Rani Channamma University Belagavi

Vidyasangama, NH-04, Bhutaramanahatti, Belagavi – 591 156

Bachelor of Computer Applications (B.C.A.)

Syllabus for III and IV Semester

(as per National Education Policy – 2020)

2022-23 onwards

RANI CHANNAMMA UNIVERSITY

Vidyasangama, NH-04, Bhutaramanahatti, Belagavi – 591 156

SYLLABUS

Bachelor of Computer Applications (B.C.A.)

(as per National Education Policy – 2020)

Submitted by

Dr. Parashuram Bannigidad
Chairperson BoS (UG) – Rani Channamma University, Belagavi

Members of Board of Studies

1. Dr. Vijayalaxmi M Belagumpi — Member
   Assistant Professor,
   Dept. of Computer Science,
   GFGC Vijayapur
2. Dr. Bhagirathi Halalli — Member
   Assistant Professor,
   Dept. of Computer Science,
   GFGC Raibag

SEMESTER-3										
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams(Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
L-5	21BCA3L5LK3	Kannada	40	60	100	4	0	0	3	2
	21BCA3L5LFK3	Functional Kannada								
L-6	21BCA3L6EN3	English	40	60	100	4	0	0	3	2
	21BCA3L6HI3	Hindi								
	21BCA3L6SN3	Sanskrit								
	21BSC3L6TE3	Telugu								
	21BCA3L6UR3	Urdu								
DSC7	21BCA3C7L	Database Management System	40	60	100	3	0	0	3	2
	21BCA3C7P	DBMS Lab	25	25	50	0	0	3	2	3
DSC8	21BCA3C8L	C# and .Net Framework	40	60	100	3	0	0	3	2
	21BCA3C8P	C# and .Net Framework Lab	25	25	50	0	0	3	2	3
DSC9	21BCA3C9L	Computer Communication and Networks	40	60	100	3	0	0	3	2
OEC3	21BCA3O3PPCL	Python Programming Concepts	40	60	100	3	0	0	3	2
AECC2	21BCA3AE2CIL	Constitution of India	20	30	50	1	0	2	2	2
VBC5	21BCA3V5PE3	Physical Education – Sports	25	-	25	-	-	2	1	-
VBC6	21BCA3V6NC2	NCC/NSS/R&R(S&G) / Cultural	25	-	25	-	-	2	1	-
Total Marks					**800**	**Semester Credits**			**26**	

SEMESTER-4										
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams(Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
L-7	21BCA4L7LK4	Kannada	40	60	100	4	0	0	3	2
	21BCA4L7LFK4	Functional Kannada								
L-8	21BCA4L8EN4	English	40	60	100	4	0	0	3	2
	21BCA4L8HI4	Hindi								
	21BCA4L8SN4	Sanskrit								
	21BSC4L8TE4	Telugu								
	21BCA4L8UR4	Urdu								
DSC10	21BCA4C10L	Python Programming	40	60	100	3	0	0	3	2
	21BCA4C10P	Python Programming Lab	25	25	50	0	0	3	2	3
DSC11	21BCA4C11L	Multimedia & Animation	40	60	100	3	0	0	3	2
	21BCA4C11P	Multimedia & Animation Lab	25	25	50	0	0	3	2	3
DSC12	21BCA4C12L	Operating System Concepts	40	60	100	3	0	0	3	2
OEC4	21BCA4O4ECL	Financial Education and Investment Awareness	20	30	50	3	-	-	2	1hr30min
SEC2	21BCA4SE2OST	Open Source Tools	20	30	50	1	0	2	2	2
VBC7	21BCA4V7PE4	Physical Education – Sports	25	-	25	-	-	2	1	-
VBC8	21BCA4V8NC3	NCC/NSS/R&R(S&G) / Cultural	25	-	25	-	-	2	1	-
Total Marks					**750**	**Semester Credits**			**25**	
Exit option with Diploma in Computer Applications (with the completion of courses equivalentto a minimum of 96 credits)					**3150**	**II year Credits**			**103**	

Syllabus for BCA III and IV Semesters

III Semester Curriculum for BCA

Sem	Core Courses	Hour / 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**
	OEC: Python Programming Concepts	**3**	**-**
IV	Python Programming	**3**	
	Computer Multimedia and Animation	**3**	
	Operating Systems Concepts	**3**	
	LAB: Python programming		**4**
	LAB: Multimedia and Animation		**4**

Course Content for BCA III Semester

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

Year	II	**Course Code:** 21BCA3C7P	**Credits**	02
Sem.	III	**Course Title: DBMS LAB**	**Hours**	40
Course Pre-requisites, ifany:		Knowledge of Programming		
Formative Assessment Marks: 25		Summative Assessment Marks: 25	Duration of ESA: 03 hrs.	

