BAGALKOT UNIVERSITY , JAMAKHANDI

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

Syllabus for IV Semester

As Per NEP – 2020 and Adapted from
RCU Belagavi Applicable from the
Academic

Year 2024-25

Curriculum Structure for BCA IV Program of BGKUB as per NEP 2020 w.e.f. 2024-25

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	----------	Languages	40	60	100	4	0	0	3	2
L-8	----------	Languages	40	60	100	4	0	0	3	2
DSC10	126BCA04XXXDSC11T	Python Programming	40	60	100	3	0	0	3	2
	126BCA04XXXDSC11L	Python Programming Lab	25	25	50	0	0	3	2	3
DSC11	126BCA04XXXDSC12T	Computer Multimedia & Animation	40	60	100	3	0	0	3	2
	126BCA04XXXDSC12L	Computer Multimedia & Animation Lab	25	25	50	0	0	3	2	3
DSC12	126BCA04XXXDSC13T	Operating System Concepts	40	60	100	3	0	0	3	2
SEC	126BCM03XXXSEC01T	Financial Education and Investment Awareness	20	30	50	3	-	-	2	1hr30min
SEC	126BCA04XXXSEC01T	Open Source Tools	20	30	50	1	0	2	2	2
VBC7	126COM04XXXVBC07B	NCC/NSS/R&R(S&G) / Cultural	25	-	25	-	-	2	1	-
VBC8	126COM04XXXVBC08B	Physical Education – Sports	25	-	25	-	-	2	1	-
Total Marks					**750**	**Semester Credits**			**25**	
Exit option with Diploma in Computer Applications (with the completion of courses equivalent to a minimum of 96 credits)					**3150**	**II year Credits**			**103**	

Syllabus for BCA IV Semesters

IV Semester Curriculum for BCA

Sem	Core Courses	Hour / Week	
		Theory	**Lab**
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 IV Semester

Course Title: **Python Programming**	Course code: 126BCA04XXXDSC11T
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., 2n 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:** 126BCA04XXXDSC11L	**Credits**	02
Sem.	III	**Course Title: Python Programming LAB**	**Hours**	40
Course Pre- requisites, if any:		Knowledge of Programming		
Formative Assessment Marks: 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: 126BCA04XXXDSC12T
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:** 126BCA04XXXDSC12L	**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: 126BCA04XXXDSC13T
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

CourseCode:126BCA04XXXSEC01T 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**