

**Progressive Education Society's
Modern College Of Arts, Science & Commerce (Autonomous)
Ganeshkhind, Pune – 411016**

**Three Year Degree Program in
Bachelor of Computer Applications (Science): B.C.A. (Science)
(Faculty of Science & Technology)**

S.Y.B.C.A. (Science)

**Choice Based Credit System Syllabus
To be implemented from Academic Year 2023-2024**



Title of the Course: Bachelor of Computer Applications (Science)

Preamble of the syllabus

The B.C.A. (Science) program is a combination of computer and applied courses from science stream. The computer related courses introduce techniques of programming, databases, web designing, system analysis, design tools and different computing environments. The applied courses include mathematics, statistics and electronics that provide theoretical and practical foundation for the learner.

Objectives:

- To produce knowledgeable and skilled human resources that is employable in IT and ITeS.
- To impart knowledge required for planning, designing and building Complex Application Software Systems as well as to provide support for automated systems or applications.
- It helps students analyse the requirements for system development and exposes students to business software and information systems.
- This course provides students with options to specialize in legacy application software, system software or mobile applications.
- To produce entrepreneurs

Introduction

The Program is of Three Years duration with six semesters. It is a Full Time Degree Program. The program will be based on Choice-based credit system comprising 132+8 (140) credit points.

Eligibility for Admission:

Any candidate who has passed the XII standard Examination in Science stream from, Maharashtra State Board of Secondary and Higher Secondary Education or equivalent Board of Examination, is eligible for admission to the First Year of this program.

OR

Passed Three Year Diploma Course approved by the DTE, Maharashtra State or Equivalent authority.

Lateral Entry to SYBCA

Any candidate who has passed three Year Diploma course in Computer Engineering/ Technology/ Information Technology/ Electronics Communication/ Electronics Telecommunications/ Electronics approved by the DTE, Maharashtra State or Equivalent authority is eligible for admission to direct second year (SYBCA) of this program.



Semester -III S.Y. B.C.A (Science)						
Course Code	Course Title	Credits		Evaluation		
		T	P	CIA	SE	Total
23-BCA-231	Data Structures	4	-	30	70	100
23-BCA-232	Database Management Systems - II	4	-	30	70	100
23-BCA-233	Programming using PHP	4	-	30	70	100
23-BCA-234	Data Structures Laboratory	-	2	15	35	50
23-BCA-235	Database Management Systems - II Laboratory	-	2	15	35	50
23-BCA-236	Programming using PHP Laboratory	-	2	15	35	50
	Environmental Science I	2	-	15	35	50
	Language –I	2	-	15	35	50
	Total	16	6	165	385	550

Total Credits: [16(TH) + 6 (PR)] = 22

TH: Theory **PR:** Practical **CIA:** Continuous Internal Assessment **SE:** Semester Examination



Semester -IV S.Y. B.C.A (Science)						
Course Code	Course Title	Credits		Evaluation		
		T	P	CIA	SE	Total
23-BCA-241	Core Java	4	-	30	70	100
23-BCA-242	Object Oriented Programming in Python	4	-	30	70	100
23-BCA-243	Object Oriented Software Engineering	4	-	30	70	100
23-BCA-244	Core Java Laboratory	-	2	15	35	50
23-BCA-245	Object Oriented Programming in Python Laboratory	-	2	15	35	50
23-BCA-246	Object Oriented Software Engineering Laboratory	-	2	15	35	50
	Environmental Science Awareness Course –II	2	-	15	35	50
	Language –II	2	-	15	35	50
Total		16	6	165	385	550

Total Credits: [16(TH) + 6 (PR)] = 22

TH: Theory **PR:** Practical **CIA:** Continuous Internal Assessment **SE:** Semester Examination



S.Y.B.C.A (Science)**SEMESTER III****Subject Code: 23-BCA-231****Subject Name: Data Structures****Course Objectives**

- To understand algorithms and analysis of algorithms.
- To learn static and dynamic data structures.

