Bengaluru North University

**National Education Policy - 2020
(NEP-2020)**

Syllabus for Bachelor of Computer Application (B.C.A)

(III & IV Semester)

Submitted to
Registrar
Bengaluru North University
Sri Devaraj Urs Extension
Tamaka, Kolar -563103

Curriculum Design/Syllabus Framing Committee

Sl. No	Name	Designation
1.	Mr. S.Manikandan Assistant Professor, Dept. of Computer Science, Government First Grade College, KR Puram, Bengaluru	Chairperson
2.	Dr. Murugan.K Assistant Professor, Dept. of Computer Science, Govt. College for Women, Kolar.	Member
3.	Dr. Hamela K Assistant Professor, Dept. of Computer Science, Government First Grade College , Malur,	Member
4.	Rashmi Rao K Associate Professor, Dept. of Computer Science, Government First Grade College, KR Puram, Bengaluru	Member
5.	Mr. Sankar Assistant Professor, Dept. of Computer Science, LBS Government First Grade College, RT Nagar, Bengaluru	Member
6.	Dr. Rajendirakumar Assistant Professor, Dept. of Computer Science, Govt. College for Women, Kolar.	Member
7.	Mr. Suresha K V Assistant Professor, Dept. of Computer Science, Government First Grade College, Chikkaballapur.	Member
8.	Mrs. Lakshmi Devi M S Assistant Professor, Dept. of Computer Science, Government First Grade College, Varthur, Bengaluru.	Member

Index:

Curriculum for BCA

Sem	Core Courses	Hour / Week		DS Elective Courses	Hours/ Week
		Theory	Lab		
III	Database Management Systems	3			
	C# and DOT NET Framework	3			
	Computer Communication and Networks	3			
	LAB: DBMS		4		
	LAB: C# and DOT NET Framework		4		
IV	Python Programming	3			
	Computer Multimedia and Animation	3			
	Operating Systems Concepts	3			
	LAB: Multimedia and Animation		4		
	LAB: Python programming		4		

Course Title: **Database Management System**	Course code: 21BCA3C7L
Total Contact Hours: 42	Course Credits: 03+02
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

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

- Explain the various database concepts and the need for database systems.
- Identify and define database objects, enforce integrity constraints on a database using DBMS.
- Demonstrate a Data model and Schemas in RDBMS.
- Identify entities and relationships and draw ER diagram for a given real-world problem.
- Convert an ER diagram to a database schema and deduce it to the desired normal form.
- Formulate queries in Relational Algebra, Structured Query Language (SQL) for database manipulation.
- Explain the transaction processing and concurrency control techniques.

DSC7: Database Management System (DBMS)

Unit	Description	Hours
1	**Database Architecture:** Introduction to Database system applications. Characteristics and Purpose of database approach. People associated with Database system. Data models. Database schema. Database architecture. Data independence. Database languages, interfaces, and classification of DBMS.	08
2	**E-R Model:** Entity-Relationship modeling: E – R Model Concepts: Entity, Entity types, Entity sets, Attributes, Types of attributes, key attribute, and domain of an attribute. Relationships between the entities. Relationship types, roles and structural constraints, degree and cardinality ratio of a relationship. Weak entity types, E -R diagram.	08
3	**Relational Data Model:** Relational model concepts. Characteristics of relations. Relational model constraints: Domain constrains, key constraints, primary & foreign key constraints, integrity constraints and null values.	10

	Relational Algebra: Basic Relational Algebra operations. Set theoretical operations on relations. JOIN operations Aggregate Functions and Grouping. Nested Sub Queries-Views. Introduction to PL/SQL & programming of above operations in PL/SQL	
4	**Data Normalization:** Anomalies in relational database design. Decomposition. Functional dependencies. Normalization. First normal form, Second normal form, Third normal form. Boyce-Codd normal form.	07
5	**Query Processing Transaction Management:** Introduction Transaction Processing. Single user & multiuser systems. Transactions: read & write operations. Need of concurrency control: The lost update problem, Dirty read problem. Types of failures. Transaction states. Desirable properties (ACID properties) of Transactions. Concurrency Control Techniques: Locks and Time stamp Ordering. Deadlock & Starvation.	09

References:
1. Fundamentals of Database Systems, Ramez Elamassri, Shankant B. Navathe, 7th Edition, Pearson, 2015
2. An Introduction to Database Systems, Bipin Desai, Galgotia Publications, 2010.
3. Introduction to Database System, C J Date, Pearson, 1999.
4. Database Systems Concepts, Abraham Silberschatz, Henry Korth, S.Sudarshan, 6th Edition, McGraw Hill, 2010.
5. Database Management Systems, Raghu Rama Krishnan and Johannes Gehrke, 3rd Edition, McGraw Hill, 2002

 Practical's:

 Student would be able to create tables, execute queries and PL/SQL programs.

