

Ahmednagar Jilha Maratha Vidya Prasarak Samaj's
New Arts, Commerce, and Science College,
Ahmednagar (Autonomous)
(Affiliated to Savitribai Phule Pune University, Pune)



National Education Policy (NEP)
Choice Based Credit System
(CBCS)

Programme Framework
B.B.A.(C.A.)(Major)

Implemented from
Academic Year 2024-25

Ahmednagar Jilha Maratha Vidya Prasarak Samaj's
New Arts, Commerce and Science College, Ahmednagar
(Autonomous)

Board of Studies in B.B.A.(C.A.)

Sr. No.	Name	Designation
1.	Mrs. Nimbalkar Sangita Sham	Chairman
2.	Mr. Talule Sonyabapu Sakharam	Member
3.	Mr. Gobare Manohar B.	Member
4.	Miss. Danave Bharati M.	Member
5.	Mr. Pachpande Suhas D.	Academic Council Nominee
6.	Dr. Patil Chandrashekhar Himmatrao	Academic Council Nominee
7.	Prof. (Mrs.) Siddavatam A. Shakilabanu	Vice-Chancellor Nominee
8.	Mrs. Mohite-Patil Amruta Rahul	Alumni
9.	Mr. Dawbhat Arun Rangnath	Industry Expert
10.	Mrs. Kulkarni Aparna A.	Member(co-opt)
11.	Mrs. Suroshi M.S.	Member
12.	Mr. Bade R.K.	Member
13.	Mr. Supekar K.A.	Member

1. Prologue/ Introduction of the programme:

With the rapid growth of IT industry in India, the demand of computer professional is increasing day by day. This increasing growth of IT industry has created a lot of opportunities for the computer graduates.

B.B.A.(C.A.) program is a full time three years degree program with six semesters. It is based on Choice-based credit system containing 132 credit points.

B.B.A.(C.A.) program is a combination of computer and applied courses from Commerce and management stream. Computer related courses introduce techniques of programming,

databases, web designing, system analysis, design tools, data mining, and different computing environments. Applied courses include Financial Accounting, Principles of Management, Organization Behavior and Human Resource Management etc that provide business administration foundation of the program.

This course provides a lot of opportunities to arts, commerce and science stream students who are interested in computer field and wants to work in the IT sector as programmer or software developer. This Degree will help students to become an IT professional and to be placed in the network support and system support/ administration roles. Student can either work in the corporate sector in an administration.

The Course is planned and structured to provide you with a dynamically engaging atmosphere in which you can develop into highly qualified IT professionals. The curriculum has been intended to provide students with a thorough understanding of numerous areas linked to information technology as well as basic management concepts. This course offers the prequalification for professionals heading for smart career in the IT field, which measures up to international standards.

There are endless opportunities in the sector. For example you can be hired as Software Developer, Technical Analyst, System Administrators, Programmer, Tech support and others.

2. Programme outcomes (Pos)

Students enrolled in the program complete a curriculum that exposes and trains students in a full range of essential skills and abilities. They will have the opportunity to master the following objectives.

1. Imparts advanced knowledge on a wide range of computer applications so that the students pursuing this course can easily face any kind of challenges and opportunities related to the IT industry.
2. Inculcate spirit of entrepreneurship.
3. Import practical skills among students.
4. Prepare as a industry ready resource.
5. Implement the spirit of entrepreneurship.
6. Develop and produce skilled computer professionals.
7. Prepare students to face the diverse challenges and opportunities in the IT industry and build competence in a particular area of business.

Credit Distribution: B.B.A.(C.A.)(Major) including Minor and OE and other courses.

	Type of Courses	III Yr	IV Yrs (Honours)	IV Yrs Research
Major Marathi	Discipline-Specific Courses (DSC)	46	74	66
	Discipline Specific Elective (DSE)	08	16	16
	Skill Enhancement Courses (SEC)	06	06	06
	Vocational Skill Courses (VSC)	08	08	08
	On-Job Training (OJT)	04	08	04
	Field Project (FP)	04	04	04
	Community Engagement and Service (CEP)	02	02	02
	Research project	00	00	12
	Research Methodology	00	04	04
	Indian Knowledge System	02	02	02
	Total (I, II and III Year)	80	124	124
Minor	Minor	20	20	20
Other Courses	Open Elective (OE)/ Multidisciplinary Courses	12	12	12
	Co-Curricular Courses	08	08	08
	Ability Enhancement Courses	08	08	08
	Value Education Courses	04	04	04
	Total	132	176	176

B.B.A.(C.A.) Programme Framework: Credit Distribution

			D S C	D S E	SEC	V S C	FP/OJT/IN/CEP	I K S							
I	I	4.5	06	-	02	-	-	02	03	03		02	02	02	22
I	II	4.5	06	-	02	02	-	-	03	03	-	02	02	02	22
Exit Option: Award of UG Certificate in Major with 44 credits and an additional 4 credit core NSQF course /Internship or Continue with Major and Minor															
II	III	5.0	08	-	02	-	FP-02	-	03	03	-	02	02	-	22
II	IV	5.0	08	-	00	02	CEP-02	-	03	03	-	02	02	-	22
Exit Option: Award of UG Diploma in Major with 88 credits and an additional 4 credit core NSQF course /Internship or Continue with Major and Minor															
III	V	5.5	10	04	-	02	FP-02	-	04	-	-	-	-	-	22

