Punyashlok Ahilyadevi Holkar Solapur University, Solapur

NAAC Accredited-2022
'B++' Grade (CGPA 2.96)

Name of the Faculty: Science & Technology

NEP 2020

Syllabus: B.C.A.

Name of the Course: B.C.A. II (Sem.– III & IV)

(Syllabus to be implemented w.e.f. June 2025)

BCA Part-II Sem.-III										
Subject / Core Course	**Name and Type of the Paper**			**Hrs. / Week**			**Total Marks**	**UA**	**CA**	**Credits**
	Type	**Code**	**Name**	**L**	**T**	**P**				
Major	DSC1-3		Core Java	2	0	0	50	30	20	2
	Practical		Practical based on DSC1-3	0	0	2	25	15	10	1
	DSC1-4		Data Structures Using C-I	2	0	0	50	30	20	2
	Practical		Practical based on DSC1-4	0	0	2	25	15	10	1
Minor	DSC2-3		Database Management Systems (DBMS)	2	0	0	50	30	20	2
	Practical		Practical based on DSC2-3	0	0	2	25	15	10	1
	DSC2-4		Advanced Python Programming	2	0	0	50	30	20	2
	Practical		Practical based on DSC2-4	0	0	2	25	15	10	1
Generic / Open Elective	GE3 / OE3		Operating Systems	2	0	0	50	30	20	2
Vocational and Skill Enhancement Courses	VSC1		Based on DSC1-3 and DSC1-4	0	0	4	50	30	20	2
	VSC2		Based on DSC2-3 and DSC2-4	0	0	4	50	30	20	2
Ability Enhancement Course (AEC)	L2-1			2	0	0	50	30	20	2

Field Project/ RP/CC/Internship/ Apprenticeship/ Community Engagement & Services	CC2			2	0	0	50	30	20	2
Total				**14**	**0**	**16**	**550**	**330**	**220**	**22**

BCA Part-II Sem.-IV

Subject / Core Course	**Name and Type of the Paper**			**Hrs. / Week**			**Total Marks**	**UA**	**CA**	**Credits**
	Type	**Code**	**Name**	**L**	**T**	**P**				
Major	DSC1-5		Advanced Java	2	0	0	50	30	20	2
	Practical		Practical based on DSC1-5	0	0	2	25	15	10	1
	DSC1-6		Data Structures Using C-II	2	0	0	50	30	20	2
	Practical		Practical based on DSC1-6	0	0	2	25	15	10	1
Minor	DSC2-5		Relational Database Management System (RDBMS)	2	0	0	50	30	20	2
	Practical		Practical based on DSC2-5	0	0	2	25	15	10	1
	DSC2-6		Software Testing and Quality Assurance	2	0	0	50	30	20	2
	Practical		Practical based on DSC2-6	0	0	2	25	15	10	1
Generic / Open Elective	GE4 / OE4		Digital Marketing	2	0	0	50	30	20	2
Vocational and Skill Enhancement Courses	VSC3		Based on DSC1-5 and DSC1-6	0	0	4	50	30	20	2
	VSC4		Based on DSC2-5 and DSC2-6	0	0	4	50	30	20	2

Ability Enhancement Course (AEC)	L2-2			2	0	0	50	30	20	2
Field Project/ RP/CC/Internship/ Apprenticeship/ Community Engagement and Services	FP1 / CEP1			0	0	4	50	30	20	2
Total				**12**	**0**	**20**	**550**	**330**	**220**	**22**

Abbreviations:				
L: Lectures	**T**: Tutorials	**P**: Practical	**UA**: University Assessment	**CA**: College Assessment

Generic/ Open Electives: **GE/OE**	Skill Enhancement Courses: **SEC**
Indian Knowledge System: **IKS**	Ability Enhancement Courses: **AES**
Value Education Courses: **VEC**	Vocational Skill and Skill Enhancement Courses: **VSEC**
Co-curricular Courses: **CC**	Field Project/ RP/CC/Internship/ Apprenticeship/ Community Engagement and Services: **FP / CEP**

Type: DSC1-3 (Major) **BCA-II (Semester III)** **Course Title: Core Java** **(Paper Code:)**	
Credits: Theory – (2)	**Practical's – (1)**
Total Lectures: 30 Hrs.	**Contact Hrs. (L) :** 2
University Evaluation: 30 Marks	**Internal Evaluation:** 20 Marks

Course Outcomes:

1.	Knowledge of the structure and model of the Java programming language.
2.	Use the Java programming language for various programming technologies.
3.	Develop software in the Java programming language.
4.	Evaluate user requirements for software functionality required to decide whether the Java programming language can meet user requirements.

Unit	Content	Lect.
1	**Introduction to java programming:** Features of Java, JVM, parts of java, steps to java programming, API Document, starting java programming, importing classes, Formatting the output, naming conventions , Data types, Operators, control statements, switch statement, break statement, continue statement, and return statement, Input/output-accepting input from the keyboard and display output with formatting. **Arrays:** Definition, types of array- single- dimensional, multi-dimensional and jagged array, array length property, array by command line argument. **String**: Concepts, String class methods, String comparison, immutability of string, Difference between mutable and immutable object String buffer and Builder: Creating string buffer objects, String buffer class methods, String builder class, String builder class methods, String vs StringBuffer, StringBuffer vs StringBuilder. **OOPs Concept in java:** Introduction to OOP′s, features of OOP′s, access specifiers, constructor, Class and object: Object creation, initializing instance variables, Methods in java-	**15**

	instance method ,static method, ′this ′ keyword, passing primitive data types to method, passing objects to method, passing arrays to method **Inheritance and polymorphism:** Inheritance Introduction, use of inheritance , types of inheritance, ′super ′ keyword, use of protected access specifier, **Polymorphism-** Introduction, static and dynamic Polymorphism. Method overriding, method overload vs method overriding., use of ′final′ keyword, Abstract method and class	
2	**Interface-** Introduction, multiple inheritance using interfaces, Abstract class vs. interfaces **Packages:** Introduction to package ,types of packages-Built in and user defined package, creating and importing package, relating sub package in package, interfaces in package, access specifier in package, use math package. **Java I/O and stream:** Streams, OutputStream vs InputStream, OutputStream class, InputStream class, Hierarchy of OutputStream and InputStream class, Java FileWriter class, Java FileReader class, file class methods, creating file, reading file, file copy, serialization and de- serialization in file. **Exception handling:** Concept and use, Exception Handling classes, Try-catch block, Multiple Catch Block, Nested try, Finally Block, Throw Keyword Throws Keyword, Throw vs Throws ,Final vs Finally, Custom Exceptions	**15**

Reference Books:

1.	Core Java by Dr.R. Nageshwar Rao	
2.	"Programming with Java" by E Balaguruswamy.	
3.	Horstmann, "Computing Concepts with Java 2 Essentials", John Wiley.	
4.	Decker & Hirshfield, "Programming Java", Vikas Publication	
5	"Java-2 the complete Reference" by Patrick Naughton and HerbertzSchidt	

Type: Practical(Major) BCA-II (Semester III) Course Title: Practical based on DSC1-3 (Paper Code:)	
Total Lectures: 30 Hrs.	**Contact Hrs. (P) :** 2
University Evaluation: 15 Marks	**Internal Evaluation:** 10 Marks

1.	To find the sum of any number of integers entered as command line arguments.
2.	Write a program that demonstrates the use of the `break` and `continue` statements within loops.
3.	Write a program to find the maximum and minimum elements in an array.
4.	To learn the use of length in the case of a two-dimensional array
5.	Write a Java program that copies elements from one array to another.
6.	Implement a program that reverses a given string using a `StringBuffer`.
7.	Write a Java program that creates a class, initializes instance variables using constructors, and displays the values.
8.	Write a program that shows the working of different functions of String and String Buffer classes like setCharAt(), setLength(), append(), insert(), concat(), and equals().
9.	Write a program to create a distance class with methods where distance is computed in terms of feet and inches.
10.	Modify the distance class by creating a constructor for assigning values (feet and inches) to the distance object. Create another object and assign the second object as a reference variable to another object reference variable. Further, create a third object, which is a clone of the first object.