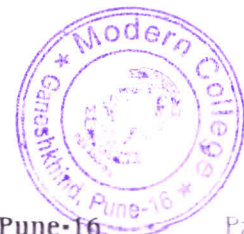
Course Outcomes

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

- Apply appropriate data structures for the given problem.
- Design an efficient algorithm for the given problem.
- Determine the time and space complexity of a given algorithm

Course Contents

Unit I	Introduction to data structure	06 Hrs
	<ol style="list-style-type: none">1. Data types and data objects2. Abstract Data Types (ADT)3. Data structure4. Algorithm analysis: Frequency counts, Space and Time complexity.5. Asymptotic notation : BigO, Omega (Ω)6. Algorithms and its complexity using simple example algorithms	
Unit II	Arrays	06 Hrs
	<ol style="list-style-type: none">1. Introduction and definition2. Matrix representation using arrays: Row and column major, operations on matrices, Sparse Matrix3. Sorting techniques with time complexity: Bubble sort, Insertion sort, Merge sort, Quick sort4. Searching techniques with time Complexity: Linear search, Binary search	
Unit III	Linked Lists	12 Hrs
	<ol style="list-style-type: none">1. Introduction and Definition2. Representation: Static & Dynamic3. Types of linked lists: singly, doubly, circular	



4. Operations on link list: create, display, insert, delete, reverse, search, sort, concatenation, Merge
5. Real world applications of Link list (implementation not expected)

Unit IV Stacks and Queues**10Hrs**

1. Representation of Stack: Using arrays and Linked Lists
2. Operations on stack: push, pop
3. Applications of Stack : Recursion, Infix to postfix, postfix to infix
4. Representation of Queues : Static & Dynamic
5. Operations on queue: insert, delete
6. Types of queue: Circular queue and Priority queue
7. Real world Applications of queue (Implementation not expected)

Unit V Trees**8 Hrs**

1. Introduction and Tree terminologies: Definitions: Tree, root, child, leaf, level,height, depth
2. Binary trees: Types: Rooted, full, complete and skewed.
3. Representation of Trees: Using arrays and Linked List
4. Types of Traversal: Preorder, Inorder, Postorder, Applications of Binary Tree-Expression Tree, Huffman Encoding
5. Binary Search Tree (BST): Introduction and definition
6. Search Trees-AVL, rotations in AVL Tree, operations on AVL Tree,
7. Introduction of B Tree, B+ Tree.

Unit VI Graphs**6 Hrs**

1. Introduction and Graph terminologies.
2. Representation of a Graph – Adjacency matrix, Adjacency list, Adjacency multi-list
3. Graph Traversals – DFS, BFS
4. Applications of graphs - Topological sort

Reference Books:

1. *Fundamentals of Data Structures* by Horowitz Sahani (Galgotia)
2. *Introduction to Data Structures using C* by Ashok Kamthane
3. *Data Structures using C* by Bandopadhyay & Dey(Pearson)
4. *Data Structures using C* by Srivastava BPB Publication



E-Books and online resources:

1. https://www.academia.edu/5900697/C_and_Data_Structures_Balaguruswamy
2. <https://www.javatpoint.com/tree>
3. <https://www.simplilearn.com/tutorials/data-structure-tutorial/graphs-in-data-structure>
4. <https://www.andrew.cmu.edu/course/15-121/lectures/Stacks%20and%20Queues/Stacks%20and%20Queues.html>



S.Y.B.C.A (Science)**SEMESTER III****Subject Code: 23-BCA-232****Subject Name: Database Management Systems - II****Course Objectives:**

- To understand advanced SQL features and procedural SQL.
- To study concurrency control and crash recovery techniques.
- To understand need of database security.
- To learn different database system architecture.

Course Outcomes:

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

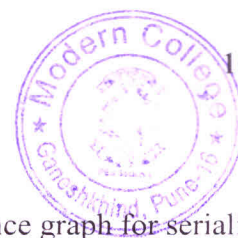
- Formulate SQL queries using advanced SQL features.
- Perform Database operations using PL/PostgreSQL.
- Compare and contrast different concurrency control and recovery techniques.
- Apply mechanisms for database security.
- Analyze various database system architectures

Course Contents**Unit-I Relational Database Design****10 Hrs**

1. PL/Postgre SQL: Language structure
2. Controlling the program flow, conditional statements, loops
3. Views
4. Functions and Procedure
5. Handling errors and exceptions
6. Cursors
7. Triggers
8. Packages