Practicals:

CO: 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

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 **DBMS**		05
Total		25

Course Title: **C# and Dot Net Framework**	Course code: **21BCA3C8L**
Total Contact Hours: 42	Course Credits: 03
Formative Assessment or IA Marks: 40	Duration of SEE/Exam: 02 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 The C# language, the .Net Architecture and .Net Framework, The Common Language Runtime (CLR), Microsoft Intermediate Language (MSIL) Code, Just In Time Compilers (JITers),The Framework Class Library (FCL), The Common Languages Specification (CLS), The Common Type System (CTS), The .Net Framework, Working with Visual Studio.Net, Similarities and Differences between C# and C++, Java, and Visual Basic, Understanding the HELLO WORLD Application Code, The System. Environment Class, The System. Console Class, Namespaces in C#, The using Keyword, The class Keyword, The Main() Method, Printing on the Console, Comments.	10
2	C# Basics: Data Types, Variables & Constants, Operators in C#, Arithmetic Operators, Prefix and Postfix notation, Assignment Operators, Relational Operators, Other Operators, Operators precedence, Flow Control and Conditional Statements. Object and Classes: Concept of a class, Objects, Fields, Methods, Access modifiers, Properties, Static members of the class, Constructors, Destructors, Method overloading.	08
3	Pillars of OOP, Encapsulation support, Class properties, C#"s Inheritance Support, C #'s Polymorphic Support, Interface: Deriving classes, calling base class constructor, Overriding Methods, Non-Inheritable Classes, Abstract Class, Interface Inheritance, Namespace and Access Modifiers, Boxing and Un-boxing. .NET Delegate type, defining a Delegate in C#, System. Delegate Base Classes, Delegate examples, C# Events, operator overloading.	08
4	Exception Handling: Handling Exceptions using try and catch, Raising Exceptions using throw, Pre- defined Exception classes, Custom Exception classes, Understanding Object Lifetime classes, Objects and References, the basics of Object Lifetime, System. GC type. Assemblies-The Role of .NET Assemblies, Understanding the format of .NET Assemblies, single file assembly, multifile assembly, Private and Shared Assemblies.	08

5	Working with Collections: List and Dictionary, Array List and Hash Table, Generic Classes, Comparable and Sorting, WinForms: Introduction, Controls, Menus and Context Menus, Menu Strip, Toolbar Strip, Graphics and GDI, SDI and MDI Applications, Dialog box (Modal and Modeless), Form Inheritance, Developing Custom, Composite and Extended Controls.	08

References:

1. E. Balagurusamy, Programming in C#, Tata McGraw Hill
2. Stephen Walthert, ASP.NET 3.5 unleased, SAMS
3. ShibiPanikkar and Kumar Sanjeev, C# with .NET Frame Work, Firewall Media
4. Jeffrey Richter, Applied Microsoft .Net Framework Programming, (Microsoft)

Additional Reading:

5. http://www.bestdotnettraining.com
6. http://www.bestsharepointtraining.com
7. https://stackoverflow.com/documentation
8. Troelsen, Andrew, Pro C# 5.0 and the .NET 4.5 Framework, 6th Edition, APress, India

Year	II	**Course Code: 21BCA3C8P**	**Credits**	02
Sem.	III	**Course Title: C# and Dot Net Framework LAB**	**Hours**	40
Course Pre-requisites, ifany:		Knowledge of Programming		
Formative Assessment Marks: 25		Summative Assessment Marks: 25	Duration of ESA: 03 hrs.	

Practicals:

1. Write a C# program to show the machine details like machine name, Operating System, Version, Physical Memory and calculate the time since the Last Boot Up. (Hint: Use System. Environment Class)
2. Write a program in C# Sharp to count a total number of alphabets, digits and special characters in a string
3. Write a program in C# Sharp to create a function to calculate the sum of the individual digits of a given number.
4. Draw a square with sides 100 pixels in length. Then inscribe a circle of radius 50 inside the square. Position the square and the inscribed circle in the middle of the screen.
5. Write a program to implement multilevel inheritance.
6. Write a program to demonstrate System exception.
7. Write an object oriented program to demonstrate bank transaction. Include methods for amount deposit, amount withdrawal and display.
8. Demonstrate operator overloading two complex numbers.
9. Demonstrate Dialog box (Modal and Modeless).
10. Write a program in C# Sharp to create Menu and menu items.