Type: DSC1-4 (Major)

BCA-II (Semester III)

Course Title: Data structures using 'C'- I

(Paper Code:)

Credits: Theory – (2)	**Practical's – (1)**
Total Lectures: 30 Hrs.	**Contact Hrs. (L) :** 2
University Evaluation: 30 Marks	**Internal Evaluation:** 20 Marks

Course Outcomes:

1	Understand the Fundamental Concepts of Data Structures..
2	Apply Stacks in Problem Solving
3	Implement Queues in Real-World Applications
4	Analyze and Implement Linked List Operations

Unit	Content	Lect.
1	**An Introduction to Data Structures:** Introduction, Definition and types of Data structure. Abstract Data Type (ADT)- ADT for array, ADT for stack, ADT for queue. Algorithm: Definition, characteristics of algorithm. **Complexity of algorithm**- Space complexity, time complexity, Big-O Notation **Array:** Introduction to Array, types of array- one dimensional, two dimensional and multidimensional, Operations of array. **Stack:** Introduction to Stack, Operations of stack- Create, isempty, isfull, push, pop, display, Implementation of stack using array (Static Implementation) Applications of Stack-Conversion of infix expression to postfix expression, Conversion of infix expression to prefix expression, Matching parenthesis in an expression (Checking expression is valid or invalid), Evaluation of postfix expression, Stack in recursion, Implementation of applications of stack. **Queue:** Introduction to Queue, Operations of queue- Create, isempty, isfull, insert, remove, display , Types of Queue- Linear Queue, Circular Queue, Deque (Double Ended	**15**

	Queue), Priority queue. Implementation of all types of queue using array (Static Implementation), Difference between stack and queue, Applications of Queue	
2	**Linked Lists:** Introduction to Linked Lists, Difference between Array and linked list. Types of linked list- 1) Linear linked list- Singly (Single) linear linked list and Doubly (Double) linear linked list. 2) Circular linked list- Singly (Single) circular linked list and Doubly (Double) circular linked list, Operations of linked list- Creation, Insertion, Deletion, Traversing, Searching, Display, count, reverse, Implementation of all types of linked list, Implementation of stack using linked list,Implementation of queue using linked list, Applications of linked list	**15**
Reference Books:		
1.	Tanenbaum: Data structures using C and C++	
2.	Data Structures Through C in Depth- S.K.Srivastava, D.Srivastava	
3.	Fundamentals of Data Structures in C by Sahni	

Type: Practical(Major) **BCA-II (Semester III)** **Course Title: Practical based on DSC1-4** **(Paper Code:)**	
Total Lectures: 30 Hrs.	**Contact Hrs.(P) :** 2
University Evaluation: 15 Marks	**Internal Evaluation:** 10 Marks
1.	Write a program to implement stack by using array. (Static Implementation of stack)
2.	Write a program, which reverses the string by using stack.
3.	Write a program to check the validity of an expression using stack.
4.	Write a program to evaluate the expression using stack.
5.	Write a program to implement linear queue by using array. (Static Implementation of queue)
6.	Write a program to implement circular queue.
7.	Write a program to implement singly linear linked list with its basic operations.
8.	Write a program to implement singly circular linked list with its basic operations.
9.	Write a program to implement stack by using linked list.
10.	Write a program to implement dubly linked list.

Type: DSC2-3 (Minor)

BCA-II (Semester III)

Course Title: Database Management System

(Paper Code:)

Credits: Theory – (2)	**Practical's – (1)**
Total Lectures: 30 Hrs.	**Contact Hrs. (L) :** 2
University Evaluation: 30 Marks	**Internal Evaluation:** 20 Marks

Course Outcomes:

1.	Gain knowledge of database systems and database management systems software.
2.	Ability to model data in applications using conceptual modeling tools such as ER Diagrams and design database schema based on the model.
3.	Formulate, using SQL, solutions to a broad range of query and data update problems.
4.	Demonstrate an understanding of normalization theory and apply such knowledge to the normalization of a database.
5.	Be acquainted with the basics of transaction processing and concurrency control.
6.	Familiarity with database storage structures and access techniques.
7.	Analyse strengths and weaknesses of the applications of database technologies to various subject areas.

Unit	Content	Lect.
1	**Introduction to database management system:** Definition, limitations of the traditional file system, advantages of DBMS, components of DBMS, database users, database structure database architecture- 2-tier and 3-level (schema) tier architecture, instances, and schema, database languages, data independence, types of data models(hierarchical, network, relational, hybrid). **Conceptual design:** ER-model: entities, attributes and its types, relationship, relationship set, generalization, specialization, aggregation, Types of Dependencies, Normalization(1NF, 2NF, 3NF, BCNF, 4NF, 5NF), introduction and features of RDBMS, the difference between DBMS and RDBMS, Codd's 12-rules.	**15**

	Relational algebra operations: select, project, Cartesian product, union, set difference. **DDL commands:** create, alter, rename, truncate, drop. **DML Commands**-insert, update, delete, select statements using where clause. **DCL Commands-** grant, revoke, user creation: creating users granting and revoking permissions on database objects, rollback, commit and savepoint datatypes **Operators:** comparison, conditional, arithmetic, logical, set and special operators – in, not in, between, not between, like, not like, is null, is not null. **Built-in functions:** arithmetic, string, date and time, conversion, aggregate and general. **Clauses:** order by, group by, having clause. **Integrity constraints:** importance of data integrity, not null, unique, foreign key constraint, on delete cascade, check, default constraints.	
2	**Sub queries:** purpose and usage of a subquery, type of subqueries- single row, multiple rows, multiple columns, applying group functions in subqueries, in, any, some, all operators in sub queries. correlated sub-queries, handling data retrieval with exists and not exists operators. **Joins:** inner join, outer joins, Cartesian, self-join and lossless join. **Sequence:** creating, retrieving data, modifying, dropping sequences, synonyms. **Index-** Definition of index, advantages of indexing, types of index, creating index, retrieving data using index. **Pseudo columns:** Types of pseudo columns, currval and nextval, level, rowid, rownum. **Views:** types of views, relational views, object views, using views for DML operations, putting check constraints upon views, creation of read-only views, materialized views.	**15**

Reference Books:

1.	Database System Concepts by Korth Silberschetz
2.	Fundamentals of Database Systems by Elmsari, Navathe
3.	SQL, PL/SQL – The Programming Language of Oracle by Ivan Bayross
4.	Database Management System by Seema Kedar

