dependency, axioms, closure of a set of attributes, closure of a set of functional dependencies; Purpose of normalization; 1NF; 2NF;3NF; BCNF.

UNIT III : Relational model and SQL [20 HRS]

Relational model: Relation, properties of relational model; Entities, integrity constraints, referential integrity constraints; Relational algebra – select, project, cross product, set operations, rename operation; Other relational operations – natural join, outer join.

SQL: Basic structure of SQL query – Create, select, where, from, rename operation; Set operations; Aggregate functions; Group by, having clauses; DDL Nested queries; Views; Insert, delete, update.

REFERENCES

MANDATORY:

1. Silberschatz, A. (2006). Database system concepts.—6th ed.

SUPPLEMENTARY:

1. Ramakrishnan, R., & Gehrke, J. (2000). Database management systems. McGraw Hill.
2. Elmasri, R., & Navathe, S. (2017). Fundamentals of database systems. Pearson.

WEB BASED:

1. https://www.db-book.com/db6/slide-dir/ (Sixth edition Powerpoint , PDF of A Silberschatz, H F Korth, S Sudarshan, Database system concepts,)
2. https://www.db-book.com/db7/
3.https://www.tutorialspoint.com › dbms
4.https://www.w3schools.in › dbms
5.https://www.studytonight.com › dbms
6.https://www.oracletutorial.com

Practical: Database Management Systems - I
Credit :1
Marks :25
Duration: 30 Hrs

1. ER diagram	(2P)
2.ER diagram with specialization/generalization and aggregation.	(1P)
3. Converting ERD into Schema.	(2P)
4 SQL	(2P)
5 Nested Queries	(2P)
6.Normalization	(2P)
7. Mini Project	(4P)

****

SEMESTER III

DISCIPLINE SPECIFIC CORES

Course Title: Data Structures
Course Code: UG-COM-201
Marks: 75
Credits: 3
Duration:45 Hrs

Course Prerequisite: Nil

Course Objectives:

- To impart the basic concepts of data structures and algorithms.
- To understand concepts about searching and sorting techniques.
- To understand basic concepts about stacks, queues, lists, trees and graphs.
- To understanding about writing algorithms and step by step approach in solving problems with the help of fundamental data structures.

Course outcomes:

Upon completion of the course students will be able to:

CLO-1 :.Understanding core data structures.
CLO-2 : Implementation and manipulation of data structures.
CLO-3 : Analyze and implement different sorting and searching algorithms.
CLO-4 : Perform fundamental operations on different types of tree structures, including insertion, deletion, and various traversals.

SYLLABUS

UNIT I: [15 HRS]

Introduction to data structures: Concept, Data type, Data object, ADT, Need of Data Structure, Types of Data Structure.

Algorithm Analysis: Algorithm – definition, characteristics, Space complexity, time complexity, Asymptotic notation (Big O).

LinkedList:

Introduction to List, Implementation of List – static & dynamic representation, Types of Linked List,
Operations on List, Applications of Linked List, polynomial manipulation, Generalized linked list –
concept & representation.

UNIT II **[15 HRS]**

Stacks:
Introduction, Representation-static & dynamic, Operations

Stack Applications:

Application - infix to postfix & prefix, postfix evaluation, Simulating recursion using stack.

Queues:
Introduction, Representation -static & dynamic, Operations, Circular queue, priority queue (with implementation), Concept of doubly ended queue.

Trees:
Concept & Terminologies, Binary tree, binary search tree, Representation – static & dynamic, Operations on BST – create, Insert, delete, traversals (preorder, inorder, postorder), counting leaf, non-leaf & total nodes, non recursive inorder traversal, Expression Tree. Introduction to AVL Trees

UNIT III **[15 HRS]**
M-Way Search Trees: Introduction, B Tree, B+ Tree.

Searching and Sorting:

Use of various data structures for searching and sorting, selection sort, merge sort, quick sort, heap sort and hashing.

Graph:
Concept & terminologies, Graph Representation – Adjacency matrix, adjacency list,Traversals – BFS & DFS, Application of BFS, DFS – Shortest path, Backtracking.

REFERENCES:

MANDATORY:

1.Horowitz,E., Sahni,S. (2008). Fundamentals of Data Structures in C, University Press.

SUPPLEMENTARY:

1.LangsamYedidyah, Augenstein J. Moshe, Tenenbaum M.A aron ,(2018),Data Structure using C, Pearson Education.

2. Richard.G, Behrouz.F, Data Structures: A Pseudocode Approach with C, Cengage Learning.

WEB BASED:

1. https://www.tutorialspoint.com/data_structures_algorithms
2. https://www.w3schools.in/data-structures-tutorial
3.https://www.studytonight.com/data-structures/
4.https://www.programiz.com/dsa
5. https://www.geeksforgeeks.org/data-structures/
6. https://www.javatpoint.com/data-structure-tutorial
7. E Book -
https://www.scribd.com/doc/261233741/Data-Structures-Through-C-Yashavant-Kanetkar

Practical: Data Structures
Credit: 1
Marks: 25
Duration: 30 Hrs

Programs using C language that covers the following concepts:

1. Stack: Static/Dynamic stack implementation. **(1P)**
2. Stack: infix to postfix. **(1P)**
3. Stack: Evaluation of Postfix expression. **(1P)**
4. Queues: Static and Dynamic **(1P)**
5. Queue :Implementation **(1P)**
6. Implementation of Circular queue **(1P)**
7. List: Singly LinkedList **(1P)**
8. List: Doubly LinkedList **(1P)**
9. List: Circular Linked List **(1P)**
10. Linked List: Polynomial addition **(1P)**
11. Trees: Binary Search Tree: create, add, delete, display nodes. **(1P)**
12. Trees: BST traversal. **(1P)**
13. Graph: Representation of Graphs, **(1P)**
14. Graph Traversals. Graph: DFS,BFS. **(2P)**

****

Course Title: Object Oriented Programming
Course Code: UG-COM-202
Marks: 75
Credits: 3
Duration: 45 Hrs

Course Prerequisites: Nil

Course Objectives:

- To teach the basic concepts and techniques which form the object oriented programming paradigm.
- To introduce object oriented programming (OOP) using Java.

Course Outcomes:
Upon completion of the course students will be able to:

CLO-1: Apply fundamental object-oriented concepts in problem solving.
CLO-2: Identify classes, their members and relationships between them.
CLO-3: Implement the object oriented model in any object oriented language.
CLO-4: Develop applications to solve real world problems.

SYLLABUS:

UNIT I: Introduction **[15 HRS]**

Principles of OOP

Programming paradigms. Basic concepts in OOP. OOP: major principles - encapsulation, abstraction, inheritance, polymorphism. Benefits of OOP. Applications of OOP.

Introduction to Java
Basics of Java programming, Data types, Variables, Operators, Control structures including selection, Looping, Java methods, Overloading, java. Math class, Arrays in java.

UNIT II Principles of OOP **[15 HRS]**

Basics of objects and classes in java, Constructors, Finalizer, Visibility modifiers, Methods and objects, object serialization, Inbuilt classes like String, Character, String Buffer, File, this reference.

