



RANI CHANNAMMA UNIVERSITY, BELAGAVI

Revised syllabus of BCA(BCA 4) course w.e.f. academic year 2019-20 and onwards

BCA revised syllabus (w.e.f. 2019-20 and onwards)							
Subject Code	Subject Title	Teaching Scheme Hrs/week		Exam. Duration (Hrs)	Examination		
		Theory	Practical		Theory/ Practical	IA	Total
BCA V Semester (w.e.f. 2019-20 and onwards)							
17BCASWET51	Software Engineering	4	--	3	80	20	100
17BCASPTT52	Software Programming and Testing	4	--	3	80	20	100
17BCACYST53	Cyber Security	4	--	3	80	20	100
17BCAPPYT54	Programming with Python	4	--	3	80	20	100
17BCANFCT55	.Net Framework Using C#	4	--	3	80	20	100
17BCAAPLP56	Android and Python Programming lab	--	4	3	80	20	100
17BCADNLP57	C# Lab.	--	4	3	80	20	100
BCA VI Semester (w.e.f. 2019-20 and onwards)							
17BCACLCT61	Cloud Computing	4	4	3	80	20	100
17BCAWDPT62	Web Designing and Programming	4	4	3	80	20	100
17BCABINT63	Business Intelligence	4	4	3	80	20	100
17BCAWPLP64	Web Programming Lab.	--	4	3	80	20	100
17BCAPADP65	Project- Application Development	--	9	3	240	60	300



17BCASWET51: Software Engineering

Teaching Hours: 4 Hrs/week

**Marks: Main Exam: 80
IA: 20**

UNIT-I

10Hrs

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, legacy software, Software Myths. A Generic View of Process: Software engineering-A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process Patterns, Process Assessment, personal and team process models. Process Models: The Waterfall model, Incremental process models, Evolutionary Process Models, Specialized Process Models, The Unified Process.

UNIT-II

10Hrs

Software Requirements: Functional and non-Functional Requirements, User Requirements, System Requirements, Interface Specification, the software requirement document. Requirement engineering process: Feasibility studies, Requirements elicitation and analysis, requirements validation, Requirements management. System models: Context Models, behavioral models, Data models, object models, structured method.

UNIT-III

10Hrs

Design Engineering: Design process and design quality, Design concepts, the design model, pattern based software design. Creating an Architectural Design: Software architecture, Data design, Architectural Styles and patterns, Architectural design, assessing alternative architectural designs, mapping data flow into software architecture. Modeling Component-level design: designing class – based components, conducting component-level design, object constraint language, designing conventional components. Performing User interface design: Golden rules, User interface analysis and design, interface analysis, interface design steps, Design evaluation.

UNIT-IV

10Hrs

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, system testing, the art of debugging. Product metrics: Software Quality, Frame work for product metrics, Metrics for Analysis Model, Metrics for Design Model, metrics for source code, metrics for testing, metrics for maintenance. Metrics for process and products: Software Measurement, Metrics for software quality.

UNIT-V

10Hrs

Risk management: Reactive vs. Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM plan. Quality Management: Quality concepts, software quality assurance, Software Reviews, Formal technical Reviews, Statistical Software quality Assurance, Software reliability, The ISO 9000 quality standards.

References:

1. Roger S Pressman, Software engineering a practitioner's approach, Seventh Edition, Mc Graw Hill International Edition.
2. Ian Sommerville, Software Engineering, Seventh edition, Pearson education.

Additional Reading:

3. PankajJalote, Software Engineering- a Precise Approach, Wiley India, 2010.
4. Waman S Jawadekar, Software Engineering: A primer, Tata McGraw-Hill, 2008.
5. Stephan Schach, Software Engineering, Tata McGraw Hill.



RANI CHANNAMMA UNIVERSITY, BELAGAVI

17BCASPTT52: Software Practices and Testing

Teaching Hours: 4 Hrs/week

Marks: Main Exam: 50

IA: 20

UNIT I: 10 Hrs
Principles of Testing, Software Development Life Cycle Models (SDLC), Phases of Software Project, Quality, Quality Assurance and Quality Control, Testing, Verification and Validation, Life Cycle Models, White Box Testing: White Box Testing, Static Testing, Structural Testing, Challenges in White Box Testing.

UNIT II: 10 Hrs
Testing Techniques: Black Box Testing, Integration Testing, Top-Down Integration, Bottom-Up Integration, Bi-Directional Integration, Defect Bash, System and Acceptance Testing, Functional versus Non-functional Testing, Functional System Testing, Non-Functional System Testing, Acceptance Testing.