III	VI	5.5	08	04	-	02	OJT-04	-	04	-	-	-	-	-	22
Exit Option: Award of 3-Yr UG Degree in Major and Minor with 132 credits or continue with Major for a 4-year Degree															
IV	VII	6.0	14	04	RM:04	-	-	-	-	-	-	-	-	-	22
IV	VII I	6.0	14	04	-	-	OJT-04	-	-	-	-	-	-	-	22
4-Yr UG Degree (Honours) with Major and Minor with 176 credits															
IV	VII	6.0	10	04	RM:04	-	RP-04	-	-	-	-	-	-	-	22
IV	VII I	6.0	10	04	-	-	RP-08	-	-	-	-	-	-	-	22
4-Yr UG Degree (Honours with Research) with Major and Minor with 176 credits															

B.B.A.(C.A.) Programme Framework: Course Distribution

Y	S	L	Major					K	MO	CA	V	TI
			D	D	V	I	J					
1	1	4	0	0	0	0	0	0	0	0	0	
1	4	5	0	0	0	0	0	0	0	0	0	
Exit Option: Award of UG Certificate in Major with 44 credits and an additional 4 credit core NSQF												
1	5	0	0	0	0	0	0	0	0	0	0	
1	5	0	0	0	0	0	0	0	0	0	0	
Exit Option: Award of UG Diploma in Major with 88 credits and an additional 4 credit core NSQF												

I I I	V	5		0 3	0 1	-	0 1	01	-	0 1		-	-	-	0 7
I I I	VI	5		0 2	0 1	-	0 1	01	-	0 1		-	-	-	0 6
			Exit Option: Award of 3-Yr UG Degree in Major and Minor with 132 credits or continue with Major for a 4-year Degree												
I V	VII	6		0 4	0 1	RM:0 1	-	-	-	-		-	-	-	0 6
I V	VII I	6		0 4	0 1	-	-	01	-	-		-	-	-	0 6
			4-Yr UG Degree (Honours) with Major with 176 credits												
I V	VII	6		0 3	0 1	RM:0 1	-	01	-	-		-	-	-	0 6
I V	VII I	6		0 3	0 1	-	-	01	-	-		-	-	-	0 5
			4-Yr UG Degree (Honours with Research) with Major with 176 credits												

Programme Framework (Course Distribution): B.B.A.(C.A.)(Major)

Year	Semester	Level	Major		SEC	VSC	FP/OJT /IN/CEP	IKS
			DSC	DSE				
I	I	4.5	02	00	01	00	00	01
I	II	4.5	02	00	01	01	00	-
II	III	5.0	03	00	01	00	FP-01	-
II	IV	5.0	03	00	00	01	CEP-01	-
III	V	5.5	03	01	00	01	FP-01	-
III	VI	5.5	02	01	00	01	OJT-01	-
IV	VII	6.0	04	01	RM:01	00	-	-
IV	VIII	6.0	04	01	00	00	OJT-01	-
IV	VII	6.0	03	01	RM:01	00	RP-01	-
IV	VIII	6.0	03	01	00	00	RP-01	-

Programme Framework (Credit Distribution): B.B.A.(C.A.)(Major)

Year	Semester	Level	Major					
			DSC	DSE	SEC	VSC	FP/OJT/IN/CEP	IKS
I	I	4.5	06	00	02	00	00	02
I	II	4.5	06	00	02	02	00	-
Exit Option: Award of UG Certificate with 44 credits and an additional 4 credit core NSQF course /Internship or Continue with Major and Minor								
II	III	5.0	08	00	02	00	FP-02	-
II	IV	5.0	08	00	00	02	CEP-02	-
Exit Option: Award of UG Diploma with 88 credits and an additional 4 credit core NSQF course /Internship or Continue with Major and Minor								
III	V	5.5	10	04	00	02	FP-02	-
III	VI	5.5	08	04	00	02	OJT-04	-
Exit Option: Award of UG Degree in Major and Minor with 132 credits or continue with a Major for 4-year Degree with honours or honours with research								
IV	VII	6.0	14	04	RM:04	00	00	-
IV	VIII	6.0	14	04	00	00	OJT-04	-
4-year Degree (Honours)								
IV	VII	6.0	10	04	RM:04	00	RP-04	-
IV	VIII	6.0	10	04	00	00	RP-08	-
4-year Degree (Honours with Research)								

Programme Framework (Courses and Credits): B.B.A.(C.A.)(Major)

Sr. No.	Year	Semester	Level	Course Type	Course Code	Title	Credits
1.	I	I	4.5	DSC-1	BBACA111T	Computer Fundamentals and Office Automation	03
2.	I	I	4.5	DSC-2	BBACA112T	Programming Language C	03
3.	I	I	4.5	SEC-1	BBACA113P	Practical (CF and C)	02
4.	I	I	4.5	IKS-1	BBACA114T	Computing Science in Ancient India	02
5.	I	II	4.5	DSC-3	BBACA121T	Database Management System	03
6.	I	II	4.5	DSC-4	BBACA122T	Web Technology	03
7.	I	II	4.5	SEC-2	BBACA123P	Practical(DBMS)	02
8.	I	II	4.5	VSC-1	BBACA124P	Practical (WEB)	02