Inheritance and Polymorphism:
Inheritance in java, Super and subclass, Overriding, java.lang. Object class, Polymorphism, Dynamic binding, Generic programming, Casting objects, Instance of operator, Abstract class, Interface in java, Package in java, java.util package.

UNIT III: Exceptions, Multi-threading and GUI programming: [15 HRS]

Event and GUI programming
Design patterns – what and why? It's classification. Introduce the Observer design pattern. Event handling in java, Event types, Mouse and key events, GUI Basics, Panels, Frames, Layout.

Managers:
Flow Layout, Border Layout, Grid Layout, GUI components like Buttons, Check Boxes, Radio Buttons, Labels, Text Fields, Text Areas, Combo Boxes, Lists, Scroll Bars, Sliders, Windows, Menus, Dialog Box, Applet and its life cycle.

Multithreading in java
Multithreading in java, Thread life cycle and methods, Runnable interface, Thread synchronization.

Exception handling
Exception handling – what and why? Try and catch blocks. Multiple catch blocks. Nested try, finally block, throw keyword, throw keyword. Custom Exception. Introduction to the Collections Framework.

REFERENCES:

MANDATORY:

1.Deitel, P., & Deitel, H. (2011). Java How to program. Prentice Hall Press.

SUPPLEMENTARY:

1. Naughton.P, Schildt.H, (2000), Java 2 – The Complete Reference TMH publications
2. Patrick,N. (1997). The Java Handbook –TMH publications
3. Mughal, K. A., & Rasmussen, R. W. (2003). A programmer's guide to Java certification: a comprehensive primer. Addison-Wesley Professional.
4. Flanagan, D. (2004). Java examples in a nutshell.
5. Arnold, G., H.(2005)"The Java Programming Language" Addison-Wesley Professional.

WEB BASED:

1.www.javapoint.com
2.www.tutorialspoint.com
3.www.gurugg.com
docs.logout.org &gt; Programmation &gt;Java&gt;Programming with Java_A primer

Practicals: Object Oriented Programming
Credit: 1
Marks: 25
Duration: 30 Hrs

Programs using Java language that covers the following concepts:

1) Classes and instances	**(1P)**
2) Working with the java.Math class	**(1P)**
3) Inheritance	**(1P)**
4) Composition v/s inheritance	**(1P)**
5) Polymorphism, abstract classes and interfaces	**(1P)**
6) Algorithm and Data Structures	**(1P)**
7) Utilizing the java.util package	**(1P)**
8) Event handling and GUI	**(1P)**
9) Applets	**(1P)**
10) I/O programming	**(1P)**
11) Exception handling	**(1P)**
12) Multi-threading	**(2P)**
13) Collections framework	**(2P)**

****

SEMESTER III

MINOR COURSE

Course Title: Foundations of Data Science with Python
Course Code: UG-COM-207
Credits: 3
Marks: 75

Course Objectives:

- To understand the concept of data science and its role in extracting insights from data
- To understand the basic Python syntax, data types, control structures, and functions
- Utilize Python libraries such as NumPy and Pandas for data manipulation and analysis tasks.

Course Outcomes:

At the end of the course students will be able to:

CLO1: Explain key principles related to data manipulation, analysis, visualization, and interpretation using Python programming language.
CLO2: Apply Python programming fundamentals to perform data manipulation, analysis, and visualization tasks efficiently.
CLO3: Employ data manipulation libraries such as NumPy and Pandas to clean, transform, and explore datasets effectively.
CLO4: Describe the basic principles and types of machine learning algorithms, including supervised and unsupervised learning approaches.

Module I: Introduction to Data Science and Python Basics [15 hrs]

Introduction to Data Science: Understanding the role and importance of data science in various fields, Introduction to key concepts: data, information, knowledge, and insights, Overview of data science tools and techniques.

Introduction to Python Programming: Introduction to Python programming language, Installing Anaconda distribution for Python, Basic Python syntax: variables, data types, operators, and expressions, Control structures: if statements, loops, and functions, Hands-on exercises and practice problems using Jupyter notebooks.

Working with Data in Python:Introduction to data structures in Python: lists, tuples, dictionaries, Accessing and manipulating data in Python, Introduction to libraries for data manipulation: NumPy and Pandas, Reading and writing data from/to different file formats (e.g., CSV, Excel), Practical exercises and projects to manipulate and analyze data

Module II: Data Visualization and Analysis [15 hrs]

Data Visualization with Matplotlib and Seaborn: Introduction to data visualization principles, Overview of Matplotlib and Seaborn libraries for data visualization, Creating basic plots: line plots, scatter plots, bar charts, histograms, Customizing plots: labels, colors, markers, and legends, Exploratory data analysis through visualization, Hands-on exercises and projects to visualize data using Jupyter notebooks

Statistical Analysis with Python: Introduction to statistical analysis concepts, Descriptive statistics:mean, median, mode, variance, standard deviation, Inferential statistics: hypothesis testing, t-tests, chi-square tests, Introduction to statistical libraries in Python: SciPy and StatsModels, performing statistical analysis on datasets using Jupyter notebooks, Practical exercises and projects to apply statistical techniques.

Module III: Introduction to Machine Learning [15 hrs]

Fundamentals of Machine Learning: Introduction to machine learning concepts and terminology, Types of machine learning: supervised learning, unsupervised learning, and reinforcement learning, Overview of popular machine learning algorithms: linear regression, logistic regression, decision trees, k-nearest neighbors, Introduction to scikit-learn library for machine learning in Python,

Time-Series Analysis: Overview of time-series analysis, Components of time-series, time-series forecasting model.

Practicals :Foundations of Data Science with Python.

Credit: 01

Marks: 25

Duration: 30 Hrs

1. Python Basics: [3P]

a) Write Python code to calculate the area of a circle given its radius.
b) Create a Python function that takes two numbers as input and returns their sum.
c) Write a Python program to check if a given number is prime.

2. Working with Data in Python: [3P]

a) Load a CSV file containing student grades into a Pandas DataFrame. Display the first few rows of the DataFrame.
b) Calculate the mean, median, and standard deviation of a numeric column in the DataFrame.
c) Filter the DataFrame to show only students with grades above a certain threshold.

3. Data Visualization with Matplotlib and Seaborn: [3P]

a) Create a line plot showing the trend of a stock price over a period of time.
b) Generate a scatter plot to visualize the relationship between two variables in a dataset.
c) Create a histogram to visualize the distribution of ages in a population dataset.

4. Statistical Analysis with Python: [3P]

a) Perform a t-test to compare the mean heights of two groups of individuals.
b) Calculate the correlation coefficient between two numeric variables in a dataset.
c) Conduct a chi-square test to analyze the association between two categorical variables.

5. Fundamentals of Machine Learning: [3P]

a) Implement linear regression to predict housing prices based on features like square footage and number of bedrooms.
b) Train a logistic regression model to classify email messages as spam or not spam based on text features.
c) Build a decision tree classifier to predict whether a passenger survived the Titanic disaster based on passenger attributes.

REFERENCE BOOKS:

MANDATORY:

1. Nandi, G., Sharma, R. K. (2020). Data Science Fundamentals and Practical Approaches: Understand Why Data Science Is the Next (English Edition). India: Bpb Publications.
2. McKinney, W. (2017). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython. United States: O'Reilly Media.
3. Let Us Python. (2019). India: BPB Publications.