Unit-II Transaction Concepts**10 Hrs**

1. Transaction, properties of transaction, states of transactions
2. Concurrent execution of transactions and conflicting operations
3. Schedules, types of schedules, concept of serializability, precedence graph for serializability



Unit-III Concurrency Control**10 Hrs**

1. Ensuring serializability by locks, different lock modes
2. 2PL and its variations
3. Multiple Granularity locking protocol
4. Basic timestamp method for concurrency, Thomas Write Rule
5. Locks with multiple granularity, dynamic database concurrency (Phantom Problem)
6. Timestamps versus locking
7. Optimistic concurrency control algorithm, multi version concurrency control
8. Deadlock handling methods -
 - Detection and Recovery (Wait for graph).
 - Prevention algorithms (Wound-wait, Wait-die)
 - Deadlock recovery techniques (Selection of Victim, Starvation, Rollback)

Unit-IV Crash Recovery**06 Hrs**

1. Transaction Failure classification
2. Recovery concepts
3. Checkpoints
4. Recovery with concurrent transactions (Rollback, checkpoints, commit)
5. Log base recovery techniques (Deferred and Immediate update)
6. Buffer Management
7. Database backup and recovery from catastrophic failures
8. Shadow paging

Unit-V Database Security**06Hrs**

1. Introduction to database security concepts
2. Methods for database security
3. Discretionary access control method
4. Mandatory access control and role based access control for multilevel security
5. Use of views in security enforcement
6. Overview of encryption technique for security
7. Statistical database security



Unit-VI Database System Architectures**06 Hrs**

1. Centralized and Client – Server Architectures
2. Server System Architectures
3. Introduction to Parallel Systems
4. Introduction to Distributed Systems
5. Introduction to Object Based Databases

Reference Books:

1. *Database System Concepts – Abraham Silberschatz, Henry F. Korth, S. Sudarshan, 6th edition-McGraw-Hill*
2. *Fundamentals of Database Systems- Ramez Elmasri, Shamkant B. Navathe, 6th edition–Pearson.*
3. *Database Management Systems -Raghu Ramakrishnan, Johannes Gehrke, 3rd edition, TataMcGraw Hill*
4. *Introduction to Database Management System- Bipin Desai, 3rd edition, Galgotia Publication*
5. *An Introduction to Database Systems - C.J. Date, 7th edition, Addison-Wesley*
6. *Practical PostgreSQL- Joshua D. Drake, John C Worsley, O'Reilly Publications*

E-Books and online resources:

1. <https://link.springer.com/content/pdf/bfm:978-3-540-48399-1/1.pdf>
2. <https://diblokdcma.files.wordpress.com/2009/10/springer-fundamentals-of-relational-database-management-systems-apr-2007.pdf>
3. <https://industri.fatek.unpatti.ac.id/wp-content/uploads/2019/03/162-Introduction-to-Database-Management-System-Satinder-Bal-Gupta-Aditya-Mittal-Edisi-2-2017.pdf>
4. https://ebooks.lpude.in/management/mba/term_3/DCAP204_MANAGING_DATABASE_DCAP402_DATABASE_MANAGEMENT_SYSTEMS.pdf



S.Y.B.C.A (Science)

SEMESTER III

Subject Code: 23-BCA-233

Subject Name: Programming using PHP

Course Objectives:

- To learn PHP programming and database connectivity
- To understand and learn AJAX and XML

Course Outcomes:

On completion of the course, students will be able to–

- Develop web-based application using suitable client side and server side web technologies.
- Build Dynamic web site using server side PHP Programming and Database connectivity.
- Build applications using AJAX and XML

Course Contents**Unit-I Introduction to PHP**

06 Hrs

1. Introduction to php, Features of PHP, PHP Fundamentals
2. Use of PHP, Lexical structure, Language basics.
3. Basic Syntax, echo & print statement
4. Variables: Local, global & static
5. Data Types, Operators, Control Statements.