1. Execute a single line query and group functions.
2. Execute DDL Commands.
3. Execute DML Commands
4. Execute DCL and TCL Commands.
5. Implement the Nested Queries.
6. Implement Join operations in SQL
7. Create views for a particular table
8. Implement Locks for a particular table
9. Write PL/SQL procedure for an application using exception handling.
10. Write PL/SQL procedure for an application using cursors.
11. Write a PL/SQL procedure for an application using functions
12. Write a PL/SQL procedure for an application using package

Course Title: **C# and Dot Net Framework**	Course code: 21BCA3C8L
Total Contact Hours: 42	Course Credits: 03+02
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:
- Describe Object Oriented Programming concepts like Inheritance and Polymorphism in C# programming language.
- Interpret and Develop Interfaces for real-time applications.
- Build custom collections and generics in C#.

DSC8: C# and Dot Net Framework

Unit	Description	Hours
1	**Introduction to .Net Technologies:** Introduction to Web Technologies. HTML Basics, Scripts. Sample Programs. Advantages and Disadvantages of Client-side and Server-side Scripts. Overview of Client-side Technologies and Server-side Technologies.	08
2	**Introduction to C#:** Overview of C#, Literals, Variables, Data Types, Operators, Expressions, Control Structures-Methods, Arrays, Strings, Structures, Enumerations. **OOPS with C#:** Classes, Objects, Inheritance, Polymorphism, Interfaces, Operator Overloading Delegates, Events, Errors and Exceptions.	08
3	**Introduction to VB.NET**: Introduction VB.NET -IDE – Creating a shortcut to start VB.NET. Maneuverings the Toolbar Auto-hide, Docking and Undocking, Placing and Resizing the Windows, Forms, Properties Window and Solution Explorer. Writing and Event Procedure. Execution Basic Keywords. Data Types. VB.NET statements. Conditional statements: If Else, Select Case, Switch and Choose Loops: Do, For Next, For Each Next, While loop. Arrays.	08
4	**Application Development on .NET:** C#.NET: Building Windows Applications, VB.NET: Windows Forms. Working with Controls, Timer, Picture-box, Group-box, Combo-box, Horizontal and Vertical Scrollbar, Numeric-up-down, Track-bar, and Progress-bar. Subroutines and Functions in VB.NET. Database applications	10
5	**ADO .NET Connectivity:** Introduction to ADO.NET, ADO vs ADO.NET. Architecture: Data reader, Data adopter, Accessing Data with ADO.NET. Programming Web Applications with Web Forms. ASP .NET applications with ADO.NET	08

References:
1. "Programming in C#", E. Balagurusamy, 4th Edition, Tata McGraw-Hill, 2017.
2. "Visual Basic.NET", Shirish Chavan, 3rd Edition, Pearson Education, 2009.
3. "ASP.NET and VB.NET Web Programming", Matt J. Crouch, Edition 2012.
4. "Computing with C# and the .NET Framework", Arthur Gittleman, 2nd Edition, Jones & Bartlett Publishers, 2011

Practicals:

1. Develop a C# .NET console application to demonstrate the conditional statements.
2. Develop a C# .NET console application to demonstrate the control statements.
3. Develop an application in C#.NET that demonstrates the windows controls
4. Demonstrate Multithreaded Programming in C#.NET
5. Demonstrate subroutines and functions in C#.NET
6. Develop an application for deploying various built-in functions in VB.NET
7. Develop an MDI application for Employee Pay-roll transactions in VB.NET
8. Construct a console application to demonstrate the OOP Concepts
9. Develop a web application in VB.NET for dynamic Login Processing
10. Develop a Windows application with database connectivity for core-banking transactions

Course Title: **Computer Communication and Networks**	Course code: 21BCA3C9L
Total Contact Hours: 42	Course Credits: 03
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

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

- Explain the transmission technique of digital data between two or more computers and a computer network that allows computers to exchange data.
- Apply the basics of data communication and various types of computer networks in real world applications.
- Compare the different layers of protocols.
- Compare the key networking protocols and their hierarchical relationship in the conceptual model like TCP/IP and OSI.