SUPPLEMENTARY:

1. Dromey R.G., How to solve it by computer, Prentice Hall of India, 2nd Edition, 2004.
2. Mark Lutz, Learning Python, O'Reilly Media, Third Edition, 2008
3. Wes McKinney, Python for Data Analysis, O'Reilly Media, 2012
4. Jure.L,,Anand. R,Jeffrey.U(2014). Mining of Massive Datasets v2.1(2nd ed.).Cambridge University Press.
5. Kevin P. Murphy,(2012).Machine Learning: A Probabilistic Perspective .MIT Press.
6. Provost, F., & Fawcett, T. (2013). Data Science for Business: What you need to know about data mining and data-analytic thinking. " O'Reilly Media, Inc.".

WEB REFERENCES:

1. https://www.python.org/doc/
2. https://www.w3schools.com/python/
3. https://matplotlib.org/stable/index.html
4. https://seaborn.pydata.org/
5. https://scikit-learn.org/stable/
6. https://www.tensorflow.org/api_docs

****

SEMESTER IV

DISCIPLINE SPECIFIC CORES

Course Title: Computer Architecture and Operating Systems
Course Code: UG-COM-203
Marks: 75
Credits: 3
Duration: 45 Hrs

Course Prerequisites: Nil

Course Objectives:

- Study of Processor architecture, memory and I/O subsystems including basic Assembly Language Programming.
- To Understand the Basic objectives, functions and types of Operating System
- To study various aspects of operating system like Process Management, Memory
- Management, Storage management etc.
- To understand different algorithms used for CPU scheduling, Memory allocation.

Course Outcomes:
Upon completion of the course students will be able to:

CLO1: Explain the detailed function of a typical Computer system and its components.
CLO2: Identify the functions of operating system
CLO3: Analyze the process, its states and process scheduling algorithms
CLO4 :Explain deadlocks and synchronization, memory management, and disk scheduling. .
CLO5: Implement Assembly Language Program and Shell Script for a given task.
CLO6: Implement algorithm of CPU Scheduling, Memory Scheduling.

SYLLABUS:

UNIT I: Computer System and Components: [20 HRS]
Processor:Function and structure of a computer, Interconnection of components, Performance of a computer. Computer Architecture – Princeton (Von Neumann) and Harvard architecture.Architecture of 8086 processor - Registers, ALU and Control unit, Data path in a CPU.Instruction cycle, Organization of a control unit – Block Diagram of Hardwired andMicroprogrammed control unit. Instruction set architectures – CISC and RISC architectures.
Memory Subsystem: Characteristics of memory system, the memory hierarchy, Semiconductormemories, Types of ROM & RAM, Cache memory unit.
Input/Output Subsystem: General block diagram of External device and I/O module, Programmed I/O, Interrupt driven I/O, DMA, I/O channels and I/O processors.

UNIT II: Operating system overview and Process Management [15 HRS]

Overview: Operating system Objectives and Functions, Evolution of operating system, major achievements.

Process Management: Process Definition, Process Control Block, Process States, Operations on Process; Interprocess communication, Process Scheduling, Scheduling Criteria, Scheduling Algorithms,

Process Coordination: Process Synchronization, Principles, Mutual Exclusion, The Critical-Section Problem, Peterson's Solution.

Deadlocks- system models, Deadlock characterization, Deadlock Handling Methods, Prevention, Avoidance, Detection, Recovery from Deadlock.

UNIT III: Memory Management and Storage Management
[10HRS]

Memory Management: Introduction, Swapping, Contiguous Memory Allocation, Paging, PageTable, Segmentation, Virtual Memory: Introduction, Demand Paging, Page Replacement, Allocation of Frames, Thrashing

Secondary Storage Structure : Overview, disk structure, Disk attachment, Disk scheduling.

REFERENCES:

MANDATORY:

1. Stallings, W. (2003). Computer organization and architecture: designing for performance. Pearson Education India.
2. Silberchatz, A., Galvin, ,& Gagne. (2008). Operating System Concepts (8th ed.). Wiley publication.

SUPPLEMENTARY:

1. Stallings, W. (2001). Operating systems: Internals and design principles.(6th ed.). Upper Saddle River, N.J: Prentice Hall.
2. David A. Patterson and John L. Hennessy (2021), Computer Organization and Design: The Hardware/Software Interface, Elsevier
3. Douglas V. Hall, (2012), Microprocessors and its Interfacing, McGraw Hill Education (India) Private Limited
4. Sumitabha Das , UNIX Concepts and Applications, Tata McGraw-Hill

WEB BASED:

1. http://williamstallings.com/ComputerOrganization/
2. https://www.tutorialspoint.com/computer_fundamentals/index.htm
3. https://www.geeksforgeeks.org/operating-systems/

Video Links and Animations

1. 4.https://www.youtube.com/watch?v=WP3uDglbPiI
1. 5.http://williamstallings.com/OS-Animation/Animations.html
2. 6.Linux Tutorials for Practical
3. 7.https://www.tutorialspoint.com/unix/index.htm

Practicals: Computer Architecture and Operating Systems
Credit: 1
Marks: 25
Duration: 30 Hrs

1. Study of Motherboard, Peripherals and the Computer System:O.S. Installation (DualBoot): BIOS; Manage disk partitions: understand MBR-style partitions, (primary, extended, logical); list/create/delete partitions; Manage logical volumes: create/remove physical volumes, create/delete logical volumes, Boot loader.Installation of drivers; updating softwarepackages DOS Commands, Tools for Computer Management (Disk Management, Disk Cleanup, Defragmentation, Performance Monitor, System Restore etc). [1]

Assembly language programs for 8086 using MASM / compatible assembler or Simulator, either in Windows or Linux.

1. Study of addressing modes. [1]
1. Programs for data transfer and arithmetic operations and logical operations [1]
2. Study of Basic commands of Linux [1]
3. Shell Programming in Unix/Linux, arithmetic operations, loops. [1]
4. .Shell Programming – advanced [1]
5. Filters and Pipes in LINUX. [2]
6. Implementation of Inbuilt Linux/UNIX commands like cp, rename etc. [1]
7. Implementation of CPU scheduling policies. [1]
8. 10.Implementation of Memory allocation Techniques. [1]

****

Course Title: Mathematical Foundations for Computer Science
Course Code: UG-COM-204
Marks: 75
Credits: 3
Duration: 45 Hrs

Course Prerequisites: Basic Knowledge of Programming.

Course Objectives:

- To introduce students to the fundamental concepts of systems of linear equations and matrices.
- Enable students to be able to apply mathematical principles to solve real-world problems.

Course Outcomes:

Upon completion of the course students will be able to:

CLO-1: Represent and solve linear systems using matrices.
CLO-2: Competence in dealing with linear transformations, null space, range, and isomorphisms.
CLO-3: Define and work with vector spaces, subspaces, and basis, emphasizing their role in computer graphics and linear algebra applications.
CLO-4: Compute eigenvalues and eigenvectors of matrices, particularly in the context of data analysis, machine learning, and optimization.
CLO-5: Use interpolation methods to estimate values between data points.
CLO-6: Application of matrices and Vectors in solving real-world problems.