Unit-II Functions and Array

06 Hrs

1. PHP Functions
2. Parameterized Function
3. PHP Call By Value, Call By Reference
4. PHP Default Arguments, Variable Arguments
5. PHP Recursive Function, Anonymous Functions
6. PHP Array
7. Types of array: Indexed Arrays, Associative Arrays, Multidimensional Arrays
8. Traversing Array Sorting Arrays

Unit-III Introduction to OOPS

08 Hrs

1. Classes, Objects
2. Introspection, Serialization, Inheritance
3. Polymorphism, Overloading Interfaces and Abstraction,



4. Encapsulation, Constructor, Destructor

Unit-IV Introduction to Web Techniques**08 Hrs**

1. HTTP Basics, Variables, Server Information,
2. Processing Forms, File uploading
3. Setting Response Headers, Maintaining State, SSL, PHP Error handling

Unit-V Database Connectivity**07Hrs**

1. Introduction to SQL
2. Using PHP to access a database
3. Relational databases and SQL
4. PEAR DB basics
5. Advanced database techniques

Unit-VI Ajax and XML**07 Hrs**

1. Understanding java scripts for AJAX
2. AJAX web application model
3. AJAX –PHP framework
4. Performing AJAX validation
5. Handling XML data using php and AJAX
6. What is XML?
7. XML document Structure
8. PHP and XML
9. XML parser
10. The document object model
11. The simple XML extension

Unit-VII PHP Framework**06 Hrs**

1. CodeIgniter - Overview, Installing CodeIgniter
2. CodeIgniter Features
3. CodeIgniter Architecture
4. MVC Framework , Basic concept of CodeIgniter, Libraries
5. Working with databases

Reference Books:

1. *Programming PHP* By Rasmus Lerdorf and Kevin Tatroe, O'Reilly publication, ISBN-13978-1565926103



2. *Beginning PHP5*, Wrox publication
3. “*Beginning PHP5, Apache, and MySQL Web Development (Programmer to Programmer)*”, by Elizabeth Naramore, Jason Gerner, Yann LeScouarnec, Jeremy Stolz, Michael K. Glass, Wrox; 2nd edition (27 January 2005),
4. *PHP for Beginners*, SPD publication
5. *PHP web services*, Wrox publication.
6. *AJAX Black Book*, Kogent solution
7. *Mastering PHP*, BPB Publication.
8. *PHP cookbook*, O'Reilly publication

E-Books and online resources:

<https://www.invezzatechnologies.com/best-php-frameworks/>

https://www.tutorialspoint.com/cakephp/cakephp_overview.htm

https://www.tutorialspoint.com/laravel/laravel_overview.html

www.php.net

<https://www.w3schools.com/php/>

<https://www.tutorialspoint.com/php/index.htm>



S.Y.B.C.A (Science)**SEMESTER III****Subject Code: 23-BCA-234****Subject Name: Data Structure Laboratory****Course Objectives:**

- To understand algorithms and analysis of algorithms
- To learn static and dynamic data structures

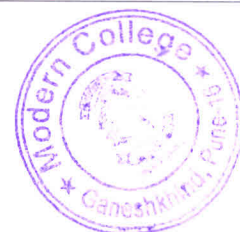
Course Outcomes:

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

- Apply appropriate data structures for the given problem.
- Design an efficient algorithm for the given problem and implement it using C Programming.
- Determine the time and space complexity of a given algorithm

Course Contents

Sr.No	Assignment
1	Non-Recursive Sorting Techniques Bubble Sort, Insertion Sort
2	Recursive Sorting Techniques : Quick Sort , Merge Sort
3	Searching Techniques :Linear search, Binary Search
4	Linked List : Linked List, Doubly Linked List Singly Circular Linked List, Doubly Circular Linked List
5	Stack: Static and Dynamic Implementation
6	Queue : Static and Dynamic Queue Implementation
7	Binary Search Tree (Dynamic
8	Graph : Adjacency Matrix Representation Adjacency List Representation In-degree and Out-degree calculation



S.Y.B.C.A (Science)

SEMESTER III

Subject Code: 23-BCA-235

Subject Name: Database Management System-II Laboratory

Course Objectives:

- To learn advanced features in SQL such as Nested queries and Views
- To become familiar with PL/Postgre SQL

Course Outcomes:

After successful completion of this course, learner will be able to

- Formulate SQL queries using advanced features
- Write stored procedures, cursors and triggers using PL/Postgre SQL.
- Design a database using database normalization technique