DSC9: Computer Communication and Networks

Unit	Description	Hours
1	**Introduction:** Computer Networks and its applications, Network structure, network architecture, Topologies, LAN, WAN, MAN, The OSI reference model, The TCP/IP reference model.	08
2	**The Physical Layer:** Transmission Media – Twisted pair, coaxial cable, optical fiber, radio transmission, microwaves and infrared transmission, Switching – message switching, Multiplexing.	07
3	**The Data Link Layer:** Data Link Layer design issues, Error detection – Single parity checking, Checksum, polynomial codes – CRC, Error correction- Hamming code, Elementary data link protocols, sliding window protocols	08
4	**The Network Layer:** Network layer design issues, Routing algorithms – Flooding, Distance vector routing, Hierarchical routing, Link state routing, Congestion, control algorithms – Leaky bucket, token bucket algorithm, admission control, Hop by Hop choke packets.	09
5	**The Transport Layer and Application Layer:** Elements of Transport service, Elements of Transport, protocols, Internet transport protocols (TCP & UDP), DNS, Electronic Mailing, and World Wide Web.	10
References:		

1. Computer Networks, Andrew S. Tanenbaum, 5th Edition, Pearson Education, 2010.
2. Data Communication & Networking, Behrouza A Forouzan, 3rd Edition, Tata McGraw

Hill,2001.
3. Data and Computer Communications, William Stallings,   10th Edition, Pearson
   Education, 2017.
4. Data Communication and Computer Networks, Brijendra Singh, 3rd Edition, PHI, 2012.
5. Data Communication & Network, Dr. Prasad, Wiley Dreamtech.
6. http://highered.mheducation.com/sites/0072967757/index.htmls

Course Title: **Python Programming**	Course code: 21BCA3C10L
Total Contact Hours: 42	Course Credits: 03+02
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:
- Explain the basic concepts of Python Programming.
- Demonstrate proficiency in the handling of loops and creation of functions.
- Identify the methods to create and manipulate lists, tuples and dictionaries.
- Discover the commonly used operations involving file handling.
- Interpret the concepts of Object-Oriented Programming as used in Python.
- Develop the emerging applications of relevant fields using Python.

DSC10: Python Programming

Unit	Description	Hours
1	**Introduction** to Features and Applications of Python; Python Versions; Installation of Python; Python Command Line mode and Python IDEs; Simple Python Program. **Python Basics:** Identifiers; Keywords; Statements and Expressions; Variables; Operators; Precedence and Association; Data Types; Indentation; Comments; Built-in Functions- Console Input and Console Output, Type Conversions; Python Libraries; Importing Libraries with Examples. Python Control Flow: Types of Control Flow; Control Flow Statements- if, else, elif, while loop, break, continue statements, for loop Statement; range () and exit () functions.	08
2	**Exception Handling:** Types of Errors; Exceptions; Exception Handling using try, except and finally. **Python Functions:** Types of Functions; Function Definition- Syntax, Function Calling, Passing Parameters/arguments, the return statement; Default Parameters; Command line Arguments; Key Word Arguments; Recursive Functions; Scope and Lifetime of Variables in Functions. **Strings:** Creating and Storing Strings; Accessing Sting Characters; the str() function; Operations on Strings- Concatenation, Comparison, Slicing and Joining, Traversing; Format Specifiers; Escape Sequences; Raw and Unicode Strings; Python String Methods.	08
3	**Lists:** Creating Lists; Operations on Lists; Built-in Functions on Lists; Implementation of Stacks and Queues using Lists; Nested Lists. Dictionaries: Creating Dictionaries; Operations on Dictionaries; Built-in Functions on Dictionaries; Dictionary Methods; Populating and Traversing Dictionaries. **Tuples and Sets:** Creating Tuples; Operations on Tuples; Built-in Functions	08