SYLLABUS:

UNIT I: Systems of Linear Equations and Matrices, Linear Combinations and Linear Independence [10 HRS]

Matrices and its representations, Types of Matrices, Matrix Operations, The Inverse of a Square Matrix, Matrix Equations, Elementary Row Operations, Applications of Matrices in Computer Science, Adjoint of matrix and Rank.

Systems of Linear Equations, Applications of Systems of Linear Equations, Linear Combinations and Linear Independence. Linear Dependence and Consistency of Systems.

UNIT II: Vector Spaces, Linear Transformations, Eigenvalues and Eigenvectors [15 HRS]

Definition of a Vector Space, Subspaces, Basis and Dimension, Coordinates and Change of Basis Linear Transformations, The Null Space and Range, Isomorphisms, Matrix Representation of Linear Transformations, Similarity. Eigenvalues and Eigenvectors, Diagonalization.

UNIT III: Interpolation, Numerical Integration, algebraic and transcendental equation [20 HRS]

Introduction; Various methods of interpolation; Various methods of curve fitting; Newton's method of forward interpolation formula; Newton's method of backward interpolation formula. Lagrange's formula.
General quadrature formula; Trapezoidal rule; Simpson's one-third rule; Simpson's three-eighth rule; Weddle's rule.

Graphical method; Bisection method; Method of false position; Secant method; Newton-Raphson method. Linear equations; Characteristics roots and vectors.

REFERENCES:

MANDATORY:

1. Iyengar,S.N.,(2010) Matrices, Anmol Publications.
2. Defranza, J., & Gagliardi, D. (2015). Introduction to Linear Algebra with applications. Waveland Press.
3. Goel,B., & Mittal,S. (1998) Numerical Analysis, Pragati Prakashan,

SUPPLEMENTARY:

1. Chatterjee,P. (1996) Numerical Analysis, RajhansPrakashanMandir.
2. Krishnamurthy,V. (1976) Introduction to Linear Algebra, Affiliated East-West Press.

WEB BASED:

1.https://www.cse.iitb.ac.in
2. https://www.wolframalpha.com/
2. https://www.khanacademy.org/
3. https://web.stanford.edu/class/cs357/

Practicals: Mathematical Foundation of Computer Science
Credit: 1
Marks: 25
Duration: 30 Hrs

Programs using Java language that covers the following concepts:

1. Implementing Matrices **(1P)**
2. Performing Basic Matrix Operations. **(1P)**
3. Performing Elementary Row operations **(1P)**
4. Finding the rank of a matrix. **(1P)**
5. Finding solutions of a system of equations using Matrices. **(1P)**
6. Matrix Representation of Linear Transformations. **(1P)**
7. Finding the Eigenvalues and Eigenvectors. **(1P)**
8. Implementing Newton's forward formula to estimate a value. **(1P)**
9. Implementing Newton's backward formula to estimate a value. **(1P)**
10. Estimate a value using Lagrange's formula. **(1P)**
11. Apply Simpson's three-eighth rule to find the value of integration. **(2P)**
12. Apply Newton-Raphson method OR secant method to estimate the root of a equation**(2P)**

****

Course Title: Web Development with PHP-MYSQL
Course Code: UG-COM-205
Marks: 75
Credits: 3
Duration: 45 Hrs

Course Prerequisites: Basic Knowledge of HTML and CSS, Programing and Database Management.

Course Objectives:

- To Provide students with a foundational understanding of web development and the role of PHP and MySQL in building dynamic websites.
- Enable students to create dynamic web pages using PHP, including displaying data from a database, processing user input, handling sessions and generating dynamic content.
- Cover user authentication and authorization mechanisms to secure web applications, including user registration and login systems.

Course Outcomes:
Upon completion of the course students will be able to:

CLO-1: Demonstrate a fundamental understanding of PHP in Object Oriented paradigm, including its syntax and basic principles.
CLO-2: Conduct CRUD (Create, Read, Update, Delete) operations with PHP and MySQL, involving the retrieval, insertion, updating, and deletion of data.
CLO-3: Apply hashing and security measures for user authentication.
CLO-4: Implement security best practices in PHP web development, including input validation and sanitation, to protect against common threats like Cross-Site Scripting (XSS) and SQL injection.

SYLLABUS:

Unit I : Introduction to PHP [15 HRS]

Introduction to PHP- PHP syntax and basics, Setting up a local development environment. PHP Variables, Data types (integers, strings, arrays), Operators (arithmetic, comparison, logical). Control Structures-Conditional statements (if, else, switch), Loops (for, while, foreach). Functions and Includes - Defining and using functions, Including external files.

Unit II : Database Connectivity with PHP [15 HRS]

MySQL Database Basics - Creating databases and tables, Data types and constraints. Connecting to MySQL - PHP database connections, Error handling for database connections. CRUD Operations with PHP and MySQL - SELECT, INSERT, UPDATE, DELETE operations, Prepared statements for security.

Unit III : Building Dynamic Web Pages using best practices in security [15 HRS]

Form Handling with PHP - Creating HTML forms, Processing form data with PHP. Displaying Database Data - Retrieving and displaying data from the database, Pagination and sorting. User Authentication - Building user registration and login systems, Password hashing and security. Security and Best Practices - Input validation and sanitation, Cross-site scripting (XSS) and SQL injection prevention.

REFERENCES:

MANDATORY:

1. Larry Ullman. (2018). "PHP and MySQL for Dynamic Web Sites." Peachpit Press.
2. Janet Valade. (2016). "PHP 7 for Dummies." For Dummies.
3. Robin Nixon. (2018). "Learning PHP, MySQL & JavaScript." O'Reilly Media.

SUPPLEMENTARY:

1. Luke Welling and Laura Thomson. (2016). "PHP and MySQL Web Development." Addison-Wesley.
2. Chris Snyder, Michael Southwell, and Tom Myer. (2005). "PHP Security." O'Reilly Media.
3. Bryan Sullivan and Vincent Liu. (2011). "Web Application Security, A Beginner's Guide." McGraw-Hill Education.

WEB BASED:

1. https://www.php.net/
2. https://www.w3schools.com/php/
3. https://phptherightway.com/
4. https://www.tutorialrepublic.com/php-tutorial/php-mysql-crud-application.php
5. https://www.geeksforgeeks.org/how-to-design-username-and-password-using-html-and-php/

Practicals : Web Development with php
Credit : 1
Marks : 25
Duration: 30 Hrs

List of suggested PRACTICALS :

1. Set up a local development environment for PHP web development and configure it. **(1P)**
2. Define and utilize functions in PHP, and incorporate external files into PHP scripts using includes. **(2P)**
3. Establish connections between PHP and MySQL databases and handle potential errors effectively during database interactions. **(2P)**
4. Create HTML forms for user input and develop PHP scripts to handle form submissions securely. **(2P)**
5. Conduct CRUD (Create, Read, Update, Delete) operations with PHP and MySQL, involving the retrieval, insertion, updating, and deletion of data. **(2P)**
6. Implement prepared statements to enhance security and prevent SQL injection vulnerabilities in PHP and MySQL interactions. **(2P)**
7. Design and build user registration and login systems, applying password hashing and security measures for user authentication. **(2P)**

****

Course Title: Software Engineering
Course Code: UG-COM-206
Marks: 75
Credits: 3
Duration: 45 Hrs

Course Prerequisites: Nil

Course Objectives:

- To understand the various software development methodologies and estimation of software projects.
- To analyze and design software projects.
- To study the various phases of a S/W Development Project.