Sr. No.	Assignments
1	Simple and Nested Queries.
2	Views Creation.
3	Stored Functions 1) A Simple Stored Function 2) A Stored Function that returns 3) A Stored Function recursive
4	Cursors 1) Simple Cursor 2) Parameterize Cursor
5	Database using normalization
6	Error and Exception handling 1) Raise Notice 2) Raise Exception
7	Triggers 1) Before (Insert, Update, Delete) 2) After (Insert, Update, Delete)



S.Y.B.C.A (Science)

SEMESTER III

Subject Code: 23-BCA-236

Subject Name: Programming using PHP Laboratory

Course Objectives:

- To get familiar with basics of the Internet Programming.
- To acquire knowledge and skills for creation of web site using client and server side programming
- To understand process of developing responsive web applications
- To explore different web extensions and web services standards

Course Outcomes:

After successful completion of this course, learners will be able to

- Design and implement static and dynamic websites using appropriate client side and server side technologies.
- Build Dynamic web site using PHP Programming and Database connectivity.
- Build applications using AJAX and XML and web services

Sr. No.	Assignments
1	Introduction to PHP
2	PHP Functions and Array
3	Classes and Objects
4	Inheritance, Interface and Constructor
5	Cookies and Session
6	Processing Forms and File Uploading
7	Databases
8	XML
9	Ajax
10	Introduction to CodeIgniter



S.Y.B.C.A (Science)**SEMESTER IV****Subject Code: 23-BCA-241****Subject Name: Core Java****Course Objectives**

- To learn Object Oriented Programming language.
- To study various java programming concept like Interface, File and Exception Handling etc.
- To design User Interface using Swing and AWT

Course Outcomes

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

- Understand the concept of classes, object, packages and Collections.
- To develop GUI based application.

Course Contents**Unit I: Introduction to Java****10 Hrs**

1. Object Oriented Programming Concepts
2. A short history of Java Features OR Buzzwords of Java
3. Java Environment, Simple Java Program
4. Java Tools – jdb, javap, javadoc
5. Types of Comments
6. Data Types Final Variable
7. Declaring 1D, 2D Array
8. Accepting Input (Command Line Arguments, BufferedReader, Scanner)

Unit II: Objects and Classes**10 Hrs**

1. Defining your own classes
2. Access Specifiers (public, protected, private, default)
3. Array of Objects
4. Constructors, Overloading Constructors and Use of 'this' keyword static block, static fields And methods
5. Predefined Classes
Object Class, Methods (equals(), toString(), hashCode(), getClass()), String Class And StringBuffer Class, Formatting String data using format() method, Creating And Using Packages Wrapper Classes



Unit III: Inheritance and Interface**10 Hrs**

1. Inheritance Basics (extends Keyword) and Types of Inheritance
2. Superclass, Subclass and use of Super Keyword
3. Method Overriding and runtime polymorphism
4. Use of final keyword related to method and class
5. Use of abstract class and abstract methods
6. Defining and Implementing Interfaces
7. Runtime polymorphism using interface
8. Concept of Marker and Functional Interfaces

Unit IV: Exception and File Handling**08 Hrs**

1. Dealing with errors , Exception class, Checked And Unchecked Exception
2. Catching Exceptions, Multiple Catch Block, Nested try block
3. Creating User Defined Exception
4. Introduction to Files And Streams
5. Input-OutputStream : FileInputStream/OutputStream, BufferedInput/OutputStream, DataInput/OutputStream
6. Reader-Writer : FileReader/Writer, BufferedReader/Writer, InputStreamReader, OutputStreamWriter

Unit IV: User Interface with AWT and Swing**10 Hrs**

1. Introduction to AWT and Swing
2. The MVC Architecture And Swing
3. Layouts And Layout Managers
4. Containers And Components – JFrame, JButton, JLabel, JText, JTextArea, JCheckBox And JRadioButton, JList, JComboBox, JMenu And related Classes
5. Dialogs (Message, Confirmation, Input), JFileChooser, JColorChooser
6. Event Handling: Event Sources, Listeners
7. Adapters And Anonymous Inner Class

Reference Books:

1. Complete reference Java by Herbert Schildt(5th edition)
2. Java 2 programming black books, Steven Horlzner
3. Programming with Java , A primer ,Forth edition , By E. Balagurusamy
4. Core Java Volume-I-Fundamentals, Eighth Edition, Cay S. Horstmann, Gary Cornell
Prentice Hall, Sun Microsystems Pres

E-References:

Open Source Initiative: <https://opensource.org/>

Wikipedia: <https://en.wikipedia.org/>

Github: <https://help.github.com/>



S.Y.B.C.A (Science)**SEMESTER IV****Subject Code: 23-BCA-242****Subject Name: Object Oriented Programming in Python****Course Objectives**

- To learn problems solving using Python
- To understand Django frameworks.