9.	II	III	5.0	DSC-5	BBACA231T	Relational Database Management System	03
10.	II	III	5.0	DSC-6	BBACA232T	Web Development with PHP	03
11.	II	III	5.0	DSC-7	BBACA233T	Data Structure using C	02
12.	II	III	5.0	SEC-3	BBACA234P	Practical (RDBMS)	02
13.	II	III	5.0	FP-01	BBACA235P	Practical (PHP & DS)	02
14.	II	IV	5.0	DSC-8	BBACA241T	Object Oriented Programming Using C++	03
15.	II	IV	5.0	DSC-9	BBACA242T	Advanced Web Development	03
16.	II	IV	5.0	DSC-10	BBACA243T	Software Engineering	02
17.	II	IV	5.0	VSC-2	BBACA244P	Practical (C++)	02
18.	II	IV	5.0	CEP-01	BBACA245P	Practical (AWT)	02
19.	III	V	5.5	DSC-11	BBACA351T	Python Programming	04
20.	III	V	5.5	DSC-12	BBACA352T	Programming in Core JAVA	04
21.	III	V	5.5	DSC-13	BBACA353T	OOSE	02
22.	III	V	5.5	DSE-01	BBACA354T	Angular JS/Block Chain(T+P)	04
23.	III	V	5.5	VSC-3	BBACA355P	Practical	02
24.	III	V	5.5	FP-02	BBACA356P	Practical/Project	02
25.	III	VI	5.5	DSC-14	BBACA361T	Advanced JAVA	04
26.	III	VI	5.5	DSC-15	BBACA362T	React	04
27.	III	VI	5.5	DSE-2	BBACA363T	Software Testing/ Mongo DB	04
28.	III	VI	5.5	VSC-4	BBACA364P	Practical	02
29.	III	VI	5.5	OJT-01	BBACA365P	Practical/Project	04

**Ahmednagar Jilha Maratha Vidya Prasarak Samaj's
New Arts, Commerce and Science College,
Ahmednagar(Autonomous)
Syllabus B.B.A.(C.A.)(Major)**

Title of the Course: Relational Database Management System(RDBMS)								
Year: I				Semester: III				
Course Type	Course Code	Credit Distribution		Credits	Allotted Hours	Allotted Marks		
		Theory	Practical			CI E	ES E	Total
DSC-5	BBACA231T	03	00	03	45	30	70	100

Learning Objectives:

- 1) To know the advanced features of DBMS.
- 2) To Understand the Basic programs in PL/SQL.
- 3) To understand how to use SQL in PL/SQL program.

Course Outcomes (Cos):

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

- Understand the basic concepts and the applications of database systems.
- Master the basics of SQL and construct queries using SQL.
- Understand the relational database design principles.
- Familiar with the basic issues of transaction processing and concurrency control.
- Familiar with database storage structures and access techniques.

Detailed Syllabus:

Unit 1. Introduction to RDBMS (5 Hours)

1. Introduction to RDBMS. Popular RDBMS Products and their Features
2. Difference between DBMS and RDBMS
3. Relationship among Application Programs and RDBMS

Unit 2. PL/SQL (10 Hours)

2.1 Overview of PL/SQL

1. Data Types
2. PL/SQL block
 - 2.2.1 %type, %rowtype
 - 2.2.2 Operators, Functions, Comparison, Numeric, Character, Date
 - 2.2.3 Control Statement
2. Exception Handling - Exception types
3. Cursor
 1. Definition
 2. Types of Cursor – Implicit, Explicit (Attribute)
 5. Trigger
6. Package

Unit 3. Transaction Management	(10 Hours)
3.1 Transaction Concepts	
3.2 Transaction Properties	
3.3 Transaction States	
3.4 Concurrent Execution	
3.5 Serializability	
3.5.1 Conflict Serializability	
3.5.2 View Serializability	
3.6 Recoverability	
3.6.1 Recoverable Schedule	
Cascadless Schedule	
Unit 4. Concurrency Control	(10 Hours)
4.1 Lock Based Protocol	
4.1.1 Locks	
4.1.2 Granting of Locks	
4.1.3 Two Phase Locking Protocol	
4.2 Timestamp based Protocol	
4.2.1 Timestamp	
4.2.2 Timestamp ordering protocol	
4.2.3 Thomas's Write Rule	
4.3 Validation Based Protocol	
4.4 Deadlock Handling	
4.4.1 Deadlock Prevention	
4.4.2 Deadlock Detection	
4.4.3 Deadlock Recovery	
Unit 5. Recovery System	(10 Hours)
5.1 Failure Classification	
5.1.1. Transaction Failure	
5.1.2 System Crash	
5.1.3 Disk Failure	
5.2 Storage Structure	
5.2.1 Storage Type	
5.2.2 Data Access	
5.3 Recovery and Atomicity	
5.3.1 Log Based Recovery	
5.3.2 Deferred Database Modification	
5.3.3 Immediate Database Modification	
5.3.4 Checkpoints	
5.4 Recovery with Concurrent Transaction	
5.4.1 Transaction Rollback	
5.4.2 Restart Recovery	
5.5 Remote Backup System	

Suggested Readings:

1. Understanding of DBMS by Dr. B. W. Khalkar and Parthasarthy
2. Database System Concepts By Henry korth and A. Silberschatz
3. SQL, PL/SQL The Programming Language Oracle :- Ivan Bayross, BPB Publication.
4. Database Systems Concepts, Designs and Application by Shio Kumar Singh, Pearson

5. Introduction to SQL by Reck F. van der Lans by Pearson
6. Modern Database Management by Jeffery A Hoffer , V.Ramesh, Heikki Topi ,Pearson
7. Database Management Systems by Debabrata Sahoo ,Tata MacgrawHill

Title of the Course: Web Development with PHP								
Year: I				Semester: III				
Course Type	Course Code	Credit Distribution		Credits	Allotted Hours	Allotted Marks		
		Theory	Practical			CI	ES	Total
						E	E	
DSC-6	BBACA232T	03	00	03	45	30	70	100

Learning Objectives:

1. To develop the ability to logically plan and develop web pages.
2. To learn to write, test, and debug web pages using HTML and JavaScript.
3. Simple and impressive design techniques, from basics till advanced to focus on goal oriented and user centric designs.
4. How to and where to start research, planning for website & actually build excellent web sites.
5. To create web elements like buttons, banners & Bars and of course complete UI designs.
6. Forms and validations for your website.