Course Outcomes:
Upon completion of the course students will be able to:

CO1: Understand the various Software Development Methodologies.
CO2: Apply Estimation techniques to live projects.
CO3: Analyze Software Projects.
CO4: Design Software Projects.

SYLLABUS:

UNIT I: **[15 HRS]**
SOFTWARE PROCESS:
Characteristics of software process.
Software Development Processes and Methodologies: waterfall, prototyping, iterative, spiral, unified process. Benefits of iterative and incremental approach with emphasis on Unified process, CASE Tools, Agile methodologies.

PROJECT MANAGEMENT:
Planning a Software Project – Cost estimation, Project Scheduling, Software configuration management plans, Quality Assurance plans, Project Monitoring plans and Risk Management. Techniques such as Interviewing, Requirement Workshop, brainstorming, prototyping. Characteristics of SRS.

UNIT II: **[15 HRS]**
OOAD and UML:
OOAD: Definition; object oriented analysis; object oriented design and modeling; Assigning responsibilities.

UML: Main UML diagrams – Class diagram, sequence diagram, activity diagram, use case diagram.
Use case model – use case diagram, use case descriptions, use case realization using sequence and activity diagrams. Supplementary requirements. Advanced use case model features.

Requirements: Functional and non-functional
System Design : Class diagram, sequence diagram, activity diagram, state chart diagram, deployment diagram. Brief introduction to other UML diagrams.

SOFTWARE ARCHITECTURE PATTERNS:
Major Architectural Styles (patterns) like Layered Architecture, Pipe and Filter, Shared (Central)Data Store, Event Driven, Model-View-Controller (MVC), "Distributed & Emerging" Service Oriented Architecture (SOA) and Elementary GRASP Patterns.

UNIT III [15 HRS]

HUMAN COMPUTER INTERACTION:
HCI Definition; User categories, Interface Design-Internal & External Interface design, user interface design, Interface design guidelines.

CODING:
Coding styles, standards, peer reviews, checklist.

TESTING:
Testing Fundamental, Functional Testing, Structural Testing, Testing Object-Oriented Programs, Testing Process and Metrics.

DOCUMENTATION and MAINTENANCE:
Need for Software Documentation. Types of documentation, Need for Maintenance; Types of Maintenance.

REENGINEERING:
Business Process Reengineering, Software Reengineering, Reverse Engineering, Restructuring, Forward Engineering, The Economics of Reengineering.

REFERENCES:

MANDATORY:

1. Pressman, R. S. (2005). Software engineering: a practitioner's approach. Palgrave Macmillan.
2. Larman C.,(2015) Applying UML and patterns. Addison Wesley.
3. Bourque, P., & Fairley, R. E. (2014). Guide to the software engineering body of knowledge (SWEBOK (R)): Version 3.0. IEEE Computer Society Press.

SUPPLEMENTARY:

1.Jalote, P. (2012). An integrated approach to software engineering. Springer Science & Business Media.
2. Sommerville I.,(2015) Software Engineering .Adison Wesley.
3. Fowler, M. (2003). UML Distilled: A Brief Guide to the Standard Modelling Object Language. Object Technology Series, 3rd edition, Addison-Wesley.

WEB BASED:

1.https://www.tutorialspoint.com/software_engineering
2.https://www.w3schools.in/sdlc-tutorial
3.https://www.geeksforgeeks.org/software-engineering
4.https://www.javatpoint.com/software-engineering-tutorial

Practicals : Software Engineering
Credit : 1
Marks : 25
Duration: 30 Hrs

List of suggested PRACTICALS :

For a given project/case study

1) Requirements Gathering Techniques	[2P]
2) Gantt Chart	[2P]
3) USE Case diagram and Use Case descriptions for the Use Cases	[3P]
4) Class Diagram	[2P]
5) Sequence Diagram	[2P]
6) Activity Diagram	[2P]
7) State Chart Diagram	[2P]

****

SEMESTER I/II/III

SKILL ENHANCEMENT COURSES

Course Title: Programming with R
Course Code: UG-COM-SEC1
Marks: 50
Credits: 2
Duration:30 Hrs.
Prerequisite Courses : Nil

Course Objectives:

- To enable students to handle data in the R software thereby helping them to understand meaningful statistical analysis performed on the data.
- To enable students to extract data, and perform basic statistical operations.
- Data analysis such as – data cleaning, data visualization, data summarization and regression.

Course Learning Outcomes:

At the end of the course students will be able to:

CLO-1: Understand the R environment and do basic programming in R

CLO-2: Analyse, extract and manipulate data by using functions in R.

CLO-3: Compute various measures like central tendency, measures of dispersion by using graphical techniques.

CLO-4: Compute the correlation coefficient for bivariate data, perform simple and multiple linear regression on data set and analyse time series.

SYLLABUS

Unit 1: Basics of R-language **[15 Hrs]**

Overview of the R language: Installing R and R Studio: Using R studio, Scripts, Text editors for R, Graphical User Interfaces(GUIs) for R, Creating and storing R workspaces, installing packages and libraries, Mathematical operations.

Data Types in R- Numeric, Integer, Character, Logical, Complex and missing data.

Data Structures in R

- Vectors-Creation, Arithmetic operations of Vectors, Vector Sub setting, Sorting and Sequencing functions.
- Matrix and Arrays-Creation Arithmetic Operations of Matrix, Sub setting, Use of drop function.

- Factors-Converting a vector into factor, assigning levels and labels, ordered Factor.
- List- Creating a List, accessing List elements, manipulating List elements, merging Lists, converting Lists to Vectors.
- Data Frames-Creation of Data Frame, adding new columns, rows and removing columns, accessing column using the $ sign, importing a data set (important file formats such as csv, txt and spreadsheet), aggregate function and subsetting of dataframes, tapply function, manipulation using dplyr package (select, filter, arrange, mutate and group by function, pipe operator).

Programming Fundamentals: Relational Operators, Relational Operators and Vectors, Logical Operators, Logical Operators and Vectors, & vs &&, | vs ||, Conditional Statements, While Loop, For Loop, Looping Over List, Loops for Vectors, Loops for Matrices, Loops for Data Frames, Loops for Lists, writing a Function in R, Nested Functions, Function Scoping, Recursion, Mathematical Functions in R, Calculus in R, Input and Output Operations.

Reading and writing data in R (file formats such as csv, txt, and xlsx).

Unit 2: Basic Statistics and Regression [15 Hrs]

Summarizing and exploring data: Descriptive statistics (mean, median, mode, variance, skewness, five-point summary), dealing with missing data in R, Data cleaning (dplyr package, tidyr package and pipe operator), Exploratory Data Analysis: data visualization using inbuilt functions and ggplot2 package (pie chart, bar chart, line chart, histogram, box plot, scatter plot, normal QQ plot).