Course Outcomes

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

- To understand basic concepts in python.
- Design an efficient algorithm for the given problem.
- Determine the time and space complexity of a given algorithm

Course Contents**Unit-I Introduction to Python****08 Hr**

1. Python Features, Application, Characteristics
2. Installing Python, Basic Example
3. Variables, Data Types, Keywords, Literals
4. Operators, Comments
5. If else Loops Break Continue Pass

Unit-II List, Tuples and Set**06 Hrs**

1. List Declaration, Characteristics,
2. List Indexing and Splitting, Updating List, Operations on List.
3. Iterating List, Adding and Removing Elements to List, Built-in functions.
4. Features of Tuples, Accessing Tuple Elements, Slicing, Deleting a Tuple.
5. Tuple Methods, Iterating through Tuple, Advantages of Tuple over List.
6. Creating set, Adding and Removing Elements from set, Set Operations,
7. Set Comparison, Frozen Set

Unit-III String and Dictionary**06 Hrs**

1. Creating String in Python, Strings indexing and splitting
2. Reassigning Strings, Deleting the String, String Operators
3. String Formatting, The format() method, String functions



4. Creating the Dictionary, Accessing the dictionary values, Adding Dictionary Values
5. Deleting Elements, Iterating Dictionary, Properties of Dictionary Keys
6. Built-in Dictionary Functions, Built-in Dictionary methods

Unit-IV File I/O and Directory**06 Hrs**

1. File Handling, Opening a file, The close() method, The with statement, Writing the file
2. Read file through for loop, Read Lines of the file, Creating a new file, File Pointer positions
3. Modifying file pointer position, Python OS module, Creating the new directory
4. Changing the current working directory, Deleting directory
5. Writing Python output to the files, File methods

Unit-IV Modules, Functions and Exception handling**06 Hrs**

1. What is Modular Programming? Python import Statement
2. Locating Path of Modules
3. The dir() Built-in Function, Namespaces and Scoping
4. What is an Exception? Try and Except Statement - Catching Exceptions
5. How to Raise an Exception, Assertions in Python, Try with Else Clause
6. Finally Keyword in Python, User-Defined Exceptions, Exceptions List

Unit-V Introduction to Object Oriented Concepts**08 Hrs**

1. What is OOPs, OOP Concepts
2. Class, Object, Method,
3. Inheritance, Polymorphism
4. Data Abstraction, Encapsulation
5. Object-oriented vs. Procedure-oriented Programming

Unit-VI Introduction to GUI Programming – Tkinter**08 Hrs**

1. What is Tkinter?, Tkinter widgets, Tkinter Geometry
2. Tkinter Button, Canvas, Checkbutton, Entry
3. Tkinter Frame, Label, Listbox, Menubutton, Menu, Message, Radiobutton
4. Tkinter Scale, Scrollbar, Text, Toplevel, Spinbox
5. Tkinter PanedWindow, LabelFrame, messagebox



Reference Books:

1. Introducing Python- Modern Computing in Simple Packages – Bill Lubanovic, O.,Reilly Publication
2. Django for Beginners : Build websites with Python and Django Kindle Edition by William S Vincent
3. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress
4. Introduction to Computer Science Using Python- Charles Dierbach, Wiley Publication
Learning with Python “, Green Tea Press, 2002
5. A Beginner’s Python Tutorial: http://en.wikibooks.org/wiki/A_Beginner%27s_Python

E-Books

<http://spoken-tutorial.org>

<https://starcertification.org/Certifications/Certificate/python>



S.Y.B.C.A (Science)**SEMESTER IV****Subject Code: 23-BCA-243****Subject Name: Object Oriented Software Engineering****Course Objectives:**

- To study fundamental concepts in software testing, including software testing objectives, process, criteria, strategies, and methods.
- To discuss various software testing issues and solutions in software unit test; integration, regression, and system testing.
- To learn how to planning a test project, design test cases and data, conduct testing operations, manage software problems and defects, generate a testing report.
- To expose the advanced software testing topics, such as object-oriented software testing methods, and component-based software testing issues, challenges, and solutions.
- To gain software testing experience by applying software testing knowledge and methods to practice-oriented software testing projects.