UNIT III: 10 Hrs
Performance Testing: Factors, Methodology, Tools, Process for performance testing, Regression Testing, Types, Testing of Object-oriented Systems, Usability and Accessibility Testing, approach, Quality factors, Aesthetics Testing, Accessibility Testing, Tools for Usability.

UNIT IV: 10 Hrs
Common People Issues: Perceptions and Misconceptions About Testing, comparison between Testing and Development Functions, Providing Career Paths for Testing Professionals, The role of the Ecosystem and a call for Action. Organization Structures for testing teams: Dimensions of Organization Structures, Structures in Single product Companies, Structures for Multi-Product Companies. (14.1 to 14.3).

UNIT V: 10 Hrs
Test Planning, Management: Test Planning: Preparation, scope management, Test approach, setting up criteria, Identifying responsibilities, test deliveries, testing tasks, activity breakdown, communication and risk management. Test Management: Choice of standards, Test Infrastructure Management, Integrating with Product release. (15.1 to 15.3) Software Test Automation: Introduction, Terms used, Skills needed, scope of automation: Identifying the types of testing amenable to automation.(16.1 to 16.4.1)

TEXT BOOKS:

Srinivasan Desikan, Gopalaswamy Ramesh: Software testing Principles and Practices, 2nd Edition, Pearson, 2012.

REFERENCE BOOKS:

- Software Testing : Aditya Mathur.
- Software Testing, Ron Patton, Second Edition, SAMS Pearson Publication 2011
- The Craft of Software Testing, Brain Marick, Pearson Publication 2010



17BCACYST53: Cyber Security

Teaching Hours: 4 Hrs/week

**Marks: Main Exam: 80
IA: 20**

UNIT I

08 Hrs

Introduction to Cybercrime: Cybercrime definition and origins of the world, Cybercrime and information security, Classifications of cybercrime, Cybercrime and the Indian ITA 2000, A global Perspective on cybercrimes.

UNIT II

10 Hrs

Cyber offenses & Cybercrime: How criminal plan the attacks, Social Engg, Cyber stalking, Cybercafé and Cybercrimes, Botnets, Attack vector, Cloud computing, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication Service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for Organizations, Organizational Measures for Handling Mobile, Devices-Related Security Issues, Organizational Security Policies and Measures in Mobile Computing Era, Laptops.

UNIT III

12 Hrs

Tools and Methods Used in Cyberline: Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Virus and Worms, Steganography, DoS DDoS Attacks, SQL Injection, Buffer Over Flow, Attacks on Wireless Networks, Phishing, Identity Theft (ID Theft) Cybercrimes and Cybersecurity: The Legal Perspectives Why do we need Cyberlaw: The Indian Context, The Indian IT Act, Digital Signature and the Indian IT Act, Amendments to the Indian IT Act, Cybercrime and Punishment, Cyberlaw, Technology and Students: Indian Scenario.

UNIT IV

10 Hrs

Understanding Computer Forensics: Historical Background of Cyberforensics, Digital Forensics Science, The Need for Computer Forensics, Cyberforensics and Digital Evidence, Forensics Analysis of Email, Digital Forensics Lifecycle, Chain of Custody Concept, Network Forensics, Approaching a Computer Forensics Investigation, Setting of a Computer Forensics Laboratory: Understanding the Requirements, Computer Forensics and Steganography, Relevance of the OSI 7 Layer Model to the Computer Forensics and Social Networking Sites: The Security/Privacy Threats, Forensics Auditing, Anti Forensics.

UNIT IV

10 Hrs

Cybersecurity: Organizational Implications Cost of Cybercrimes and IPR Issues: Lesson for Organizations, Web Treats for Organizations: The Evils and Perils, Security and Privacy Implications from Cloud Computing, Social Media Marketing: Security Risk and Perils for Organization, Social Computing and the Associated Challenges for Organizations, Protecting People's Privacy in the Organization, Organizational Guidelines for Internet Usage, Safe Computing Guidelines and Computer Usage Policy, Incident Handling: An Essential Component, Intellectual Property in the Cyberspace of Cybersecurity, Importance of Endpoint Security in Organizations.