Type: Practical(Minor)

B.C.A-II (Semester III)

Course Title: Practical based on DSC2-3 (Paper Code:)

Total Lectures: 30 Hrs. **Contact Hrs. (P) :** 2

University Evaluation: 15 Marks **Internal Evaluation:** 10 Marks

Database Management System:

A) salesman

Field Name	Datatype	Size	Description
sno	Varchar2	6	Primary Key
sname	Varchar2	20	Account Holders name
address	Varchar2	30	Residential address
address1	Varchar2	30	Permanent address
pincode	number	6	Pin code
dob	Date		Date of birth
state	char	20	State
doj	date		Date of Joining Must greater than Date of birth
department	Char	20	department name
salary	number	9,2	Salary of salesman

1.	Display List of all Salesmen.
2.	Display List of all Salesmen who have salary greater than 5000.
3.	Display List of all Salesmen by salary.
4.	Display List of all Salesmen whose address located in pune.
5.	Change the salary of Salesmen whose sno is 's00009' by 10000.
6.	Modify the name Raj to Rajkumar.
7.	Delete the records whose state is Andhrapradesh.

8.	Delete the salary that have salary small than 1000.
9.	Add the column mobile_no to salesmen.
10.	Change the size of salesmen s_name column by 30.
11.	List of all salesmen whoes address start with 'N'.
12.	List of salesmen whose Department is not Computer and HR.
13.	List of salesmen whose Mobile_no is 9423376798 or 9763658485.
14.	List of salesmen whose is join in month 'june'.
15.	List of salesmen whose salary is > 5000 and < 10000.
16.	Change the column name address1 t0 permanent address.
17.	List of salesmen whose salary in order of Highest and lowest.
18.	Display average salary of salesmen.
19.	Display List of salesmen whose salary is maximum,minimum and average salary by department wise.
20.	Display how many salesmen are working on computer department.
21.	Allocate 25 charters for name if remaining charters than charter are Right padded with ' * ' .
22.	Display the list of salesmen whose middle is sanjay;
23.	Display average salary of salesmen by computer department.
24.	Display list of salesmen who joined in 12-dec-2009 .
25.	How many salesmen working whose name is 'Dipak'.
26.	Display the state which contain 6 charter.
27.	Display all the salesmen whose salary is either 5000 or 10000.
28.	Display all salesmen whose s_name contain third letter ' n '.
29.	Delete the record whose pincode is NULL values.
30.	Remove all the constraints from salesmen.
31.	Change the table name salesmen to Salesmen_master.
32.	Display Total Amount of salary of salesmen.

33.	Display the list of salesmen with exprience year;
34.	Display the User name in which salesmen_master is created.
35.	Find out the Total salary of salesmen who have joined in month 'March'.
36.	Rename the column pincode to ZIP.

Type: DSC 2-4 (Minor)

BCA-II (Semester III)

Course Title: Advanced Python Programming

(Paper Code:)

Credits: Theory – (2) Practical – (1)

Total Lectures: 30 Hrs. **Contact Hrs. (L) :** 2

University Evaluation: 30 Marks **Internal Evaluation:** 20 Marks

Course Outcomes:

1.	To know advanced conditional and looping statements in Python.
2.	To understand the Regular Expression for pattern matching.
3.	To understand the parsing of JSON, CSV, Excel files.

Unit	Content	Lect.
1	**Advanced conditional and looping statements-** List comprehension, Dictionary comprehension, Use of global, local and nonlocal in functions, Lambda/ Anonymous functions, Higher order functions, Type hinting, filter (), map (), reduce () functions, Recursive functions, Iterables, Iterators and Generators, itertools functions- Infinite iterators- count(), cycle(), repeat(), Iterators terminating on the shortest input sequence- accumulate(), batched(), chain(), chain.from_iterable(), compress(), dropwhile(), filterfalse(), groupby(), islice(), pairwise(), starmap(), takewhile(), tee(), zip_longest() , Combinatoric iterators-product(), permutations(), combinations(), combinations_with_replacement(), Closures and Decorators, @property decorator, Magic methods, *args and **kwargs, Memory management, Garbage collection, Shallow copy Vs Deep copy. **Regular expression-** Introduction to regular expression, Advantages and Operations, Specify pattern using RegEx-Meta characters- Square brackets [], Period '.', Dollar '$,' Star '*', Plus '+', Question Mark '?', Curly braces '{}', Alternation '|', Group '()', Backslash '\', Sequence characters in RegEx- '\A', '\b', '\B', '\d', '\s', '\S', '\w', '\W', '\Z'. functions and constants in re module- findall(), split(), sub(), subn(), search(). Match object- match.group(), match.start(), match.end() and match.span(), match.re and match.string, match…case Statement, Raw string using r prefix.	**18**

2	**Advanced file handling-** Use of with statement, Serialization and Deserialization, **working with json file-** json file, json module, convert dict to json string, convert dict to json file, convert json string to dict, convert json file to dict. **Working with CSV Files-** CSV File, csv module, reading csv file with csv, Reading CSV files into a dictionary with csv, Writing CSV files with csv, Writing CSV file from a dictionary with csv. **Working with excel file-** Basic excel terminology, installing openpyxl, Reading excel spreadsheets, Creating a Simple Spreadsheet, Basic Spreadsheet Operations- Adding and Updating Cell Values, Managing Rows and Columns- insert row, delete row, insert column, delete column. Managing Sheets, Freezing Rows and Columns, Adding Filters, Adding Formulas, Adding Styles, Conditional Formatting, Adding Images- Install pillow module, add image into spreadsheet. Adding Charts like Barchart, Linechart, Pie-chart into spreadsheet.	**12**

Reference Books:

1.	Programming through Python, M. T. Savaliya, R. K. Maurya, G. M. Magar, STAREDU Solutions
2.	Introducing Data Science: Big Data, Machine Learning, and More, Using Python Tools, Davy Cielen, Arno D.B. Meysman, et al.
3.	Python Programming: An Introduction To Computer Science, Fourth Edition Grayscale Indian Edition, Zohn Zelle

Type: Practical(Minor) **BCA-II (Semester III)** **Course Title: Practical based on DSC2-4** **(Paper Code:)**	
Total Lectures: 30 Hrs.	**Contact Hrs. (P) :** 2
University Evaluation: 15 Marks	**Internal Evaluation:** 10 Marks
1.	Write a program to demonstrate list comprehension.
2.	Write a program to demonstrate dictionary comprehension.
3.	Write a program to demonstrate global, local, nonlocal keywords.
4.	Write a program to demonstrate anonymous functions.
5.	Write a program to demonstrate filter, map, reduce functions.
6.	Write a program to demonstrate any five itertools functions.
7.	Write a program to demonstrate decorators.
8.	Write a program to demonstrate findall, sub, split, search functions in regular expression.
9.	Write a program to demonstrate match…case statement.
10.	Write a program to convert dict to JSON string.

Type: GE3 / OE3 (Generic / Open Elective)

B.Sc(ECS)-II (Semester III)

Course Title: Operating Systems

(Paper Code:)

Credits: Theory – (2)

Total Lectures: 30 Hrs. **Contact Hrs. (L) :** 2

University Evaluation: 30 Marks **Internal Evaluation:** 20 Marks

Course Outcomes:

1.	To understand the main components of an OS and their functions.
2.	To describe the functions of a modern OS concerning convenience, efficiency, and the ability to evolve.
3.	To make aware of different types of OS and their services.
4.	To learn different process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.