Regression analysis using R: Regression vs Correlation, Simple and multiple regression, Ordinary least square, Assumptions of Classical normal linear regression model, corrplot package, car package, lmtest package, scatter plot(using plot function and ggplot2 package) to understand the relationship between variables, lm, abline, predict, resid function, interpreting 'summary table' of the regression model, normality of residuals (qqnorm and qqPlot functions), multicollinearity (correlation matrix, corrplot and vif function), autocorrelation (acf plot and Durbin Watson test), heteroscedasticity (graphically, bptest, ncvTest).

Time series data, components of a time series data, additive and multiplicative time series model, ts function, diff function, plot of atime series data, time series data with linear trend; regression analysis using 'lm' function.

REFERENCES:

MANDATORY:

1. K. G. Srinivasa, G. M. Siddesh et al, Statistical Programming in R, Oxford University Press.
2. Gardener, M. (2018), Beginning R: The Statistical Programming Language, Wiley & Sons.

SUPPLEMENTARY:

1. Sekhar, S.R.M., et al. (2017), Programming with R, Cengage Learning India.
2. Wickham, H., et al. (2017), R for Data Science: Import, Tidy, Transform, Visualize, and Model Data, O’Reilly’.
3. Field, A., Miles, J and Field (2012), Z. Discovering Statistics using R (Indian Reprint 2022), SAGE
4. Sandip Rakshit, R Programming for Beginners, McGraw Hill Education.
5. Tilman M. Davies,The Book for R, No Starch Press.
6. Emmanuel Paradis, R Programming for Beginners.

WEB BASED:

1. https://intellipaat.com/blog/tutorial/r-programming/
2. https://www.geeksforgeeks.org/r-statistics/?ref=lbp
3. https://www.tutorialspoint.com/r/index.htm
4. https://www.javatpoint.com/r-tutorial
5. https://www.w3schools.com/r/
6. https://www.programiz.com/r

Practicals: Programming with R

Credit:1

Marks:25

Duration:30 Hrs

1. R-Programming Basics: Installing R Studio, Using R console to perform basic arithmetic operations, display strings and workspace variables. [1P]
2. Data Structures: Vectors, Lists, Matrices, Data Frames, Factors and Arrays [3P]
3. Control Flow and Iterations. [1P]
4. Functions in R. [1P]
5. Implement Problems based on measures of central tendency. [1P]
6. Implement Problems based on measures of dispersion. [1P]
7. Implement Problems based on moments, skewness and kurtosis. [1P]
8. Loading, data cleaning and data visualization. [2P]
9. Perform simple and multiple linear regression. [2P]
10. Perform time series analysis [2P]

Course Title: Programming in Python
Course Code: UG-COM-SEC2
Marks: 50
Credits: 2
Duration:30 Hrs.
Course prerequisite: Introduction to Programming(UG-COM-101)

Course Objectives:

- To apply various data types and control structures.
- To apply python data structures - list, tuple and dictionary.
- To structure a python program as a set of functions.
- To do input/output with files in Python.
- To provide skills of data analysis using Python programming language.

Course Learning Outcomes:

At the end of the course students should be able to :

CLO1: Understand the basics (Data types, Operators etc.)

CLO2: Write programs using conditional statements, loops

CLO3: Apply required List,Tuple and dictionary function

CLO4: Write Python program specific to the domain of the given problem

SYLLABUS

UNIT I **[15 HRS]**

Motivation, programming paradigms, What Python can do, Python's technical strength, Python interpreter, Program execution, Execution model variations, How to run programs.

Basic Syntax :
Variable and Data Types, Operator, Conditional Statements - if, if- else, Nested if-else. Looping – For, While, Nested loops. Control Statements – Break, Continue, Pass.

Strings :
Accessing Strings, Basic Operations, String slices, Function and Methods.

Tuples and Lists :
Introduction, Accessing list, Operations, Working with lists, Function and Methods. Introduction Accessing tuples, Operations, Working, Functions and Methods.

UNIT II **[15 HRS]**

Dictionaries:
Introduction, Accessing values in dictionaries, Working with dictionaries, Properties.

Functions:
Defining a function, Calling a function, Types of functions, Function Arguments, Anonymous functions, Global and local variables.

Modules:
Importing module. Math module. Random Module.

Exception Handling:
Exception. Exception Handling - Except clause, Try , except,finally clause. User Defined Exceptions.

Input-Output :
Printing on screen, Reading data from keyboard, Opening and closing file, Reading and writing files, Functions. Printing on screen, Reading data from keyboard, Opening and closing file, Reading and writing files.

REFERENCES:

MANDATORY:

1. Mark Lutz,Learning Python, O'Reilly Media, Fifth Edition.

SUPPLEMENTARY:

1. Alex Martelli, (2006)Python – A Nutshell, O'Reilly Media, Second Edition.
2. Wes McKinney, (2012)Python for Data Analysis, O'Reilly Media.

WEB BASED:

1. https://www.w3schools.com
2. https://www.tutorialspoint.com
3. https://www.javatpoint.com
4. https://www.geeksforgeeks.org
5. https://www.guru99.com

Practicals: Programming in Python

Credit: 01

Marks: 25

Duration:30 Hrs

List of Experiments using Python Language:

1. Write a python program to compute a given formula. [1P]

2. Write a python program to implement an if else statement. [1P]

3. Write a python program to implement nested if else statements. [1P]

4. Implement for and while loop in python. [2P]

5. Write a string manipulation in python. [2P]

6. Write a program to implement lists in python. [2P]

7. Write a python program to implement tuple. [1P]

8. Write a python program to implement a dictionary. [2P]

9. Write python program to implement function [1P]

10. Write a python program to implement Input-Output File Operations in python. [1P]

11. Using a Python package to manage the environmental impact of computation. [1P]

****

Course Title: UI/UX Design
Course Code: UG-COM-SEC3
Marks: 50
Credits: 2
Duration:30 Hrs.

Prerequisite: Nil

Course Objectives:

- To study the different aspects of UI/UX Design.
- To study computer interface design concepts.

Course Outcomes:

Upon completion of the course student will be able to:

CLO-1: Understand the principles of UI/UX design interaction with a computer System.
CLO-2: Understand the concept of a graphical user interface, and its design characteristics.
CLO-3: Apply fundamental design principles including typography, color theory, screen design and layouts.
CLO-4: To create interactive and intuitive user interfaces.

SYLLABUS:

UNIT I: Introduction to UI and UX Design: [15 HRS]

Explain the distinctions between UI and UX design. Understand the significance of UI/UX in digital product development.

Human-Computer Interaction: Importance of user Interface, Human characteristics, Human consideration, Human interaction speeds, Understanding business functions. User centered design, Persona.

Rapid Prototyping: Storyboarding. Paper Prototyping and Mockup, Video Prototyping, Creating and Comparing Alternatives.

Graphical User Interface Design: HCI guidelines, Windows: Navigation schemes selection of window; Selection of devices based and
screen-based controls, Components, Text and messages, Icons, Multimedia, Colors., controls, help &amp; error messages design.

UNIT II: Heuristic Evaluation and Visualization **[15 HRS]**

Web user interface design – Jessy James Garette five layers of user experience. Heuristic Evaluation: Heuristic Evaluation — Why and How? Visualization, Amount of information, Focus and emphasis, Presentation information simply and meaningfully, Statistical graphics.