Reference:

1. Nina Godbole, SunitBelapure, Cyber Security, Wiley India, New Delhi
2. Marjie T. Britz , Computer Forensics and Cyber Crime An Introduction, Pearson publication,
3. Nina Godbole, Information Systems Security, Wiley India, New Delhi
4. Kenneth J. Knapp, Cyber Security & Global Information Assurance Information Science Publishing.
5. William Stallings, Cryptography and Network Security, Pearson Publication



RANI CHANNAMMA UNIVERSITY, BELAGAVI

Additional Reading:

6. John W. Rittinghouse, William M. Hancock, Cyber Security Operations Handbook, ElsevierPub.
7. Dr. Surya Prakash Tripathi, Ritendra Goyal, Praveen Kumar Shukla ,”Introduction to Information Security and Cyber Law” Willey Dreamtech Press.

W.e.f. 2019-20 & onwards



17BCAPPYT54: Programming With Python

Teaching Hours: 4 Hrs/week

Marks: Main Exam: 80

IA: 20

Preamble: Python is an extremely easy-to-use programming language that has steadily gained in popularity over the last few years. Python is great for backend web development, data analysis, artificial intelligence, and scientific computing. Many developers have also used Python to build productivity tools, games, and desktop apps.. Some major ways in which Python is used commercially include Corel, D-Link, Eve-Online, HP, Honeywell, and Industrial Light and Magic. With the rise of big data, Python developers are in demand as data scientists, especially since Python can be easily integrated into web applications to carry out tasks that require machine learning.

UNIT I

10 Hrs

Introduction to Python: The basic elements of python, Branching Programs, Control Structures, Strings and Input, Iteration.Functions, Scoping and Abstraction: Functions and scoping, Specifications, Recursion, Global variables, Modules, Files, System Functions and Parameters.

UNIT II

10 Hrs

Structured Types, Mutability and Higher-Order Functions: Strings, Tuples, Lists and Dictionaries, Lists and Mutability, Functions as Objects

UNIT III

10 Hrs

Testing, Debugging, Exceptions and Assertion: Types of testing – Black-box and Glass-box, Debugging, Handling Exceptions, Assertions

Object-Oriented Programming: Abstract Data Types and Classes, Inheritance, Encapsulation and Information Hiding.

UNIT IV

10 Hrs

Regular Expressions: Concept of regular expression, various types of regular expressions, using match function.GUI Programming in Python (using Tkinter/wxPython/Qt): Introduction to GUI library. Layout management, events and bindings, fonts, colours, drawing on canvas (line, oval, rectangle, etc.) Widgets such as : frame, label, button, checkbutton, entry, listbox, message, radiobutton, text, spinbox etc.

UNIT V

10 Hrs

Database connectivity in Python: Installing mysql connector, accessing connector module, using connect, cursor, execute & close functions, reading single & multiple results of query execution, executing different types of statements, executing transactions, understanding exceptions in database connectivity.

References:

1. John V Guttag. "Introduction to Computation and Programming Using Python", Prentice Hall of India
2. Magnus Lie Hetland, Beginning Python: From Novice to Professional, Apress
3. Paul Gries , Jennifer Campbell, Jason Montojo, Practical Programming: An Introduction to Computer Science Using Python 3, Pragmatic Bookshelf, 2/E
4. A. Lukaszewski, MySQL for Python: Database Access Made Easy, Pact Publisher
5. <http://www.openbookproject.net/thinkcs/python/english2e/>

Additional Reading:



RANI CHANNAMMA UNIVERSITY, BELAGAVI

6. James Payne , Beginning Python: Using Python 2.6 and Python 3, Wiley India,
7. Python Programming, http://en.wikibooks.org/wiki/Python_Programming
8. The Python Tutorial, <http://docs.python.org/release/3.0.1/tutorial/>
9. Learn Python the Hard way, <http://learnpythonthehardway.org/>
10. Swaroop C H. A Byte of Python, <http://www.swaroopch.com/notes/python>

w.e.f. 2019-20 & onwards



RANI CHANNAMMA UNIVERSITY, BELAGAVI

17BCADNFT55: .NET Framework Using C#

Teaching Hours: 4 Hrs/week

**Marks: Main Exam: 80
IA: 20**

Preamble: C# is a very powerful language because it runs within the .NET framework and takes the best features of its predecessors C++ and Java. C# is currently the most used language in the development of third-party apps for Windows. It also is a very popular language on mobile devices, including the Windows Phone.

UNIT I

10 Hrs

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), Garbage Collection (GC), 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, Namespaces in C#, The using Keyword, The class Keyword, The Main() Method, Printing on the Console, Comments..