Course Outcomes (Cos)

On completion of this course, students will be able to:

1. Understand how server-side programming works on the web.
2. Using PHP built-in functions and creating custom functions
3. Understanding POST and GET in form submission.
4. How to receive and process form submission data.
5. Read and process data in a MySQL database.

Unit I **PHP Basics** (5)

- 1.1 Setting up a development environment
- 1.2 Variables, numbers and strings
- 1.3 Calculations with PHP
- 1.4 Using Arrays

Unit II **Control Structures and Loops** (10)

Title of the Course: Data Structure Using 'C'								
Year: I				Semester: III				
Course Type	Course Code	Credit Distribution		Credits	Allotted Hours	Allotted Marks		
		Theory	Practical			CI	ES	Total
						E	E	
DSC-7	BBACA233T	03	00	03	45	30	70	100

Learning Objectives:

1. To provide the knowledge of basic data structures and their implementations
2. To understand concepts about searching and sorting techniques.
3. Enhance ability to analyze algorithms and algorithm correctness.
4. To understand various data structures like stack, queue and linked list operation.
5. To develop skills to apply appropriate data structures in problem solving.

Course Outcomes (Cos)

- 1: Understand basic data structures such as arrays, strings, and linked lists.
- 2: Study linear data structures such as stacks and queues and understand their difference.
- 3: Describe the hash function and concepts of collision and its resolution methods.
- 4: Understand the concept of memory management.
- 5: Study tree, heap and graphs along with their basic operations.
- 6: Study different techniques for solving problems like sorting and searching

Unit I: BASIC CONCEPTS (04)

1. Pointers and dynamic memory allocation
2. Algorithm-Definition and characteristics
3. Algorithm Analysis -Space Complexity -Time Complexity -Asymptotic Notation
4. Data structures: organizations of data(Contiguous and Non-Contiguous.)
5. Abstract Data Type (ADT)
6. Introduction to Arrays and Structure
7. Types of array and Representation of array
8. Self Referential Structure

Unit II: RECURSION (05)

- 2.1 Introduction to Recursion
- 2.2 Differences between recursion and iteration
- 2.3 Programs using Recursion
 - 2.3.1 Factorial of a given number
 - 2.3.2 The Towers of Hanoi
 - 2.3.3 Fibonacci Sequence Problem
 - 2.3.4 To calculate the NCR of a given number

Unit III: SEARCHING AND SORTING (07)

- 3.1 Linear Search
 - 3.1.1 A non-recursive program for Linear Search
 - 3.1.2 A Recursive program for linear search
- 3.2 Binary Search
 - 3.2.1 A non-recursive program for binary search
 - 3.2.2 A recursive program for binary search
- 3.3 Bubble Sort

- 3.3.1 Program for Bubble Sort
- 3.4 Selection Sort
 - 3.4.1 Non-recursive Program for selection sort
 - 3.4.2 Recursive Program for selection sort
- 3.5 Quick Sort
 - 3.5.1 Recursive program for Quick Sort

Unit IV: LINKED LISTS (08)

- 4.1. Linked List Concepts
- 4.2. Implementation of Linked List – Static & Dynamic representation
- 4.3. Types of Linked Lists
 - 4.3.1. Single Linked List (All type of operation)
 - 4.3.2. Double Linked List (All type of operation)
 - 4.3.3. Circular Single Linked List (Create and Display)
 - 4.3.4. Circular Double Linked List (Create and Display)
- 4.4. Polynomials
 - 4.4.1. Source code for polynomial creation with help of linked list
 - 4.4.2. Addition of Polynomials

Unit V: Stack and Queue (08)

- 5.1. Representation of Stack
 - 5.1.1. Program to demonstrate a stack, using array
 - 5.1.2. Program to demonstrate a stack, using linked list
- 5.2. Algebraic Expressions
 - 5.2.1. Converting expressions using Stack
 - 5.2.2. Conversion from infix to postfix
 - 5.2.3. Conversion from infix to prefix
 - 5.2.4. Conversion from postfix to infix
 - 5.2.5. Conversion from postfix to prefix
 - 5.2.6. Conversion from prefix to infix
 - 5.2.7. Conversion from prefix to postfix
 - 5.2.8. Evaluation of postfix expression
- 6. Applications of stacks
- 7. Representation of Queue
 - 7.1. Program to demonstrate a Queue (Static and Dynamic Implementation)
 - 7.2. Program to demonstrate a Queue using linked list
 - 7.3. Applications of Queue
 - 7.4. Types of Queue
 - 7.4.1. Circular Queue
 - 7.4.2. Deque
 - 7.4.3. Priority Queue

Unit VI: TREES (07)

- 6.1 Trees - Concept & Terminologies
- 6.2 Types of Tree data structure
 - 6.2.1 General Trees
 - 6.2.2 Binary Tree
- 6.3 Binary Tree Traversal Techniques
- 6.4 Binary Tree Creation and Traversal Using Arrays
- 6.5 Binary Tree Creation and Traversal Using Pointers
- 6.6 Non Recursive Traversal Algorithms
- 6.7 Recursive Traversal Algorithms

6.8 Binary Search Tree

- 6.8.1 Tree Traversals (preorder, inorder, postorder)
- 6.8.2 Application - Heap sort
- 6.8.3 Height balanced tree- AVL trees- Rotations, AVL tree examples.