REFERENCES:

Mandatory:

1. Cooper, A., Reimann, R., & Dubberly, H. (2003). About face 2.0: The essentials of interaction design. John Wiley & Sons, Inc..
2. Alan.D, Janet.F, Gregory D. and Russell,B. (2012) Human-Computer Interaction, Prentice Hall.
3. D.T,(2018) Web Technologies, Black Book, Dream Tech

Supplementary:

1. Shneiderman, B., & Plaisant, C. (2010). Designing the user interface: strategies for effective human- computer interaction. Pearson Education India.
2. Donald.A.N. (2010) The Design of Everyday Things Basic Books.

WEB BASED:

1. http://hcibib.org/
2. https://www.tutorialspoint.com/human_computer_interface/index.htm.
   https://www.academia.edu/4955516/Wiley_The_Essential_Guide_to_User_Interface_De sign_3rd_Edition_Apr_2007?auto=download.
3. https://www.slideshare.net/busaco/hci-2015-110-humancomputer-interaction-overview?qid=1c116f30-ec87-4eb4-a375-49b2bbe65d75&amp;v=&amp;b=&amp;from_search=2
4. https://www.w3schools.com
5. https://www.tutorialspoint.com/html/index.htm
6. https://www.tutorialspoint.com/css/index.htm
7. https://www.tutorialspoint.com/javascript/index.htm
8. https://www.tutorialspoint.com/jquery/index.htm
9. https://www.udemy.com/courses/development/web-development/

Practicals: UI/UX Design
Credit : 1
Marks : 25
Duration: 30 Hrs

List of practicals:
Suggested list of practical (Numbers in brackets indicate number of practicals)

1. Paper Prototyping using templates (1P)
2. Persona- conducting contextual interview and developing persona (1P)
3. Storyboarding (2P)
4. GUI design: Using FIGMA: (3P)
5. Web UI design: Using FIGMA: (3P)
6. Prototyping in FIGMA (3P)
7. Visualization and infographics (1P)
8. Heuristic Evaluation (1P)

A Mini Project combining all the technologies learnt using a front-end development framework such as bootstrap is recommended.

****

Course Title: E-Commerce
Course Code: UG-COM-SEC4
Marks: 50
Credits: 2
Duration:30 Hrs.

Prerequisite: Nil

Course Objectives:

- To analyze different e-commerce business models and strategies
- Identify key technologies and tools used in e-commerce
- Develop a basic e-commerce website

Course Outcomes:

At the end of the course students will be able to:

- CLO-1 : Understand the fundamental concepts and terminology of e-commerce.
- CLO-2 : Analyze different e-commerce business models and strategies.
- CLO-3 : Identify key technologies and tools used in e-commerce.
- CLO-4 : Develop a basic e-commerce website.

SYLLABUS

Unit 1: **[15 hrs]**

Introduction to E-Commerce

Overview of e-commerce, Benefits and challenges

E-Commerce Business Models

B2C, B2B, C2C, and other models, Subscription models

E-Commerce Technologies

Internet infrastructure, Payment systems, Security and privacy

Unit 2: **[15 hrs]**

E-Commerce Website Development

Website design principles, Content management systems (CMS), User experience (UX) and user interface (UI) design

E-Commerce Payment and Fulfillment

Payment gateways, Shipping and logistics, Inventory management

E-Commerce Marketing

Digital marketing strategies, SEO and SEM, Social media marketing Analytics

REFERENCES:

MANDATORY:

1. "E-commerce for Dummies" by Don Jones and Mark D. Scott - A comprehensive guide for beginners covering everything from setting up an online store to marketing and selling products online.

SUPPLEMENTARY:

1. Ravi Kalakota and Andrew B. Whinston.(2014).Electronic Commerce A Manager's Guide .Pearson Education.
2. "Don't Make Me Think: A Common Sense Approach to Web Usability" by Steve Krug
3. "E-commerce Evolved: The Essential Playbook to Build, Grow & Scale a Successful E-commerce Business" by Tanner Larsson
4. "Invisible Selling Machine" by Ryan Deiss

WEB BASED:

1) https://nptel.ac.in/
2)https://www.tutorialspoint.com/
3) http://www.wordpress.comhttp://www.wordpress.com/
4) https://www.wix.comhttp://www.wix.com/
5) https://www.shopify.comhttp://www.shopify.com/
6) https://www.blogger.comhttp://www.blogger.com/
7) https://www.google.co.in/analytics/
8)https://www.mailchimp.comhttp://www.mailchimp.com/
9)https://www.zendesk.com/
10) https://www.appointlet.com/
11) http://www.unbounce.com/

Practicals: E-Commerce
Credit : 1
Marks : 25
Duration: 30 Hrs

List of suggested practical's:

ECOMMERCE PLATFORMS:

1. WORDPRESS: (4P)
Primarily designed for creating blogs but can be used to create online store by adding appropriate themes & plugins. Basic Programming Knowledge in PHP may be required in later stages.

2. WIX: (2P)
Wix is a drag & drop website builder which can also be used to build an ecommerce website without any programming experience.

3. SHOPIFY: (2P)
Made specially to create online stores, add products, categories & handle payments all without Any Programming knowledge required.

4. BLOGGER : **(2P)**
Similar to WordPress but a blogging service provided by google which again can be customized to create an online store.

TOOLS USED TO FACILITATE ECOMMERCE (5P)

1. GOOGLE ANALYTICS: Present in most ecommerce platforms and can be integrated in almost any application, giving the owner insights of the customer base visiting the website. Can also be fine- tuned to give more detailed analysis like how many visits actually got converted into leads etc.

2. MAIL CHIMP: A service to send emails to customers. Useful in marketing.

3. ZENDESK : Zendesk is used to setup a support centre for your application users .Users can open a ticket and get their issues resolved. Useful in issue tracking and management.

4. APPOINTLET : A Service integrated with google calendar and helps manage all appointments. Useful in applications where an appointment is required.

5. UNBOUNCE : Used to create landing pages for an application . A good landing page sometimes defines whether a user will visit the site or turn away.

****

Course Title: Graphic Design
Course Code: UG-COM-SEC5
Marks: 50
Credits: 2
Duration: 30 Hrs
Course Prerequisites: Nil

Course Objectives:

- To understand the basic concept of Multimedia.
- To develop skills for designing graphical images.
- To develop skills for audio and video editing.
- To acquire skills in using audio/video editing software.

Course Outcomes:
Upon completion of the course students will be able to:

CLO-1: Understand the basic concepts and skills required for Multimedia
CLO-2: Demonstrate proficiency in the text, graphics, visual and audio medium via the respective tools used
CLO-3: Classify and realize the types of Authoring tools and their functions.
CLO-4: Assemble and deliver multimedia projects.

SYLLABUS:

UNIT I: Introduction to Multimedia and Computer Graphics:
[15HRS]
Commonly used terms associated with multimedia like CDROM, Storyboard, Script and Authoring tools. Stages of a Multimedia Project-Planning and Costing, Designing and Producing, Testing and Delivering. The Multimedia team and their roles- Project Manager, Writer, Video specialist, Audio specialist and Multimedia programmer. Multimedia Software. Multimedia Hardware. Social & Ethical considerations, Digital Representations & Standards. A survey of Computer Graphic:
Computer Aided Design, Presentation Graphics, Computer art, Entertainment, Education and Training, Visualization, Image Processing, Graphical User Interface
Multimedia Authoring Tools:
Types of authoring tools; Card and Page based tools; Icon-based; Time-based; Object-Oriented Tools.
TEXT: Text in graphics, character set, fonts, layout.
Character Attributes, Text Attributes, Marker Attributes, Bundled Text Attributes, Bundled Marker Attributes.
Vector graphics fundamentals, shapes, transforms and filters, Bitmapped graphics: resolution, image compression, manipulation, Geometrical transformations.