Course Outcomes:

After completion of the course, a student will be able to:

- Have an ability to apply software testing knowledge and engineering methods.
- Have an ability to design and conduct a software test process for a software testing project.
- Have an ability to identify the needs of software test automation, and define and develop a test tool to support test automation.
- Have an ability understand and identify various software testing problems, and solve these problems by designing and selecting software test models, criteria, strategies, and methods.

Course Contents:**Unit - 1: Introduction to Object Oriented Concepts****04 Hrs****1.1 Introduction to Software Engineering**

1.1.1 Program vs. Software

1.1.2 Characteristics of software

1.2 Introduction to object orientation

1.2.1. Classes and objects

1.2.2 Messages, Attributes and Methods



1.2.3 Encapsulation, Inheritance, Polymorphism

1.3. Object Oriented Methodology

1.3.1 Coad and Yourdon Methodology

1.3.2. Booch Methodology

1.3.3 Rumbaugh Methodology

1.3.4 Jacobson Methodology

1.4 Object Oriented Modeling

Unit – 2: Modeling and UML

12 Hrs

2.1 Importance of Modeling

2.2 Principles of Modeling, Object Oriented Modeling

2.3 Overview of UML

2.4 Conceptual Model of UML and Architecture

2.5 Structural Modeling and Diagrams

2.5.1 Classes and advanced classes, Relationships and Advanced Relationships

2.5.2 Interface, Types, Roles, Packages, Common Mechanisms

2.5.3 Class Diagram, Instances, Object Diagram

2.6 Behavioural Modeling and Diagrams

2.6.1 Use Case Diagram, Interaction Diagram

2.6.2 Activity Diagram, State Machine and State Chart Diagram

2.6.3 Events and Signals, Processes and Threads. Time and Space

2.7 Architectural Modeling

Unit – 3: Software Development Life Cycle (SDLC)

06 Hrs

3.1 Conventional Software Life Cycle Methods

3.1.1. Build-and-fix model

3.1.2. Waterfall Model

3.1.3. Prototyping Model

3.1.4. Iterative Enhancement Model

3.1.5 Spiral Model

3.2 Object Oriented Software Life Cycle Models

3.2.1 Fountain Model

3.2.2 Rational Unified Process



Unit – 4: Software Requirement Elicitations and Analysis

04 Hrs

4.1 Software Requirement

- 4.1.1 Identification of Stakeholders
- 4.1.2 Functional and Non-functional requirements
- 4.1.3 Known and Unknown requirement

4.2 Requirements Elicitation Techniques

- 4.2.1 Interviews
- 4.2.2 Brainstorming Sessions
- 4.2.3 Facilitated Application Techniques
- 4.2.4 Prototyping

4.3 Software Requirements Specification (SRS) Documents

- 4.3.1 Nature of the SRS documents
- 4.3.2 Organization of the SRS Documents.

Unit – 5: Software Quality and Metrics

06 Hrs

5.1 Software Quality

- 5.1.1 Software Quality Attributes
- 5.1.2 Elements of Quality system

5.2 Software Quality Models

- 5.2.1 McCall’s software quality model
- 5.2.2 Boehm’s software quality model
- 5.2.3 ISO 9000 and ISO 9126
- 5.2.4 Capability Maturity Model (CMM)

5.3 Measurements Basics

- 5.3.1 Software Metrics
- 5.3.2 Application Areas of Metrics
- 5.3.3 Categories of Metrics
- 5.3.4 Object Oriented Metrics – Coupling Metrics, Inheritance Metrics, Reuse Metrics, Size Metrics

Unit – 6: Software Testing Strategies and Techniques

04 Hrs

6.1. Software Testing

- 6.1.1 Testing Objectives
- 6.1.2 Principles of Testing
- 6.1.3 Verification and Validation