Unit	Content	Lect.
1	Operating system: Definition of operating system, Types of Operating Systems-Batch, Multiprogramming, Time Sharing, Real-Time, Distributed, Parallel, OS Services, System components, and System Calls. **Process Management:** Concept of Process, Process states, Process Control Block, Context switching, Operations on Process.	**10**
2	**Process Synchronization and Deadlocks:** **Scheduling-** Concept of Process Scheduling, Types of Schedulers, Scheduling criteria, Scheduling algorithms Preemptive and Non-preemptive, FCFS, SJF, Round Robin, Priority Scheduling, Multilevel Queue Scheduling, Multilevel- feedback Queue Scheduling. **Process Synchronization:** The Producer Consumer Problem, Race Conditions, Critical Section Problem, Semaphores, and Classical Problems of Synchronization: Reader-Writer Problem, Dinning Philosopher Problem.	**20**

Reference Books:	
1.	Operating System Concepts By Siberchatz and Galvin.
2.	Modern O.S. By Andrews Tanenbaum.
3.	Operating System Concepts by Abraham Silberschatz, Peter B. Galvin, Greg Gagne.

Type: VSC1 (Vocational and Skill Enhancement Courses)

BCA-II (Semester III)

Course Title: Based on DSC1-3 and DSC1-4 **(Paper Code:)**

Total Lectures: 60 Hrs. **Contact Hrs. (P) :** 4

University Evaluation: 30 Marks **Internal Evaluation:** 20 Marks

Tree:	
1.	Write a program that sorts an array in ascending and descending order .
2.	Implement matrix addition and multiplication using multi-dimensional arrays.
3.	Write a Java program to demonstrate the length property of arrays.
4.	Write a program that demonstrates the immutability of strings in Java.
5.	Develop a Java program that checks whether a given string is a palindrome.
6.	Write a program that replaces all occurrences of a specific word in a string with another word.
7.	Implement a Java program that demonstrates method overloading.
8.	Write a program that illustrates method overriding.
9.	Write a program that demonstrates the use of this keyword.
10.	Write a program that initializes an object using a parameterized constructor.
11.	Write a Java program that uses static variables and static methods.
12.	Develop a program that demonstrates multiple levels of inheritance in Java.
13.	Implement a program that uses an abstract class and an interface together in Java.
14.	Create an abstract class Shape with an abstract method calculateArea()
15.	Create an interface Vehicle and extend it into Car and Bike interfaces.
Array:	
1.	Write a program that finds only even elements in an array.
2.	Write a program that finds only odd elements in an array.
3.	Write a program that finds the addition of matrices.
4.	Write a program that finds multiplication of matrices.
Stack:	
1.	Write a program to check entered string is palindrome or not by using stack.

2.	Write a program to convert decimal number into binary number by using stack.
3.	Write a program to count total number of vowels present in string by using stack.
4.	Write a program which convert infix expression into prefix expression.
5.	Write a program which convert infix expression into Postfix expression.
6.	Write a program which check entered expression is valid or not.
7.	Write a program for evaluation of postfix expression.
8.	Write a program to calculate the factorial of the entered number by using recursion.
9.	Write a program to calculate the digit sum of the entered number by using recursion.
10.	Write a program to find face value of entered number by using recursion.
Queue:	
1.	Write a program to implement Priority queue.
2.	Write a program to implement IRD (Input Restricted Deque)
3.	Write a program to implement ORD (Output Restricted Deque)
Linked List:	
1.	Write a program to implement queue by using linked list. (Dynamic implementation)
2.	Write a program to implement doubly linear linked list with its basic operations.
3.	Write a program to implement singly circular linked list with its basic operations.
4.	Write a program to implement doubly circular linked list with its basic operations.

Type: VSC2 (Vocational and Skill Enhancement Courses)

BCA-II (Semester III)

Course Title: Based on DSC2-3 and DSC2-4 **(Paper Code:)**

Total Lectures: 60 Hrs. **Contact Hrs. (P) :** 4

University Evaluation: 30 Marks **Internal Evaluation:** 20 Marks

Subquery and Join:

A] Table Structure of Account _Master

Field Name	Datatype	Size	Description
Accent	Varchar2	6	Primary Key
Acc_name	Varchar2	6	Account Holders name
Gender	Char	1	Either M or F
Op_Date	Date		The date on which the Account is opened
Chq_fac	Char	1	Cheque facility statue set Default 'N'
Cl_Date	Date		

B] Table Structure of Account _Transaction

Field Name	Datatype	Size	Description
Accent	Varchar2	6	Foreign Key
Tr_amount	Number	7,2	Transaction Amount
Tr_date	Date	1	Transaction Date
Chq_no	Varchar2	10	Cheque Number
Tr_type	Char	1	Receipt or Withdrawal

1.	Display all the details of account holders in ascending order of Op_date.
2.	Display the details of account holders who are having cheque facility and balance >4000.
3.	Display the details of account holders whose balance is between 4000 and 9000.

4.	Display the name and number of female account holder.
5.	Display the details of account holders whose 2nd char of name 'a'.
6.	Display the account of those who have deposited.
7.	Display the details of account holders those who have opened account in 2012.
8.	Display the no of female account holders having cheque facility.
9.	Display the details of 2nd min balance of account holder.
10.	Display the account_no, account_nm, transaction amount, transaction_type of all account holders.
11.	Display the details of account holders who is having maximum balance.
12.	Display the details of account holders whose month of opening the account and transaction are same.
13.	Display the details of account holders who have not made any transaction and whose name is starts with 'R'.
Synonym, Index, View:	
1.	Create a synonym for table student.
2.	Apply DML command on synonym.
3.	Destroy synonym.
4.	Create a view read only view.
5.	Create updateable view.
6.	Apply DML commands on updatable view.
7.	Create read only view for single table.
8.	Destroy view.
9.	Create a sequence for roll_no field in student table.
10.	Insert sequence into table.
11.	Create a cyclic sequence.
12.	Destroy sequence.
13.	Create a cluster table.
14.	Apply DML command on cluster table.
15.	Destroy cluster table.
16.	Create a simple index.