	on Tuples; Tuple Methods; Creating Sets; Operations on Sets; Built-in Functions on Sets; Set Methods.		
4	**File Handling:** File Types; Operations on Files– Create, Open, Read, Write, Close Files; File Names and Paths; Format Operator. **Object Oriented Programming:** Classes and Objects; Creating Classes and Objects; Constructor Method; Classes with Multiple Objects; Objects as Arguments; Objects as Return Values; Inheritance- Single and Multiple Inheritance, Multilevel and Multipath Inheritance; Encapsulation- Definition, Private Instance Variables; Polymorphism- Definition, Operator Overloading.		08
5	**GU Interface:** The tkinter Module; Window and Widgets; Layout Management- pack, grid and place. **Python SQLite:** The SQLite3 module; SQLite Methods- connect, cursor, execute, close; Connect to Database; Create Table; Operations on Tables- Insert, Select, Update. Delete and Drop Records. **Data Analysis:** NumPy- Introduction to NumPy, Array Creation using NumPy, Operations on Arrays; Pandas- Introduction to Pandas, Series and DataFrames, Creating DataFrames from Excel Sheet and .csv file, Dictionary and Tuples. Operations on DataFrames. **Data Visualisation:** Introduction to Data Visualisation; Matplotlib Library; Different Types of Charts using Pyplot- Line chart, Bar chart and Histogram and Pie chart.		10

References:

1. Think Python How to Think Like a Computer Scientist, Allen Downey et al., 2nd Edition, Green Tea Press. Freely available online @ https://www.greenteapress.com/thinkpython/thinkCSpy.pdf, 2015.
2. Introduction to Python Programming, Gowrishankar S et al., CRC Press, 2019.
3. Python Data Analytics: Data Analysis and Science Using Pandas, matplotlib, and the Python Programming Language, Fabio Nelli, Apress®, 2015
4. Advance Core Python Programming, MeenuKohli, BPB Publications, 2021.
5. Core PYTHON Applications Programming, Wesley J. Chun, 3rd Edition, Prentice Hall, 2012.
6. Automate the Boring Stuff, Al Sweigart, No Starch Press, Inc, 2015.
7. Data Structures and Program Design Using Python, D Malhotra et al., Mercury Learning and Information LLC, 2021.
8. http://www.ibiblio.org/g2swap/byteofpython/read/
9. https://docs.python.org/3/tutorial/index.html

Programs for Practical Component:

Part-A
1. Check if a number belongs to the Fibonacci Sequence
2. Solve Quadratic Equations
3. Find the sum of n natural numbers
4. Display Multiplication Tables
5. Check if a given number is a Prime Number or not
6. Implement a sequential search
7. Create a calculator program
8. Explore string functions

9. Implement Selection Sort
10. Implement Stack
11. Read and write into a file

Part-B
1. Demonstrate usage of basic regular expression
2. Demonstrate use of advanced regular expressions for data validation.
3. Demonstrate use of List
4. Demonstrate use of Dictionaries
5. Create SQLite Database and Perform Operations on Tables
6. Create a GUI using Tkinter module
7. Demonstrate Exceptions in Python
8. Drawing Line chart and Bar chart using Matplotlib
9. Drawing Histogram and Pie chart using Matplotlib
10. Create Array using NumPy and Perform Operations on Array
11. Create DataFramefrom Excel sheet using Pandas and Perform Operations on DataFrames

Note: A minimum of 10 Programs should be done in each Part.

Evaluation Scheme for Lab Examination:

Assessment Criteria		Marks
Program – 1 from Part A	Writing the Program	03
	Execution and Formatting	07
Program -2 from Part B	Writing the Program	03
	Execution and Formatting	07
Viva Voice based on Python Programming		05
Total		25

Course Title: **Computer Multimedia & Animation**	Course code: 21BCA3C11L
Total Contact Hours: 42	Course Credits: 03+02
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

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

- Understand basic elements using in web development
- Understand and develop animations using CSS
- Understand and develop HTML5-SVG animation