6.2 Testing Techniques

- 6.2.1 White Box Testing – Basis Path Testing, Control Structure Testing
- 6.2.2 Black Box Testing – Boundary Value Analysis, Equivalence Partitioning
- 6.2.3 Difference between White Box and Black Box Testing

Unit – 7: Level of Testing**04 Hrs**

- 7.1 Strategic Approach for Software Testing
- 7.2 Unit Testing
- 7.3 Integration Testing – Top down, Bottom-up Integration
- 7.4 System Testing – Acceptance, Performance, Regression, Load / Stress testing, Security Testing, Internationalization Testing
- 7.5 Alpha and Beta Testing
- 7.6 Agile Testing

Unit – 8: Introductions to Test Case Design**04 Hrs**

- 8.1 How to identify Error, Bug
- 8.2 Design Entry and Exit criteria for Test Case, Design Test Cases in Excel
- 8.3 Test Case and Test Plan, Test Cases for Simple Program
- 8.4 Defect Report

Unit – 9: Introduction to Automation Testing & Selenium**04 Hrs**

- 9.1 Automation Testing
- 9.2 Advantages and Disadvantages of Automation Testing
- 9.3 Introduction of Selenium, History of Selenium
- 9.4 Selenium Tool
- 9.5 Different Components in Selenium
- 9.6 Major Components in Selenium like Selenium IDE, Selenium RC, Selenium Web driver and Selenium Grid.

Reference Books:

1. Ian Sommerville, "Software Engineering," 7th Edition, Pearson Education Asia, 2007.
2. Shari Lwarence Pfleeger, Joanne M. Atlee, "Software Engineering Theory and Practice," 3rd Edition, Pearson Education, 2006.
3. Roger S. Pressman, "Software Engineering - A Practitioner's Approach," 6th Edition, McGraw-Hill, International Edition, 2001.



4. M.A. Parthasarathy, "Practical Software Estimation," 1st Edition, Pearson Education, 2007.

E-Books

https://ebooks.lpude.in/computer_application/mca/term_3/DCAP503_SOFTWARE_TESTING_AND_QUALITY_ASSURANCE.pdf

<https://www.gcreddy.com/2021/07/software-testing-study-material.html>



S.Y.B.C.A (Science)**SEMESTER IV****Subject Code: 23-BCA-244****Subject Name: Core Java Laboratory****Course Objectives:**

- Bringing uniformity in the way course is conducted across different colleges.
- Continuous assessment of the students.

Course Outcomes:

- Use an integrated development environment to write, compile, run, and test simple object-oriented Java programs.
- Read and make elementary modifications to Java programs that solve real-world problems.
- Validate input in a Java program.

Sr. No.	Assignment
1.	Java Tools and IDE, Simple java programs.
2.	Classes an Objects
3	Array of Objects and Packages
4	Inheritance and Interfaces
5	Exception And File Handling
6	GUI Designing, Event Handling



S.Y.B.C.A (Science)

SEMESTER IV

Subject Code: 23-BCA-245 Subject Name: Object Oriented Programming in Python Laboratory**Course Objectives:**

- To understand programming constructs in Python.
- To learn problems solving using Python

Course Outcomes:

After successful completion of this course, learner will be able to-

- Write programs using Python programming constructs.
- Develop applications using Python programming.

Sr. No.	Assignment
1	Basics of Python
2	Python List, Tuple and Set
3	Python Dictionary and String
4	Python File and Directories
5	Python Modules and Functions
6	Classes and Object
7	Inheritance and Polymorphism
8	Introduction to GUI Tkinter



S.Y.B.C.A (Science)**SEMESTER IV****Subject Code: 23-BCA-243 Subject Name: Object Oriented Software Engineering Laboratory****Course Objectives:**

- To have hands on experience in developing a software project by using various software engineering principles and methods in each of the phases of software development.

Course Outcomes:

- Ability to translate end-user requirements into system and software requirements
- Ability to generate a high-level design of the system from the software requirements
- Will have experience and/or awareness of testing problems and will be able to develop a SRS report.

Sr. No.	Assignment
1.	Assignments on UML Diagrams
2.	Assignments on Test Cases and Test Plan
3	Assignments on Test Strategies
4	Assignments of Test Methodologies
5	Assignments on Defect Managements.
6	Assignments on Automation Testing using Selenium.
7	Mini Project