Unit VII: GRAPHS

(06)

- 7.1 Introduction to Graphs (Concept & terminologies)
- 7.2 Representation of Graphs
- 7.3 Graph Representation – Adjacency matrix, adjacency list, inverse Adjacency list, adjacency multilist, orthogonal list
- 7.4 Degree of Graph
- 7.5 Traversals – BFS and DFS
- 7.6 Breadth first search and traversal
- 7.7 Depth first search and traversal

Suggested Readings:

1. Fundamentals of Data Structures ---- By Horowitz Sahani (Galgotia)
2. Data Structures using C and C++ --- By YedidyahLangsam, Aaron M. Tenenbaum, Moshe J. Augenstein
3. Introduction to Data Structures using C---By Ashok Kamthane
4. Data Structures using C --- Bandopadhyay&Dey (Pearson)
5. Data Structures using C ---By Srivastava BPB Publication.

Title of the Course: Practical (RDBMS)								
Year: I					Semester: III			
Course Type	Course Code	Credit Distribution		Credits	Allotted Hours	Allotted Marks		
		Theory	Practical			CI E	ES E	Total
SEC-3	BBACA234P	00	02	02	30	15	35	50

Assn No	WEEK	ASSIGNMENT
1.	First	Simple program in PL/SQL
2.	Second	Loops in PL/SQL with an example
3.	Third	Exception Handling – Predefined and User defined Exception programs
4.	Fourth	Cursor – Basic programs
5.	Fifth	Cursor – Implicit Cursor with an example
6.	Sixth	Cursor – Explicit Cursor with an example
7.	Seventh	Cursor – To solve the case study on cursor
8.	Eight	Trigger – Types with and example
9.	Ninth	Trigger – Types with an example
10.	Tenth	Trigger – To solve the case study on trigger
11.	Eleventh	Procedure – Programs with an example
12.	Twelfth	Procedure – To solve the case study of procedure
13.	Thirteenth	Function – programs with an example
14.	Fourteenth	Function – To solve the case study of function
15.	Fifteenth	Package – with an example
16.	Sixteenth	Package – To solve the case study of package

Title of the Course: Practical (PHP & DS)								
Year: I				Semester: III				
Course Type	Course Code	Credit Distribution		Credits	Allotted Hours	Allotted Marks		
		Theory	Practical			CI E	ES E	Total
FP-01	BBACA235P	00	02	02	30	15	35	50

Assignments: PHP Assignments

Assignment 1

- 1) Write a program for Currency notes cashier have to return
- 2) Tax on income (take input from user)
- 3) Profit and loss (take c.p. and s.p. from user)
- 4) find a entered year is leap year or not
- 5) find a number is palindrome or not
- 6) find a power of a number

Assignment 2

- 1) Find a entered number is armstrong or not
- 2) check whether entered number is prime or not
- 3) check whether entered number is perfect or not
- 4) find out the factorials of a given number
- 5) find sum of digits of a given number
- 6) count even and odd digits from given number
- 7) write a program to find fibonacci series upto n

Assignment 3

- 1) Write a program to count string length
- 2) write a program to count words from given string
- 3) Write a program to reverse a entered string
- 4) write a program to find largest between 5 given numbers
- 5) write a program to connect two strings entered by a user.

Assignment 4

- 1) Write a program to display a multiplication table of entered number.
- 2) Write a PHP program to display elements of an array along with the keys.
- 3) Write a PHP program to delete an element from an array from the given index
- 4) write a PHP program to perform addition of two matrix

5) write a PHP program to perform Multiplication of two matrix

Assignment 5

- 1) Write a program to find transpose of a given matrix
- 2) Write a program to convert a number from Binary to decimal
- 3) write a program to convert a number from decimal to binary
- 4) write a program to convert a number from decimal to octal
- 5) Write a program to convert a given string to uppercase.

Assignment 6

Q. print the following patterns

1)

```
*
**
***
****
*****
```

2)

```
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
```

3)

```
1
2 3
4 5 6
7 8 9 10
```

4)

A

```
B B
C C C
D D D D
E E E E E
```

5)

A

BC
DEF
GHIJ

Assignment 7

1)

1
1 1
1 1 1
1 1 1 1
1 1 1 1
1 1 1
1 1
1

2)

1
1 2
1 2 3
1 2 3 4
1 2 3 4 5

3)

5
5 4
5 4 3
5 4 3 2
5 4 3 2 1

5)

*
* * *
* * * * *
* * *
*

Assignment 8

1)

```
* * * * *
* * * *
* * *
* *
*
```

2)

```
      *
     * *
    * * *
   * * * *
  * * * * *
```

3)

```
0
0 1
0 1 0
0 1 0 1
0 1 0 1 0
0 1 0 1 0 1
```

4)

```
5 4 3 2 1
5 4 3 2
5 4 3
5 4
5
```

Assignment 9

1) Write a PHP script for the following: Design a form to accept a string. Write a function to count the total number of vowels (a,e,i,o,u) from the string. Show the occurrences of each vowel from the string. Check whether the given string is a palindrome or not, without using built-in function. (Use radio buttons and the concept of function).