17.	Display an index.
18.	Create composite index.
19.	Create reverse index.
20.	Rebuild reverse index to normal index.
21.	Destroy index.
22.	Create type.
23.	Use type in table.
24.	Insert value for type in table.
25.	Display the table.
26.	Create array for name.
27.	Use array as data type in table.
28.	Insert value for array in table.
29.	Display table.
30.	Create user.
31.	Give grants to the user.
32.	Give grants to the user from own database.
33.	Give grant to a user from another user.
34.	Withdraw once given grant.
35.	Commit work done.
36.	Rollback to commit.
37.	Create savepoint.
38.	Rollback to the particular savepoint.
1.	Write a Python program to calculate mean, median, mode, variance, standard deviation, range, and interquartile range (IQR).
2.	Visualize data using histograms and scatter plots.
3.	Solve numerical problems involving measures of dispersion using Python. Compute and compare ranges, variances, standard deviations, and coefficients of variation for given datasets. Interpret results to conclude data variability and distribution.
4.	Develop Python functions to compute measures of dispersion: I. Range for ungrouped and grouped data II. Variance and Standard Deviation (SD) for ungrouped and grouped data

	III. Coefficient of Variation (CV) to compare variability across different datasets
5.	Implement Python functions to calculate and classify types of correlation: positive, negative, and zero correlation.
6.	Write a Python script to plot a scatter diagram when r = -0.85, r = 0 and r = 0.85.
7.	Write Python scripts to verify basic assumptions underlying the correlation coefficient (linearity, nonlinearity, etc.). Solve numerical problems involving correlation coefficient calculations and interpretation.
8.	Implement Python functions to calculate regression line of x on y using the method of least squares.
9.	Implement Python functions to calculate the regression line of y on x using the method of least squares.
10.	Build a simple linear regression model to predict outcomes (e.g., predict student scores based on study hours).
Practical on Advanced Python Programming	
1.	Write a program to demonstrate higher order functions.
2.	Write a program to demonstrate type hinting.
3.	Write a program to demonstrate recursive functions.
4.	Write a program to demonstrate iterators and generators.
5.	Write a program to demonstrate the use of *args and **kwargs arguments.
6.	Write a program to demonstrate difference between shallow copy and Deep copy.
7.	Write a program to demonstrate various functions of RegEx.
8.	Write a program to demonstrate functions of match object.
9.	Write a program to demonstrate parsing dict to JSON string.
10.	Write a program to demonstrate parsing dict to JSON file.
11.	Write a program to demonstrate parsing JSON string to dict.
12.	Write a program to demonstrate parsing JSON file to dict.
13.	Write a program to read CSV file to dict.
14.	Write a program to write CSV file from dict.

15.	Write a program to create simple spreadsheet and do the following operations- a) Adding cell values b) Updating cell values c) Insert row d) Delete row e) Insert column f) Delete column
16.	Write a program to create simple spreadsheet and do the following operations to manage the sheet- a) Freezing rows and columns b) Adding filter c) Adding formulaes d) Adding styles e) Adding images f) Adding barchart

Type: DSC1-5 (Major)

BCA-II (Semester IV)

Course Title: Advanced Java

(Paper Code:)

Credits: Theory – (2)	**Practical's – (1)**
Total Lectures: 30 Hrs.	**Contact Hrs. (L) :** 2
University Evaluation: 30 Marks	**Internal Evaluation:** 20 Marks

Course Outcomes:

1	Understand and Apply Type Casting & Wrapper Classes.
2	Implement various collection classes such as ArrayList, LinkedList, HashSet, TreeSet, and HashMap for effective data management
3	Implement Multithreading Concepts for Concurrent Execution
4	Develop Network-Based Applications Using Java Networking
5	Design GUI Applications Using Swing and Database Connectivity

Unit	Content	Lect.
1	**Collection Framework**: Type casting, types of type casting **Wrapper classes**: use of Wrapper classes, Number classes(Long,Integer,Byte,Short,Float and double) and importance methods of Number class, Character class and importance methods of character class, auto boxing and unboxing, **Collection Framework**:-Use of Collection framework, Hierarchy of Collection Framework, Collection objets-Set,List, Map, Queue **Collection classes**-Stack,ArrayList,vector,Linked List,priority queue,HashSet,LinkedHashSet,Sotred Set,TreeSet,Hashtable and HashMap **Multithreading:** Single tasking, Multi-tasking, use of thread, creating and running thread, terminating thread, thread class methods, multiple threading, Thread communication, thread priorities,Application of thread and thread life cycle. **Networking:** Introduction to Networking, TCP/IP protocol, UPD protocol, socket programming, InetAdress Class, URL Connection class, communication between client	**15**

	and server, two way communication between client and server.	
2	**Swing:** Hierarchy of Swing classes JButton, JLabelJava ,JTextField, JTextArea, JPasswordField, JCheckBox, JRadioButton,JComboBox, , JList, JOptionPaneJava JScrollBar, JMenuItem & JMenuJava , Image **Event handling**:- Java Event Handling, Java Event classes and Listener interfaces. LayoutManager- BorderLayout FlowLayout, GridLayout, CardLayout , BoxLayout **JDBC**: Introduction, JDBC Driver, DB Connectivity Steps ,Connectivity with Oracle or MySql DriverManager, ConnectionStatement, ResultSet, PreparedStatement, ResultSetMetaData, CallableStatement	**15**

Reference Books:

1.	"Programming with Java" by E Balaguruswamy
2.	Horstmann, "Computing Concepts with Java 2 Essentials", John Wiley.
3.	Decker & Hirshfield, "Programming Java", Vikas Publication
4.	"Java-2 the complete Reference" by Patrick Naughton and HerbertzSchidt

Type: Practical (Major) **BCA-II (Semester IV)** **Course Title: Practical based on DSC1-5** **(Paper Code:)**	
Total Lectures: 30 Hrs.	**Contact Hrs. (P) :** 2
University Evaluation: 15 Marks	**Internal Evaluation:** 10 Marks
1	Write a Java program to demonstrate implicit and explicit type casting.
2	Develop a Java program to illustrate the use of wrapper classes.
3	Write a program to implement a Java program to compare ArrayList, LinkedList, Vector, and Stack
4	Create a Java application to demonstrate the differences between HashSet, LinkedHashSet, TreeSet, and SortedSet.
5	Write a program to implement a Java program using both Thread class and Runnable interface.
6	Write a Java program where two threads communicate with each other using the wait() and notify() methods.
7	Write a Java application demonstrating thread priorities.
8	Write a program for two-way chat application using Java TCP sockets.
9	Write a Java program to demonstrate UDP communication.
10	Develop a student registration form using Swing components such as JTextField, JPasswordField, JCheckBox, JRadioButton, JComboBox**,** and handle different events.
11	Create **a** Java application with a menu bar containing options like "File," "Edit," and "Help." Use different layout managers (BorderLayout, FlowLayout, GridLayout, etc.) for different sections.
12	Write a JDBC program to connect to a MySQL or Oracle database, create a table, insert, update, delete, and retrieve records using Statement and PreparedStatement**.**

Type: DSC1-6(Major)

BCA-II (Semester IV)

Course Title: Data Structures Using C++-II

(Paper Code:)

Credits: Theory – (2) Practical – (12)

Total Lectures: 30 Hrs. **Contact Hrs. (L) :** 2

University Evaluation: 30 Marks **Internal Evaluation:** 20 Marks

Course Outcomes:

1.	Understand Tree Structures and Their Applications.
2.	Apply Operations on Binary Search Trees (BST) and Balanced Trees
3.	Analyze Graph Theory Concepts and Graph Algorithms
4.	Implement and Compare Sorting Techniques
5.	Develop Efficient Searching and Hashing Algorithms

Unit	Content	Lect.
1	**Tree:** Introduction to Tree, Introduction to Binary Trees, Types of Binary tree- Strictly Binary tree, Complete Binary tree, Extended (2-Tree) Binary tree, Binary expression tree, Binary Search tree, Heap Tree- Min heap tree, Max heap tree, Representation of Binary tree using- Array, Linked list **Operations of Binary search tree**-Creating and inserting node, Searching node, Counting total nodes, Counting and displaying leaf nodes, Tree Traversal methods- Preorder, Inorder, Postorder, Deletion of Nodes, Implementation of binary search tree, Height balanced tree/Balanced Binary Tree/AVL tree, Application of tree **Graph:**Concept & terminologies used in graph, Graph Representation using- Array and linked list, Graph traversals – BFS & DFS, Dijakstra's shortest path algorithm, and application of graph.	**15**
2	**Sorting:** Introduction and definition of Sorting, Types of Sorting-Bubble sort, Quick sort, Shell	**15**

	sort, Selection sort, Insertion sort, Heap Sort, Merge sort, Radix Sort, Tree Sort techniques, Complexity of these algorithms **Searching:** Introduction and definition of Searching, Types of searching-Linear (Sequential) Search, Binary Search, Indexed sequential search, Hashing and different Hash functions.	