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 **C# and Dot Net Framework**		05
Total		25

Course Title: **Computer Communication and Networks**	Course code: **21BCA3C9L**
Total Contact Hours: 42	Course Credits: 03
Formative Assessment or IA Marks: 40	Duration of SEE/Exam: 02 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

Open Elective for III Semester

OEC3: Python Programming Concepts

Course Code: **21BCA3O3PPCL** Course Title: **Python Programming Concepts**	Course Credits: 3 (3L+0T+0P)
Semester: III	Duration of SEE: 02 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 **6 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

Course Content for BCA IV Semester

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

Year	II	**Course Code: 21BCA4C10P**	**Credits**	02
Sem.	III	**Course Title: Python Programming LAB**	**Hours**	40
Course Pre- requisites, ifany:		Knowledge of Programming		
Formative AssessmentMarks: 25		Summative Assessment Marks: 25	Duration of ESA: 03 hrs.	

Practicals:

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 DataFrame from Excel sheet using Pandas and Perform Operations on Data Frames

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
Formative Assessment or IA Marks: 40	Duration of SEE/Exam: 02 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

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

- Write a well-designed, interactive Web site with respect to current standards and practices.
- Demonstrate in-depth knowledge of an industry-standard multimedia development tool and its associated scripting language.
- Determine the appropriate use of interactive versus standalone Web applications.

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 and 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/

Year	II	**Course Code:** 21BCA4C11P	**Credits**	02
Sem.	III	**Course Title: Computer Multimedia & Animation LAB**	**Hours**	40
Course Pre-requisites, ifany:		Knowledge of Programming		
Formative Assessment Marks: 25		Summative Assessment Marks: 25	Duration of ESA: 03 hrs.	
		Practicals: **Part A** 1. Program to Design LOG IN Form in Html. 2. Program for Creating animation of "Bouncing Cloud" using HTML and CSS 3. Program to demonstrate a keyframe animation. 4. Program to demonstrate a Font style, font weight, and font size properties using CSS. 5. Program to demonstrate multiple animations. 6. Program to use table tag to format web page. Also create the Time Table of your class using table tag. 7. Program to Demonstrate Longhand properties in CSS. 8. Program to Demonstrate shorthand properties in CSS. 9. Program to Demonstrate animation in reverse direction or alternate cycles. 10. Write JavaScript Program to show light ON/OFF Demo		
		Part B 1. Program to Demonstrate SVG (Scalable Vector Graphics) Circle. 2. Program to Demonstrate SVG (Scalable Vector Graphics) Eclipse. 3. Program to Demonstrate SVG (Scalable Vector Graphics) Star. 4. Program to demonstrate "**StrokeText()"** method using HTML Canvas. 5. Program to demonstrate **BezierCurveTo()** method using HTML canvas. 6. Program to demonstrate different **line patterns** with different colors using Canvas. 7. Program to demonstrate **Gradients** using HTML Canvas. 8. Program to demonstrate **Text shadows** using HTML Canvas. 9. Program to Demonstrate Source-Over, Source-in, and Source-Out properties for composition using HTML Canvas. 10. Program to create a rectangle and animate increase and decrease the size of rectangle.		

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 **Computer Multimedia & Animation**		05
Total		25

Course Title: **Operating System Concepts**	Course code: **21BCA3C12L**
Total Contact Hours: 42	Course Credits: 03
Formative Assessment or IA Marks: 40	Duration of SEE/Exam: 02 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., 10th Edition, 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.

(Skill Enhancement Course: SEC for BCA Course)
Open Source Tools

Semester: IV

Course Code: 21BCA4SE2OST Course Title: **Open Source Tools**	Course Credits: 2 (1L+0T+2P)
Semester: IV	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:

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

Note:

Semester End Exam Question Paper Pattern for Skill Enhancement Course (SEC) is of Multiple Choice Questions

Semester End Exam Question Paper Pattern

Duration of the examination: **2hour** **Max. Marks:60**

Section A

Answer any TEN from the following, each carries 2 marks: [10X2=20]

1. **----------**
2. **----------**
3. **----------**
4. **----------**
5. **----------**
6. **----------**
7. **----------**
8. **----------**
9. **----------**
10. **----------**
11. **----------**
12. **----------**

Section B

Answer any FOUR from the following questions each carries 5 marks.

[4X5=20]

13. **----------**
14. **----------**
15. **----------**
16. **----------**
17. **----------**

Section C

Answer Any two from the following questions each carries 10 marks.

(The Question may consist two sub-questions)

[2X10=20]

18. **----------**
19. **----------**
20. **----------**

Theory Paper IA 40 Marks distribution

	C1	C2	Total Marks
First IA	Test-1: 15marks	Assignment/Activity-1: 05Marks	**20**
Second IA	Test-2: 15marks	Assignment/Activity-2 : 05Marks	**20**
	30	**10**	**40**

Theory Paper IA 20 Marks distribution

	C1	C2	Total Marks
First IA	Test-1: 10 marks	--	**10**
Second IA	Test-2: 5marks	Assignment/Activity-2 : 05	**10**
	15	**5**	**20**