- Understand and develop HTML5-CANVAS animations

DSC11: Computer Multimedia & Animation

Unit	Description	Hours
1	Web Design: Origins and evolution of HTML, Basic syntax, Basic text markup, Images, Lists, Tables, Forms, Frame, Overview and features of HTML5. CSS: Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The <span> and <div> tags; Overview and features of CSS3. JavaScript: Object orientation and JavaScript; General syntactic characteristics; Primitives, operations, and expressions; Screen output and keyboard input.	10
2	Animation: What is an Animation? The Start and End States, Interpolation, Animations in HTML. All About CSS Animations, Creating a Simple Animation, Detailed Look at the CSS Animation Property, Keyframes, Declaring Multiple Animations, Wrap-up. All About CSS Transitions, Adding a Transition, Looking at Transitions in Detail, The Longhand Properties, Longhand Properties vs. Shorthand Properties, Working with Multiple Transitions.	09
3	HTML5 – SVG: Viewing SVG Files, Embedding SVG in HTML5, HTML5 − SVG Circle, HTML5 − SVG Rectangle, HTML5 − SVG Line, HTML5 − SVG Ellipse, HTML5 − SVG Polygon, HTML5 − SVG Polyline, HTML5 − SVG Gradients, HTML5 − SVG Star.	08
4	HTML5 – CANVAS: The Rendering Context, Browser Support, HTML5 Canvas Examples, Canvas - Drawing Rectangles, Canvas - Drawing Paths, Canvas - Drawing Lines, Canvas - Drawing Bezier Curves, Canvas - Drawing Quadratic Curves, Canvas - Using Images, Canvas - Create Gradients,	08
5	HTML5 - Styles and Colors, Canvas - Text and Fonts, Canvas - Pattern and Shadow, Canvas - Save and Restore States, Canvas - Translation, Canvas - Rotation, Canvas - Scaling, Canvas - Transforms, HTML5 Canvas - Composition, Canvas – Animations.	07

References:
1. The Complete Reference HTML and CSS, 5th Edition, Thomas A Powell, 2017.
2. Animation in HTML, CSS, and JavaScript, Kirupa Chinnathambi,   1st   Edition, Createspace Independent Pub, 2013.
3. https://www.w3.org/Style/CSS/current-work#CSS3
4. http://bedford-computing.co.uk/learning/cascading-style-sheets-css/

Practicals

Part A:

1) Write a HTML/5 program to demonstrate the use of Font family, font variant, font style, and font size.

2) Write a HTML/5 program to display random contents using list properties:
   a) Ordered list b) Unordered list

3) Write a HTML/5 program to create gradient using CSS.

4) Write a HTML/5 code to demonstrate following CSS animation properties:
   a) Delay   b) Direction   c) Duration

5) Write a HTML/5 program to demonstrate key frames

6) Write a HTML/5 code to demonstrate CSS transition and transformation.

7) Write a HTML/5 program to turn on/off a light bulb using JavaScript. Make use of .gif image and buttons.

Part B:

1) Write a HTML/5 program to draw rectangle, line, polygon, polyline using SVG.

2) Write a HTML/5 program to draw linear and radial gradient ellipse using SVG.

3) Write a HTML/5 program to draw a star using SVG.

4) Write a HTML/5 program to draw line, circle, rectangle, gradient, text using canvas.

5) Write a HTML/5 program to demonstrate translation, rotation, scaling, and transform using canvas.

6) Write a HTML/5 program to demonstrate Bezier Curves and Quadratic Curves.

7) Write a HTML/5 program to create canvas and add a red square onto the game area with up/down/left/right controller buttons.

8) Write a HTML/5 canvas program to add random size obstacles with red square controller button.

Evaluation Scheme for Lab Examination

Assessment Criteria		Marks
Program – 1 from Part A	Writing the Program	3
	Execution and Formatting	7
Program – 2 from Part B	Writing the Program	3
	Execution and Formatting	7
Viva Voice based on Lab Activities		5
Total		**25**

Course Title: **Operating System Concepts**	Course code: 21BCA3C12L
Total Contact Hours: 42	Course Credits: 03
Formative Assessment Marks: 40	Duration of SEE/Exam: 03 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:
- Explain the fundamentals of the operating system.
- Comprehend multithreaded programming, process management, process synchronization, memory management and storage management.
- Compare the performance of Scheduling Algorithms
- Identify the features of I/O and File handling methods.