Reference Books:

1.	Tanenbaum: Data structures using C and C++
2.	Fundamentals of Data Structures and Algorithms by Dr. Mente R.S.
3.	Data Structures Through C in Depth- S.K.Srivastava, D.Srivastava
4.	Fundamentals of Data Structures in C by Sahni

Type: Practical(Major) **BCA-II (Semester IV)** **Course Title: Practical based on DSC1-6** **(Paper Code:)**	
Total Lectures: 30 Hrs. **Contact Hrs. (P) :** 2	
University Evaluation: 15 Marks **Internal Evaluation:** 10 Marks	
1.	Implement Binary Tree representation using both an array and a linked list.
2.	Write a Java program to create a Binary Search Tree (BST).
3.	Write a program to implement functions for searching a node, counting total nodes, and counting/displaying leaf nodes in a BST.
4.	Write a program to represent a graph using an adjacency matrix and an adjacency list.
5.	Write a program to implement Breadth-First Search (BFS) and Depth-First Search (DFS)
6.	Write a Java program to implement Dijkstra's algorithm for finding the shortest path between nodes in a weighted graph.
7.	Write a program to implement Bubble Sort, Selection Sort, and Insertion Sort
8.	Write a program to implement both Quick Sort and Merge Sort
9.	Write a program to implement Linear Search.
10.	Write a program to implement Binary Search

Type: DSC2-5(Minor)

BCA-II (Semester IV)

Course Title: Relational Database Management System

(Paper Code:)

Credits: Theory – (2)	**Practical – (1)**
Total Lectures: 30 Hrs.	**Contact Hrs. (L) :** 2
University Evaluation: 30 Marks	**Internal Evaluation:** 20 Marks

Course Outcomes:

1.	To Understand the PL/SQL architecture.
2.	To learn programming, management, and security issues of working with PL/SQL program units,
3.	To understand the built-in packages and user-defined packages.
4.	To write PL/SQL code for procedures, triggers, cursors, exception handling, etc.

Unit	Content	Lect.
1	**Introduction of Transaction**: ACID properties, transaction states, scheduling, and its types, conflict, and view serializability. **Introduction of Concurrency Control:** Problems of concurrency control, lock-based protocols, timestamp-based protocol, deadlock, and deadlock handling methods. **Introduction, recovery algorithms**: log base recovery, shadow paging, recovery with concurrent transactions, checkpoints or savepoints. **Query Optimization:** Overview Query Processing and Optimization – Heuristics and Cost Estimates in Query Optimization.	**15**
	Introduction to PL/SQL: Advantages, Architecture, Datatypes, Variable and Constants, Using Built_in Functions,	**15**

2	Conditional, Looping and Iterations Statements, Selection Case, Simple Case, Goto Label and exit, SQL Within PL/SQL. **Procedures in PL/SQL:** Stored Procedures, Procedure with Parameters (In, Out and In Out), Positional Notation and Named Notation, Dropping a Procedure. **Functions in PL/SQL:** Difference between Procedures and Functions, types of functions and parameter modes, **Packages in PL/SQL:** importance, advantages Implementing packages, Private and Public Objects in Package. **Cursor in PL/SQL:** Types of Cursors, Cursor Attributes, Cursor with Parameters, Cursors with Loops Nested Cursors, Cursors with Sub Queries and Procedure. **Exceptions in PL/SQL:** Types of exceptions, Raise_Application_Error, Pragma_Autonomous_Transaction **Database Triggers in PL/SQL:** Types of Triggers, Row Level Triggers, Statement Level Triggers, Implementing triggers for various DML operations (insert, delete, update), DDL Triggers, Trigger Auditing.	

Reference Books:

1.	Database System Concepts by Korth Silberschetz
2.	Fundamentals of Database Systems by Elmsari, Navathe
3.	SQL, PL/SQL – The Programming Language of Oracle by Ivan Bayross
4.	Database Management System by Seema Kedar

Type: Practical(Minor) **BCA-II (Semester IV)** **Course Title: Practical based on DSC2-5** **(Paper Code:)**	
Total Lectures: 30 Hrs.	**Contact Hrs. (P) :** 2
University Evaluation: 15 Marks	**Internal Evaluation:** 10 Marks
1.	Write a PL/SQL code to demonstrate the use of constant keyword.
2.	Write a PL/SQL code to calculate simple interest.
3.	Write a PL/SQL code to calculate Compound interest.
4.	Write a PL/SQL code to calculate factorial by using a function.
5.	Write a PL/SQL code to calculate Armstrong by using the procedure.
6.	Create a cursor for updating the salary of employees by 5%.
7.	Create a cursor to display the 5 highest earners.
8.	Create a trigger for generating the primary key.
9.	Create a trigger for generating a primary key when some data is already present.
10.	To generate the exception when Data is not found in the table.

Type: DSC2-6(Minor)

BCA-II (Semester IV)

Course Title: Software Testing and Quality Assurance

(Paper Code:)

Credits: Theory – (2) **Practical – (1)**

Total Lectures: 30 Hrs. **Contact Hrs. (L) :** 2

University Evaluation: 30 Marks **Internal Evaluation:** 20 Marks

Course Outcomes:

1.	List a range of different software testing techniques and strategies and be able to apply specific (automated) unit-testing method for the projects.
2.	Distinguish characteristics of structural testing methods.
3.	Demonstrate the integration testing which aims to uncover interaction and compatibility problems as early as possible.
4.	Discuss about the functional and system testing methods.
5.	Demonstrate various issues for object-oriented testing.

Unit	Content	Lect.
1	**Introduction To Software Testing:** Importance or need of software testing Differences between Manual and Automation Testing **Introduction to White Box Testing:** Advantages and Disadvantages of White box testing Static Techniques: Informal Reviews, Walkthroughs, Technical Reviews, Inspection Dynamic Techniques or Structural Techniques Statement Coverage Testing, Branch Coverage Testing Path Coverage Testing, Conditional Coverage Testing, Loop Coverage Testing	**15**

	Introduction to Black Box Testing: Advantages and Disadvantages of black box testing Black Box Techniques: Boundary Value Analysis, Equivalence Class Partition, State Transition Technique, Cause Effective Graph, Decision Table, Use Case Testing Experienced Based Techniques: Error guessing, Exploratory testing Levels of Testing Functional Testing Integration Testing and types - Top Down , Bottom Up , Non Incremental System Testing Acceptance Testing- Alpha and Beta Smoke Testing Regression Tesng- Unit , Regional, Full Non Functional Testing Adhoc Testing Performance Testing: Load Testing, Stress Testing, Volume Testing, Soak Testing Recovery Testing	
2	**Test cases design Techniques:** Introduction to Test Case and Types. Test Case Template How to write a test case and examples Preparing Review Report Software Test Life cycle Software Test Life Cycle: Writing Test Plan Preparing Traceability Matrix Writing Test Execution Report and Summary Report **Defect Life Cycle:** Bug/ Defect Life Cycle: Difference between Bug, Defect, Failure, Error	**15**

	Defect Tracking and Reporting Types of Bugs, Identifying the Bugs, Reporting the Bugs **Introduction to automated testing**- Install and configure selenium testing tool, Case study: Design test case for login page, Internet Banking Login, and Online shopping. **TestNG Testing framework:** Introduction TestNG, Annotation, Methods, Create Auto Testcase, create auto report.	