2) Write a PHP script for the following: Design a form to accept two strings from the user. Find the first occurrence and the last occurrence of the small string in the large string. Also count the total number of occurrences of small string in the large string. Provide a text box to accept a string, which will replace the small string in the large string.

- 3) Write a PHP script for the following: Design a form to accept two numbers from the user. Give options to choose the arithmetic operation (use radio buttons). Display the result on the next form.

- 4) Write a PHP script for the following: Design a form to accept two strings from the user. Find whether the small string appears at the start of the large string. Provide a text box to accept the string that will replace all occurrences of small string present in the large string. Also split the large string into separate words. (Use regular expressions).

- 5) Write a PHP script for the following: Design a form to accept the details of 5 different items, such as item code, item name, units sold, rate. Display the bill in the tabular format. Use only 4 text boxes. (Hint : Use of explode function.)

Assignment 10

- 1) Write a PHP script for the following: Design a form to accept two strings. Compare the two strings using both methods (== operator & strcmp function). Append the second string to the first string. Accept the position from the user; from where the characters from the first string are reversed. (Use radio buttons).

- 2) Write a menu driven PHP program to perform the following operations on an associative array: i. Display the elements of an array along with the keys. ii. Display the size of an array iii. Delete an element from an array from the given index. iv. Reverse the order of each element's key-value pair.[Hint: use array_flip()] v. Traverse the elements in an array in random order [[Hint: use shuffle()].

- 3) Write a menu driven PHP program to perform the following operations on associative arrays: a) Sort the array by values (changing the keys) in ascending, descending order. b) Also sort the array by values without changing the keys. c) Filter the odd elements from an array. d) Sort the different arrays at a glance using single function. e) Merge the given arrays. f) Find the Union, intersection& set difference of two arrays.

- 4) Write a PHP script to define an interface which has methods area(), volume(). Define constant PI. Create a class cylinder which implements this interface and calculate area and volume.

- 5) Write class declarations and member function definitions for an

employee(code, name, designation). Design derived classes as emp_account (account_no, joining_date) from employee and emp_sal(basic_pay, earnings, deduction) from emp_account. Write a menu driven PHP program a) to build a master table b) to sort all entries c) to search an entry d) Display salary.

Assignment 11

1) Derive a class square from class Rectangle. Create one more class circle. Create an interface with only one method called area(). Implement this interface in all the classes. Include appropriate data members and constructors in all classes. Write a PHP program to accept details of a square, circle and rectangle and display the area.

2) Write PHP Script to create a class account (accno,cust_name). Derive two classes from account as saving_acc(balance, min_amount) and current_acc(balance, min_amount). Display a menu a) Saving Account b) Current Account For each of this displays a menu with the following options. 1. Create account 2. Deposit 3. Withdrawal.

3) Implement calculator to convert distances between (both ways) miles and kilometres. One mile is about 1.609 kilometres. User interface (distance.html) has one text-input, two radio-buttons, submit and reset -buttons. Values are posted to PHP-script (distance.php) which calculates the conversions according the user Input.

4) Using regular expressions check for the validity of the entered email-id. The @ symbol should not appear more than once. The dot (.) can appear at the most once before @ and at the most twice or at least once after @ symbol. The substring before @ should not begin with a digit or underscore or dot or @ or any another special character.

5) Write PHP program to create input form for Grocery that displays List of grocery items with checkboxes and create a bill according to list of items selected after clicking the submit button.

Assignment 12

1) Write a PHP program that accept customer name, consumer number and number of electricity units consumed from an input form and print electricity bill from following data · For first 50 units – Rs. 3.50/unit · For next

100 units – Rs.

4.00/unit · For next 100 units – Rs. 5.20/unit · For units above 250 –
Rs. 6.50/unit ·

Fixed meter and service charge- Rs. 150/-.

2) Write a PHP program for course registration of Learner in an institute that accept Name, Course to be admitted, Mobile number using input form validation such as Name should be only string of character, mobile number should contain digits with valid length and so on. and give feedback to Learner with registration details including registration number.

3) Write a PHP script to create a login form with a username and password. Once the user logs in, the second form should be displayed to accept user details (name, city, phone no). If the user doesn't enter information within a specified time limit, expire his session and give a warning.

4) Write a PHP script to keep track of number of times the web page has been Accessed.

5) Write a PHP script to change the preferences of your web page like font style, font size, font color, background color using cookies. Display selected settings on next web page and actual implementation (with new settings) on the third web page.

Assignment 13

1) Write a PHP script to create a form to accept student information (name, class, address). Once the student information is accepted, accept marks in next form (Phy, Bio, Chem, Maths, Marathi, English). Display the mark sheet for the student in the next form containing name, class, marks of the subject, total and Percentage.

2) Write a PHP program to create a shopping mall UI. User must be allowed to do purchase from two pages. Each page should have a page total. The third page should display a bill, which consists of a page total of whatever the purchase has been done and printed the total. (Use http session tracking).