DSC12: Operating System Concepts

Unit	Description	Hours
1	**Introduction to Operating System:** Definition, History and Examples of Operating System; Computer System organization; Types of Operating Systems; Functions of Operating System; Systems Calls; Operating System Structure. **Process Management:** Process Concept- Process Definition, Process State, Process Control Block, Threads; Process scheduling- Multiprogramming, Scheduling Queues, CPU Scheduling, Context Switch; Operations on Processes- Creation and Termination of Processes; Inter process communication (IPC)- Definition and Need for Inter process Communication; IPC Implementation Methods- Shared Memory and Message Passing;	08
2	**Multithreaded Programming**: Introduction to Threads; Types of Threads; Multithreading- Definition, Advantages; Multithreading Models; Thread Libraries; Threading Issues. **CPU Scheduling**: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Multiple-processor scheduling; Thread scheduling; Multiprocessor Scheduling; Real-Time CPU Scheduling.	10
3	**Process Synchronization:** Introduction; Race Condition; Critical Section Problem and Peterson's Solution; Synchronization Hardware, Semaphores; Classic Problems of Synchronization- Readers and Writers Problem, Dining Philosophers Problem; Monitors. **Deadlocks:** System Model; Deadlocks Characterization; Methods for Handling Deadlocks; Deadlock Prevention; Deadlock Avoidance; Deadlock Detection; and Recovery from Deadlock.	10

4	**Memory Management:** Logical and Physical Address Space; Swapping; Contiguous Allocation; Paging; Segmentation; Segmentation with Paging. **Virtual Memory:** Introduction to Virtual Memory; Demand Paging; Page Replacement; Page Replacement Algorithms; Allocation of frames, Thrashing.	08
5	**File System:** File Concepts- Attributes, Operations and Types of Files; File System; File Access methods; Directory Structure; Protection; File System Implementation- File System Structure, Allocation Methods, Free Space Management	06

References:

1. Operating System Concepts, Silberschatz' et al., 10thEdition, Wiley, 2018.
2. Operating System Concepts - Engineering Handbook, Ghosh PK, 2019.
3. Understanding Operating Systems, McHoes A et al., 7th Edition, Cengage Learning, 2014.
4. Operating Systems - Internals and Design Principles, William Stallings, 9th Edition, Pearson.
5. Operating Systems – A Concept Based Approach, Dhamdhere, 3rd Edition, McGraw Hill Education India.
6. Modern Operating Systems, Andrew S Tanenbaum, 4th Edition, Pearson.

Course Title: **Open Source Tools**	Course Credits: 2 (1L+0T+2P)
Semester: III	Duration of SEE: 01 Hour
Total Contact Hours: 13 hours of theory and 26-28 hours of practical's	SEE: 30 Marks IA: 20 Marks

Course Outcomes (COs):
- Recognize the benefits and features of Open Source Technology and to interpret, contrast and compare open source products among themselves
- Use appropriate open source tools based on the nature of the problem
- Write code and compile different open-source software.

Course Content (Open Source Tools)

Module	Details of topic	Duration
Module 1: Open Source Softwares	i. Introduction to Open sources, Need of Open Sources, Open Source –Principles, Standard Requirements, Advantages of Open Sources – ii. Free Software – FOSS iii. Licenses – GPL, LGPL, Copyrights, Patents, Contracts & Licenses and Related Issues iv. Application of Open Sources. Open Source Operating Systems : FEDORA, UBUNTU	05 hours
Module 2: Programming Tools And Techniques	i. Usage of design Tools like Argo UML or equivalent ii. Version Control Systems like Git or equivalent iii. Bug Tracking Systems (Trac, BugZilla) iv. BootStrap	04 hours
Module 3: Case Studies	i. Apache ii. Berkeley Software Distribution iii. Mozilla (Firefox) iv. Wikipedia v. Joomla vi. GNU Compiler Collection vii. Libre Office	04 hours

Text Book:
1. KailashVadera, Bhavyesh Gandhi, "Open Source Technology", Laxmi Publications Pvt. Ltd 2012, 1st Edition.

Reference Book:
1. Fadi P. Deek and James A. M. McHugh, "Open Source: Technology and Policy", Cambridge Universities Press 2007.

Course Title: **Python Programming Concepts**	Course Credits: 3 (3L+0T+0P)
Semester: III	Duration of SEE: 03 Hour
Total Contact Hours: 42	SEE: 60 Marks IA: 40 Marks

Course Outcomes (COs):

- Explain the fundamentals of Computers.
- Explain the basic concepts of Python Programming.
- Demonstrate proficiency in the handling of loops and the creation of functions.
- Identify the methods to create and store strings.