Reference Books:

1.	The art of Software Testing – Glenford J. Myers
2.	Lessons learned in Software Testing – Cem Kaner, James Bach, Bret Pettichord
3.	A Practitioner’s Guide to Software Test Design- Lee Copeland

Type: Practical(Minor) **BCA-II (Semester IV)** **Course Title: DSC2-6 (Paper Code:)**	
Total Lectures: 30 Hrs.	**Contact Hrs. (P) :** 2
University Evaluation: 15 Marks	**Internal Evaluation:** 10 Marks
1.	Write and test a program to handle alert, frame and windows
2.	Write and test a program to upload file.
3.	Write and test a program to download the file.
4.	Write and test a program to handing Menus.
5.	Write and test a program handling web table records.
6.	Write and test a program handling slider.
7.	Write and test a program color, fonts and size of text on a web page
8.	Write and test a program handling browser windows
9.	Write and test a program handling date picker.
10.	Write and test a program to provide a total number of objects present/available on the page.

Type: GE4 / OE4 (Generic / Open Elective)

BCA-II (Semester IV)

Course Title: Digital Marketing

(Paper Code:)

Credits: Theory – (2)

Total Lectures: 30 Hrs. **Contact Hrs. (L) :** 2

University Evaluation: 30 Marks **Internal Evaluation:** 20 Marks

Course Outcomes:

1.	Understand the Fundamentals of Digital Marketing and Its Business Impact
2.	Apply Search Engine Optimization (SEO) Techniques for Traffic and Revenue Generation
3.	Leverage Social Media Optimization (SMO) for Brand Building and Marketing
4.	Implement Search Engine Marketing (SEM) and Online Advertising Strategies
5.	Analyze Website Traffic and User Behavior Using Web Analytics Tools

Unit	Content	Lect.
1	**Introduction to Digital Marketing:** Meaning of Digital Marketing, Differences from Traditional Marketing, Return of Investments on Digital Marketing vs. Traditional Marketing, E Commerce, Tools used for successful marketing, SWOT Analysis of Business for Digital Marketing, Media and promotion plan, Blogs, Websites, Portal and Their Differences, Visibility,Visitor Engagement, Conversion Process, Retention, Performance Evaluation, Online Reputation Management **Search Engine Optimization (SEO):** Optimization, Definition, its importance, Strategies and techniques used to optimize any article/page/website/blog for traffic generation and revenue, different On page Optimization Techniques, different Off Page Optimization Techniques, Preparing Reports, Creating Search Campaigns, and Creating Display Campaigns. **Social Media Optimization (SMO):** Introduction to Social media, Types of Social Media platforms (Facebook, Twitter, Instagram, YouTube), Roles of Social Media in Marketing, Goals and Strategies, Facebook Marketing, Email Marketing, Google plus marketing , Word press Blog Creation,	**15**

	Twitter Marketing, Linkedln Marketing, Pinterest, Instagram Marketing, eCommerce Marketing, Affiliate Marketing, SMS Marketing, mage Optimization, social media Analytical Tools.	
2	**Search Engine Marketing:** Introduction and Use of Search Engine Marketing, Introduction to Online Advertising and Ad words, Tools used — Pay Per Click, Google Adwords, Display Advertising Techniques, Advertisement Designing, Adwords Account And Campaign Basics, Adwords Targeting And Placement, Adwords Bidding And Budgeting, Adwords Tools, Opportunities, Optimizing Performance, Ads Type, Bidding Strategies, Search Network, Display Network, Shopping Ads, Video Ads, Universal App Ads, Tracking Script, Remarketing, Performance Monitoring, Report Generation. **Website Traffic Analysis:** Web Analytics Tools, Google Analytics, Navigating Google Analytics, Traffic Sources, Acquisition, Behavior, Content, Visitors, Live Data, Demographics	**15**
Reference Books:		
1.	Ryan, D. (2014). Understanding Digital Marketing: Marketing Strategies for Engaging the Digital Generation, Kogan Page Limited.	
2.	The Beginner's Guide to Digital Marketing (2015). Digital Marketer. Pulizzi,J.(2014) Epic Content Marketing, Mcgraw Hill Education.	
3.	Lorrie Thomas. The McGraw-Hill 36-Hour Course: Online Marketing, McGraw Hill.	

Type: VSC3 (Vocational and Skill Enhancement Courses) **BCA-II (Semester III)** **Course Title: Based on DSC1-5 and DSC1-6** **(Paper Code:)**	
Total Lectures: 60 Hrs.	**Contact Hrs. (P) :** 4
University Evaluation: 30 Marks	**Internal Evaluation:** 20 Marks
1.	Write a program to show that during function overloading, if no matching argument is found, then java will apply automatic type conversions (from lower to higher data type).
2.	Write a program to show the difference between public and private access specifiers. The program should also show that primitive data types are passed by value and objects are passed by reference and to learn use of final keyword .
3.	Write a program to show the use of static variable, static method and static block.
4.	Write a program to demonstrate the concept of boxing and unboxing.
5.	Write a program to demonstrate Arithmetic Exception.
6.	Write a program to demonstrate FileNotFound Exception.
7.	Write a program to demonstrate ArrayIndexOutOfRange Exception.
8.	Write a program to demonstate the use of throw and throws keyword.
9.	Write a program to create a package and import it.
10.	Write a program to demonstrate the use of Interfaces
11.	Write a program to demonstrate dafualt and parametrised constructor.
12.	Write a program to demonstrate the polymorphism.
13.	Write a rogram to demonstate the use of super keyword.
14.	Write a program to demonstrate thread life cycle.
15.	Write a program to demonstrate the inter-thread communication
16.	Write program to count the number of ovels in the file.
17.	Write a program to copy the content of one file and append in another file.

Practical on Data structure-II	
1.	Write a program to implement BST with the following operations: I) Insert Node II) Count Leaf nodes III)Count Non-Leaf nodes IV) Count Total nodes
2.	Write a program to implement BST with the following operations: I) Insert Node II) Find Maximum node III) Find Minimum Node IV) Search node V) Display only odd nodes VI) Display only even nodes VII) Display leaf nodes VIII) Find the level of node IX) Find the degree of node X) Delete Node
3.	Write a program to implement BFS
4.	Write a program to implement DFS
5.	Write a program to implement Linear Search.
6.	Write a program to implement Binary Search
7.	Write a program to implement Bubble Sort.
8.	Write a program to implement Merge Sort.
9.	Write a program to implement Quick Sort.
10.	Write a program to implement Selection Sort.
11.	Write a program to implement Insertion Sort.