UNIT II

10 Hrs

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, events and delegates, operator overloading.

UNIT III

10 Hrs

Inheritance, Interface and Polymorphism: Deriving classes, Calling base class constructor, Overriding Methods, Non-Inheritable Classes, Abstract Class, Interface Inheritance, Namespace and Access Modifiers, Boxing and Un-boxing.
Working With Collections: List and Dictionary, ArrayList and HashTable, Generic Classes, Comparable and Sorting, IEnumerable and IEnumerator.

UNIT IV

10 Hrs

Exception Handling: Handling Exceptions using try and catch, Raising Exceptions using throw, Pre-defined Exception classes, Custom Exception classes, Using Statement Working with Components / Assemblies: Creating a Simple .NET Framework Component, Private and Shared Assemblies, Signing Assemblies with Strong Names, DLL Hell, Side-by-side versioning, Deploying .NET Assemblies in Global Assembly Cache.
WinForms: Introduction, Controls, Menus and Context Menus, MenuStrip, ToolStrip, Graphics and GDI, SDI and MDI Applications, Dialogbox (Modal and Modeless), Form Inheritance, Developing Custom, Composite and Extended Controls.

UNIT V

10 Hrs

Windows Services: Purpose and Advantage: Developing and Deploying, Debugging Windows Service, Sending Custom events.
Packaging and Deployment: Compiled Deployment of ASP.NET application, Types of Deployment Modules, Working with a Setup and Web Setup Project, Working with a Merge Module Project, Debugging and Tracing.

References:

1. E. Balagurusamy, Programming in C#, Tata McGraw Hill
2. Stephen Walthert, ASP.NET 3.5 unleashed, SAMS



RANI CHANNAMMA UNIVERSITY, BELAGAVI

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



17BCAAAPP56: Android and Python Lab.

Practical Hours: 4 Hrs/week

Marks: Main exam: 80

IA: 20

Android

1. Creating a "Hello, Android" Android Application (description available on the official developer website of Android: <http://developer.android.com>)
2. Creating an application with multiple activities and a simple menu using listview (<https://sites.google.com/site/hkustcomp4521/home/lab-exercises>)
3. Develop the android mobile application which consists of GUI components for Login Page creation.
4. Develop an application to illustrate using of images and colors.
5. Develop an application to illustrate WebView
6. Illustrate using of audio functions in Android
7. Write an application that draws basic graphical primitives on the screen.

Python

1. Program related to string manipulation
2. Program related to dictionaries
3. Programs using list comprehensions and anonymous functions
4. Programs using the built-in methods of the string, list and dictionary classes
5. Programs with iterables and iterators.
6. Program to demonstrate exception handling.
7. Program to demonstrate the use of regular expressions.
8. Program to show draw shapes & GUI controls.
9. Program to create server-client and exchange basic information



17BCADNLP57: C# Lab.

Practical Hours: 4 Hrs/week

**Marks: Main exam: 80
IA: 20**

1. Write a C# Sharp program to calculate root of Quadratic Equation
2. Write a program in C# Sharp to count the frequency of each element of an array
3. Write a program in C# Sharp to count a total number of alphabets, digits and special characters in a string
4. Write a program in C# Sharp to create a function to calculate the sum of the individual digits of a given number.
5. Create an application that allows the user to enter a number in the textbox named 'getnum'. Check whether the number in the textbox 'getnum' is palindrome or not. Print the message accordingly in the label control named lbldisplay when the user clicks on the button 'check'.
6. Write a C# program to balance a checkbook. The program needs to get the initial balance, the amounts of deposits, and the amounts of checks. Allow the user to process as many transactions as desired; use separate functions to handle deposits and checks
7. 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.
8. Write a program that inputs the coordinates of three mouse clicks from the user and then draws a triangle in the output window using those three points.
9. Write a program to implement multilevel inheritance.
10. C# Program to Demonstrate iList Interface.
11. Write a Program to display the selected date in the calendar.
12. Write a Program to insert the data in to database using Execute-Non Query.
13. Write a Program to display the Contact Number of an author using database
14. Create an application which will ask the user to input his name and a message, display the two items concatenated in a label, and change the format of the label using radio buttons and check boxes for selection, the user can make the label text bold ,underlined or italic and change its color . include buttons to display the message in the label, clear the text boxes and label and exit.
15. List of employees is available in listbox. Write an application to add selected or all records from listbox (assume multi-line property of textbox is true).