3) Write a PHP script to create a form to accept customer information (name, address, ph no). Once the customer information is accepted, accept product

information in the next form (Product name, qty, rate). Display the bill for the customer in the next form. Bill should contain the customer information and the information of the products entered.

- 4) Write a PHP script to accept username and password. If in the first three chances, username and password entered is correct, then display second form, otherwise display an error message.

- 5) Consider the following entities and their relationships Emp (emp_no, emp_name, address, phone, salary) Salary (emp_no, Basic, HR, TA, DA) Dept (dept_no, dept_name, location) Emp-Dept are related with one-many relationship Create a RDB for the above and solve the following Using the above database. write a PHP script which will print a salary statement for specified emp_no with his details.

Assignment 14

- 1) Consider the following entities and their relationships Doctor (doc_no, doc_name, address, city, area) Hospital (hosp_no, hosp_name, hosp_city) Doctor and hospitals are related with many-many relationships. Create a RDB in 3 NF for the above and solve following Using above database, write a PHP script which accepts hospital name and print information about doctors visiting / working in that hospital in tabular format.

- 2) Consider the following entities and their relationships project (pno integer, p_name char(30), p_type char(20), duration integer), employee (eno integer, e_name char(20), qualification char(15), join_date date). The relationship between project - employee: M-M, with descriptive attributes as start_date (date), no_of_hours_worked (integer). Using above database write a script in PHP to accept a project name from user and display information of employees working on the project.

- 3) Consider the following entities and their relationships student (sno integer, s_name char(30), s_class char(10), s_addr char(50)), teacher (tno integer, t_name char(20), qualification char(15), experience integer). The relationship between student-teacher: m-m with descriptive attribute subject. Using above database write a script in PHP to accept a teacher name from user and display the names of students along with subjects to whom the teacher is teaching.

- 4) Consider the following entities and their relationships Movie (movie_no, movie_name, release_year) Actor (actor_no, name) Relationship between movie

and actors are many – many with attribute rate in Rs. Create a RDB in 3NF for the above and solve following Using above database, write PHP scripts for the following:(Hint: Create HTML form having three radio buttons) a) Accept actor name and display the names of the movies in which he has acted. b) Insert new movie information. c) Update the release year of a movie. (Accept the movie name from user).

5) Considerer the following entities and their relationships Student (Stud_id,name,class) Competition (c_no,c_name,type) Relationship between student and competition is many-many with attribute rank and year. Create a RDB in 3NF for the above and solve the following. Using above database write a script in PHP to accept a competition name from user and display information of student who has secured 1st rank in that competition.

Subject : Data Structure

Practical 1:

1. Write a program in C to find the sum of all elements of the array using fuction.
2. Write a program in C to find the maximum and minimum element in an array using function.
3. Write a program in C to sort elements of array in ascending and descending order using function.
4. Write a program in C to swap two numbers using function.(Call by value)
5. Write a program in C to demonstrate the use of &(address of) and *(value at address) operator.
6. Write a program in C to swap elements using call by reference.
7. Write a program in C to sort an array using Pointer.
8. Write a program in C to show the usage of pointer to structure.

Practical 2:

1. Write a program in C to show the usage of pointer to structure.
2. Write a program in C to calculate the sum of numbers from 1 to n using recursion.
3. Write a program in C to Print Fibonacci Series using recursion.
4. Write a program in C to find the sum of digits of a number using recursion
5. Write a program in C to find the Factorial of a number using recursion.

Practical 3:

1. Write a C program to sort a list of elements using the selection sort algorithm.
2. Write a C program to sort a list of elements using the bubble sort algorithm

3. Write a C program to sort a list of elements using the insertion sort algorithm.
4. Write a C program to sort a list of elements using the merge sort algorithm.
5. Write a C program to sort a list of elements using the quick sort algorithm.

Practical 4:

6. Write a non-recursive program for Linear Search
7. Write a Recursive program for linear search
8. Write a non-recursive program for binary search
9. Write a recursive program for binary search

Practical 5:

1. Write a program in C to create and display Singly Linked List.
2. Write a program in C to insert a new node at the beginning of a Singly Linked List.
3. Write a program in C to insert a new node at the end of a Singly Linked List.
4. Write a program in C to insert a new node at the middle of Singly Linked List.
5. Write a program in C to delete first node of Singly Linked List.

Practical 6:

6. Write a program in C to delete a node from the middle of Singly Linked List.
7. Write a program in C to search an existing element in a singly linked list
8. Write a program in C to create a doubly linked list and display in reverse order.
9. Write a program in C to insert a new node at any position in a doubly linked list.
10. Write a program in C to delete a node from the beginning of a doubly linked list.

Practical 7:

11. Write a program in C to delete a node from any position of a doubly linked list
12. Write a program in C to create and display a circular linked list.
13. Write a program in C to insert a node at the end of a circular linked list.
14. Write a program in C to delete the node at the end of a circular linked list
15. Write a program in C to search an element in a circular linked list.