Type: VSC4 (Vocational and Skill Enhancement Courses)

BCA-II (Semester III)

Course Title: Based on DSC2-5 and DSC2-6 **(Paper Code:)**

Total Lectures: 60 Hrs. **Contact Hrs. (P) :** 4

University Evaluation: 30 Marks **Internal Evaluation:** 20 Marks

Simple PL/SQL Block:	
1.	Write a PL/SQL code to calculate Area ,circumference and perimeter of circle.
2.	Write a PL/SQL code to calculate Area and perimeter of rectangle.
3.	Write a PL/SQL code to calculate Area and perimeter of Triangle.
4.	Write a PL/SQL code to calculate Area and perimeter of Square.
5.	Write a PL/SQL code to find out maximum no between two no's.
6.	Write a PL/SQL code to find out maximum no between three no's.
7.	Write a PL/SQL code to find out the minimum no between two no's.
8.	Write a PL/SQL code to find out minimum no between three no's.
9.	Write a PL/SQL code to find out given no is even or odd.
10.	Write a PL/SQL code to find out given no is Positive, Negative or Zero.
11.	Write a PL/SQL code to display the message when user entering no then displays the appropriate no in word (eg.1-one).
12.	Write a PL/SQL code to display for month name.
13.	Write a PL/SQL code to display for week day name.
14.	Write a PL/SQL code to display the grade of the student.
15.	Write a PL/SQL code to display for 1-10 numbers.
16.	Write a PL/SQL code to display for 1-10 numbers in descending order.
17.	Write a PL/SQL code to calculate the factorial of any no.
18.	Write a PL/SQL code to calculate sum of digits.
19.	Write a PL/SQL code to find given no is prime or not.

20.	Write a PL/SQL code to find given no is palindrome or not.
21.	Write a PL/SQL code to find given no is Armstrong or not.
22.	Write a PL/SQL code to find given no is composite or not.
23.	Write a PL/SQL code to find given no is strong or not.
24.	Write a PL/SQL code to find given no is perfect or not.
25.	Write a PL/SQL code to find face value of given no.
26.	Write a PL/SQL code to find area of circle and insert this area in to the table. (Take radius from table)
27.	Write a PL/SQL code to calculate addition, substation, division, and multiplication of any two no's.
28.	Write a PL/SQL code to find prim no bet 1 to 1000.
29.	Write a PL/SQL code to find perfect no bet 1 to 1000.
30.	Write a PL/SQL code to find strong no bet 1 to 1000.
31.	Write a PL/SQL code to find palindrome no bet 1 to 1000.
32.	Write a PL/SQL code to find Armstrong no bet 1 to 1000.
33.	Write a PL/SQL code to find composite no bet 1 to 1000.
34.	Write a PL/SQL code to find 1st 10 palindrome no’s.
35.	Write a PL/SQL code to find 1st 10 composite no’s.
36.	Write a PL/SQL code to print the multiplication table.
37.	Write a PL/SQL code to find given string palindrome or not.
Functions and Procedures:	
1.	Write a PL/SQL code for calculate Armstrong by using a function.
2.	Write a PL/SQL code calculate palindrome by using a function.
3.	Write a PL/SQL code for calculate reverse no by using the function.
4.	Write a PL/SQL code for calculate face value by using a function.
5.	Write a PL/SQL code for calculate the perfect no by using a function.
6.	Write a PL/SQL code for calculate the prime by using a function.

7.	Write a PL/SQL code for calculate strong no by using procedure.
8.	Write a PL/SQL code for calculate palindrome by using procedure.
9.	Write a PL/SQL code for calculate the perfect no by using the procedure.
10.	Write a PL/SQL code for calculate prime by using procedure.
11.	Write a PL/SQL code for calculate reverse no by using procedure.
12.	Write a PL/SQL code for calculate factorial by using function.
Cursor:	
1.	Create a cursor for display 5 lowest earners.
2.	Create a cursor for calculation of area of circle and insert in to table. (Take radius from table)
3.	Create a cursor for transaction balance becomes less than 500 or the day is Sunday of transaction then display message transaction is not allowed.
4.	For display user defined message when the cursor not fetching row.
5.	For display user message when cursor fetching more than one row or one value.
Trigger:	
1.	Create a trigger for backup.
2.	Create a trigger for display message when user enters <0 value in students roll_no column.
3.	Create a trigger for transaction if transaction balance becomes <500 or transaction day is Sunday then reject the transaction.
Exception Handling and package :	
1.	To generate the exception Too many Rows are found in the table.
2.	To generate the exception for Duplicate index.
3.	To generate the exception for display value_Error.
4.	To generate user defined exception for divide by zero value.
5.	W.A.PL/SQL block to generate your own exception for more amounts withdrawing than balance.
6.	Create a package for combine function and procedure.

Practical on Software Testing and Quality Assurance	
1.	Design test case for college admission Application
2.	Design test case for social networking sites.
3.	Design test case for MS-word application.
4.	Design test case for simple calculator.
5.	Design test case for ball pen.
6.	Design test case for Paint application.
7.	Design test case for ATM application.
8.	Install Selenium and integrated testng framework, Write a test suite containing minimum 4 test cases for login page.
9.	Write and test a program to sign up a specific web page.
10.	Write and test a program to update records of 10 students from Excel file.
11.	Write and test a program to select the number of students who have scored more than 60 in any one subject (or all subjects).
12.	Write and test a program to get the number of items in a list / combo box.
13.	Write and test a program to count the number of check boxes on the page checked and unchecked count.
14.	Write and test a program to count the number of links on the web page.
15.	Write and test a program for online payment transaction.
16.	Write and test a program create email account
17.	Write and test a program apply job on job portal.

Punyashlok Ahilyadevi Holkar Solapur University, Solapur

Faculty of Science and Technology

Equivalent Subject for CBCS Syllabus of B.C.A. - II (Semester–III and IV)

Semester-III		
Sr. No.	**Name of the CBCS Paper (w.e.f. 2023-2024)**	**Name of the New Paper as per NEP2020 (w.e.f. 2025-2026)**
1	Object Oriented Programming using Java-I	Core Java
2	Data Structure Using 'C'-I	Data Structures Using C-I
3	Database Management System-I	Database Management Systems (DBMS)
4	Software Engineering	No Equivalence
5	FA Using Tally	No Equivalence
6	Operating System-II	Operating Systems
7	Web Development using PHP	No Equivalence
Semester-IV		
Sr. No.	**Name of the CBCS Paper (w.e.f. 2023-2024)**	**Name of the New Paper as per NEP2020 (w.e.f. 2025-2026)**
1	Object Oriented Programming using Java-II	Advanced Java
2	Data Structure Using 'C'-II	Data Structures Using C-II
3	Database Management System-II	Relational Database Management System (RDBMS)
4	Software Testing & QA	Software Testing and Quality Assurance
5	Digital Marketing	Digital Marketing
6	Python-II	Advanced Python Programming (Sem-III)

For Science faculty: CA- Continuous Assessment (Internal Examinations) of Total Marks: (40%)

The pattern / Examination nature may be as follows:

One internal examination of 40% marks or two examinations of 20% marks each.

Open book examination / Home Assignment / Classroom test / Seminar / Field Work report / Project Report etc.