Unit I   Fundamentals of Computers   10 Hrs

Introduction to Computers - Computer Definition, Characteristics of Computers, Evolution and History of Computers, Types of Computers, Basic Organization of a Digital Computer; Number Systems – different types, conversion from one number system to another; Computer Codes – BCD, Gray Code, ASCII and Unicode; Boolean Algebra – Boolean Operators with Truth Tables; Types of Software – System Software and Utility Software; Computer Languages - Machine Level, Assembly Level & High Level Languages, Translator Programs – Assembler, Interpreter and Compiler; Planning a Computer Program - Algorithm, Flowchart and Pseudo code with Examples.

Unit II   Python Basics   10 Hrs

Introduction to Features and Applications of Python; Python Versions; Installation of Python; Python Command Line mode and Python IDEs; Simple Python Program. Identifiers; Keywords; Statements and Expressions; Variables; Operators; Precedence and Association; Data Types; Indentation; Comments; Built-in Functions- Console Input and Console Output, Type Conversions; Python Libraries; Importing Libraries with Examples; Illustrative programs.

Unit III   08 Hrs

Python Control Flow: Types of Control Flow; Control Flow Statements- if, else, elif, while loop, break, continue statements, for loop Statement; range() and exit () functions; Illustrative programs.

Unit IV   08 Hrs

Python Functions: Types of Functions; Function Definition- Syntax, Function Calling, Passing Parameters/arguments, the return statement; Default Parameters; Command line Arguments; Key Word Arguments; Illustrative programs.

Unit V   **06 Hrs**

Strings: Creating and Storing Strings; Accessing Sting Characters; the str() function; Operations on Strings- Concatenation, Comparison, Slicing and Joining, Traversing; Format Specifiers; Escape Sequences; Raw and Unicode Strings; Python String Methods; Illustrative programs.

References
1. Computer Fundamentals (BPB), P. K. Sinha & Priti Sinha
2. Think Python How to Think Like a Computer Scientist, Allen Downey et al., 2nd Edition, Green Tea Press. Freely available online 2015.
   @https://www.greenteapress.com/thinkpython/thinkCSpy.pdf
3. Introduction to Python Programming, Gowrishankar S et al., CRC Press, 2019.
4. http://www.ibiblio.org/g2swap/byteofpython/read/
5. http://scipy-lectures.org/intro/language/python_language.html
6. https://docs.python.org/3/tutorial/index.html

Open Elective

PRINCIPLES OF INTERNET

Course Title: **PRINCIPLES OF INTERNET**	Course Credits: 3 (3L+0T+0P)
<mark>Semester: IV</mark>	Duration of SEE: 03 Hour
Total Contact Hours: 42	SEE: 60 Marks IA: 40 Marks

Course Outcomes (COs):
- Able to explain understand the internet standards and recent web technologies like Conferencing, newsgroup etc.
- Able to implement, compile, test and run HTML program
- Able to explain understand web Browsers.

UNIT I   **09 Hrs**

Basics of Networks – Advantages and disadvantages- Types of networks - LAN, WAN, MAN - Topologies of Networks – Bus Topology, Ring Topology, Star Topology, Mesh Topology, Tree Topology.

UNIT II   **06 Hrs**

Internet Basics – Advantages and disadvantages – Internet Applications

– IP Address- Internet Services– URL- E-mail - E-mail features – Emil Providers.

UNIT III   **09 Hrs**

Introduction to Internet Protocol –TCP –UDP-FTP – The Telnet –The Usenet – Internet Chat.

           09 Hrs

UNIT IV

WWW Overview – Web pages -The Web Servers – Web Browsers– Search Engines- Introduction to Firewalls – Uses of Firewalls – Types of Firewalls.

UNIT V   **09 Hrs**

Art of creating a website – Hypertext and HTML – Features – Elements – Attributes- Structure of HTML– Basic tags in HTML – Formatting tags in HTML – HTML Lists –Types of Lists.

Text Book:

1. Raj Kamal , "Internet and Web Technologies", Mc GrawHill Education,2007.

References:
1. Joe krayank & Joe Habraken, "Internet 6 in 1", Prentice Hall of India Private Limited, New Delhi, 1998.
2. "Internet Complete", BPB publications, New Delhi, 1998.