UNIT II: Audio, Video and Color [15HRS]

Basic Sound Concepts, Digitizing and processing sound, Music, Speech, Compression, formats, MIDI and Digital Audio Human vision, Camera systems, Gamma correction, Color matching, different Color models – RGB, CYMK, Transformations among color model.

Video:Analog and Digital Video, video standards, Video on PC. Introduction to graphics accelerator cards, Video Broadcast Standards - NTSC, PAL, SECAM, HDTV. Introduction to video capturing, Media & Instrument – Videodisk, DVCAM, Camcorder.

Recording Formats like S-VHA Video, Component (YUV), Component Digital, Composite Digital, and Video Hardware Resolutions.

Video Tips like shooting platforms, Lighting, Chroma Key or Blue Screen.

REFERENCES:

MANDATORY:

1. Chapman, N., & Chapman, J. (2005). Digital multimedia. John Wiley & Sons, Inc.
2. Li, Z. N., Drew, M. S., & Liu, J. (2004). Fundamentals of multimedia (pp. 253-265). Upper Saddle River (NJ): Pearson Prentice Hall.
3. Vaughan, Tay; Multimedia: Making it Work; Tata McGraw-Hill, 9th edition.

SUPPLEMENTARY:

1. Jeffcoate, J. (1995). Multimedia in practice. Technology and Applications. Great Britain: Prentice Hall.

WEB BASED:

1. https://www.gimp.org/
2. https://www.audacityteam.org/
3. https://swayam.gov.in/nd2_ugc19_hs42/
4. https://www.tutorialspoint.com/multimedia
5. https://libguides.bc.edu/

Practicals: Graphic Design
Credits : 1
Marks: 25
Duration: 30 Hours

Note : Practical can be done using GIMP, Audacity, Windows Video editor, Pencil 2D, Synfig2D or any other open-source image editing and audio video editing software

List of practicals:

1. Design a Brochure for a given product, give details. Learn about different Image file Formats (2P)
2. Design a poster with given information and learn about image compression (2P)
3. Record and Edit the Sound file and Learn about Effects and Filters of sound (3P)
4. Record your voice and learn about audio compression and audio mixing (3P)
5. Prepare Video content for a given information with title and special effects and filters.(3P)
6. Prepare effective video content (2P)

****

Course Title: Front End Web Development
Course Code: UG-COM-SEC6
Marks: 50
Credits: 2
Duration:30 Hrs

Course Prerequisites: Nil

Course objectives:

- Design good user interfaces.
- Apply design principles such as learnability, visibility, error prevention, efficiency and graphic design.

Course Outcomes:

On completion of the course students will be able to:

CLO-1: Design Content for a web application using HTML and CSS.

CLO-2: Style content so as to provide an effective User Interface.

CLO-3: Provide dynamism in the User Interface to enhance usability using Javascript and JQuery.

CLO-4: Develop a static web application.

SYLLABUS:

UNIT I: Web Essentials, HTML and CSS [15 HRS]

Clients, Servers, and Communication. The Internet-Basic Internet Protocols The World Wide Web-HTTP request message-response message-Web Clients; Web Servers, Domain and Hosting.

HTML - Introduction. The development process, basic HTML, formatting and fonts, commenting code, basic HTML tags, Meta tags.

HTML5 - Introduction, New Elements, Canvas, SVG, Drag/Drop, Geolocation, Video, Audio, Input types, form elements, form attributes, web storage, app cache, web workers, SSE

CSS - Introduction – Syntax, Id & Class, Backgrounds, Text, Fonts, Links, Lists, Tables. CSS Box Model – Border, Outline, Margin, Padding. Advanced - Grouping/Nesting, Dimension, Display, Positioning, Floating, Align, Pseudo-class, Pseudo-element, Image Gallery, Image Opacity, Image Sprites.

CSS3 - Introduction, Borders, Backgrounds, Gradients, Text Effects, Fonts, 2D Transforms, 3D Transforms, Transitions, Animations, Multiple Columns.

UNIT II: Dynamism in UI [15 HRS]

JavaScript - Introduction - What is JavaScript, Understanding Events, JavaScript Example, External JavaScript. Basic Elements – Comment, Variable, Global Variable, Data Types, Operators, If Statement, Switch, Loop: for and while, Function. JavaScript Objects – objects, Array. Browser Object Model - Browser Objects, Window Object, Document Object – get Element ById, get Elements ByName, getElementsByTagName, innerHTML property, inner Text property. Validation- form validation, email validation.

JQuery - Introduction - Syntax, Selectors, Events. Effects- Hide/Show, Fade, Slide, Animate, stop(), Call back, Chaining. HTML/CSS- Add, Remove, CSS Classes, css(), Dimensions, slider. Traversing – ancestors, descendants, siblings, filtering. Responsive Design. Web Accessibility and Search Engine Optimization.

REFERENCES:

MANDATORY:

1. D.T,(2018) Web Technologies, Black Book, Dream Tech

WEB BASED:

1.https://www.w3schools.com

2.https://www.tutorialspoint.com/html/index.htm

3.https://www.tutorialspoint.com/css/index.htm

4.https://www.tutorialspoint.com/javascript/index.htm

5.https://www.tutorialspoint.com/jquery/index.htm

6.https://www.udemy.com/courses/development/web-development/

Practicals : Web Designing

Marks: 25

Credits: 1

Duration: 30 Hrs

List of Practicals: (the numbers in brackets indicate number of practicals) :

1) Create a HTML page with the following: [3 P]

a) title heading paragraph emphasis strong and image elements

b) complex HTML table

c) simple HTML Form covering major form elements

d) Embed Video in an HTML page

3) Using CSS do the following: [3 P]

a) Create a Navigation bar (with dropdown) with CSS

b) Create a CSS Grid

c) Create a CSS3 based button

d) Make an image rounded shape

e) Create a CSS based sticky footer

f) Create CSS3 Corner Ribbon

g) Create CSS3 blurry text effect

h) Create CSS3 speech bubble shape

i) Create image cross fade with CSS3 transition

j) Set style for link hover active and visited states of hyperlink

4) Write JavaScript functions to : [3 P]

a) accept a string as a parameter and converts the first letter of each word of the string in upper case.

b) check whether a given credit card number is valid or not.

c) check whether a given value is a valid url or not.

d) check whether a given email address is valid or not.

e) print an integer with commas as thousands separators

f) remove items from a dropdown list.

5) Use J Query to : [3 P]

a) Disable buttons

b) Make textbox read only

c) Uncheck checkboxes

d) Confirm again

e) Sort

f) Switch rows and columns

6) A mini project combining all the technologies learnt using a front-end web development framework such as bootstrap is recommended.
[3 P]

****