Practical 8:

1. Write a program for representation of Stack(PUSH,POP)
2. Write a program to demonstrate a stack, using array
3. Write a program to demonstrate a stack, using linked list
4. Write a program to Conversion from infix to postfix
5. Write a program to Conversion from infix to prefix

Practical 9:

1. Write a program to Conversion from postfix to infix
2. Write a program to Conversion from postfix to prefix
3. Write a program to Conversion from prefix to infix
4. Write a program to Conversion from prefix to postfix
5. Write a program to Evaluation of postfix expression
6. Write a program to Representation of Queue
7. Write a program to Program to demonstrate a Queue (Static and Dynamic Implementation)
8. Write a program to Program to demonstrate a Queue using linked list
9. Write a program to implementation of Circular Queue

Practical 10:

1. Write a 'C' program to create a Binary tree, traverse it using recursive operations like inorder, preorder and postorder and display the result of each one separately.
2. Write a 'C' program to read 'n' integers and store them in a binary tree structure and count the following and display it.
 - Number of nodes
 - Degree of tree
 - Leaf nodes
3. Write menu driven program using 'C' for Binary Search Tree. The menu includes
Create a BST
Insert element in a BST
Display
4. Write a 'C' program to count indegree and outdegree of each node in graph.

Practical 11:

5. Write a 'C' program to read the adjacency matrix of directed graph and convert it into adjacency list.
6. Write a 'C' program to read an adjacency matrix of a directed graph and traverse it using DFS.
7. Write menu driven program using 'C' for Binary Search Tree. The menu includes -
Create a BST
Traverse it by using Inorder traversing technique
Search a given value in BST
8. Write a 'C' program to read 'n' integers and store them in a Binary search tree and display the nodes level wise.

SEMESTER IV

Title of the Course: Object Oriented Programming Using C++								
Year: I				Semester: III				
Course Type	Course Code	Credit Distribution		Credits	Allotted Hours	Allotted Marks		
		Theory	Practical			CI E	ES E	Total
DSC-8	BBACA241T	03	00	03	45	30	70	100

Learning Objectives:

- To understand how C++ improves C with object-oriented features.
- To learn how to write inline functions for efficiency and performance.
- To learn the syntax and semantics of the C++ programming language.
- To learn how to design C++ classes for code reuse.
- To learn how to implement copy constructors and class member functions.
- To understand the concept of data abstraction and encapsulation.
- To learn how to overload functions and operators in C++.
- To learn how containment and inheritance promote code reuse in C++.
- To learn how inheritance and virtual functions implement dynamic binding with polymorphism.
- To learn how to design and implement generic classes with C++ templates.
- To learn how to use exception handling in C++ programs.

Course Outcomes:

1. Acquire an understanding of basic object-oriented concepts and the issues involved in effective class design.
2. Enable students to write programs using C++ features like operator overloading, constructor and destructor, inheritance, polymorphism and exception handling.

Unit I: Introduction to C++ (04)

- 1.1 Basic concepts, features, advantages and applications of OOP
- 1.2 Introduction, applications and features of C++
- 1.3 Input and Output operator in C++
- 1.4 Simple C++ program

Unit II: Beginning with C++ (08)

- 2.1 Data type and Keywords
- 2.2 Declaration of variables, dynamic initialization of variables, reference variable
- 2.3 Operators:
 - 2.3.1 Scope resolution operator
 - 2.3.2 Memory management operators
- 2.4 Manipulators
- 2.5 Functions:
 - 2.5.1 Function prototyping, call by reference and return by reference
 - 2.5.2 Inline functions
- 2.6 Default arguments

Unit III: Classes and Objects (08)

- 3.1 Structure and class, Class, Object
- 3.2 Access specifiers, defining data member
- 3.3 Defining member functions inside and outside class definition.
- 3.4 Simple C++ program using class
- 3.5 Memory allocation for objects
- 3.6 Static data members and static member functions
- 3.7 Array of objects, objects as a function argument
- 3.8 Friend function and Friend class
- 3.9 Function returning objects

Unit IV: Constructors and Destructors (08)

- 4.1 Constructors
- 4.2 Types of constructor: Default, Parameterized, Copy
- 4.3 Multiple constructors in a class
- 4.4 Constructors with default argument
- 4.5 Dynamic initialization of constructor
- 4.6 Dynamic constructor
- 4.7 Destructor

Unit V: Inheritance (08)

- 6.1 Introduction
- 6.2 Defining Base class and Derived class
- 6.3 Types of Inheritance
- 6.4 Virtual Base Class
- 6.5 Abstract class
- 6.6 Constructors in derived class

Unit VI: Polymorphism (08)

- 7.1 Compile Time Polymorphism
 - 7.1.1 Introduction, rules for overloading operators
 - 7.1.2 Function overloading
 - 7.1.3 Operator Overloading unary and binary
 - 7.1.4 Operator Overloading using friend function
 - 7.1.5 Overloading insertion and extraction operators
 - 7.1.6 String manipulation using operator overloading
- 7.2 Runtime Polymorphism
 - 7.2.1 this Pointer, pointers to objects, pointer to derived classes
 - 7.2.2 Virtual functions and pure virtual functions

Unit VII: Managing Console, I/O operations (04)

- 8.1 C++ streams and C++ stream classes
- 8.2 Unformatted I/O operations
- 8.3 Formatted console I/Operations
- 8.4 Output formatting using manipulators
- 8.5 User defined manipulators

Unit VIII: Working with Files (08)

- 9.1 Stream Classes for File operations
- 9.2 File operations - Opening, Closing and updating
- 9.3 File updating with random access.
- 9.4 Error handling during File operations
- 9.5 Command Line arguments

Unit IX: Templates (04)

- 10.1 Introduction
- 10.2 Class Template and class template with multiple parameters
- 10.3 Function Template and function template with multiple parameter
- 10.4 Exception Handling Introduction.

Suggested Readings:

1. Object Oriented programming with C++ by E Balagurusamy
2. Object Oriented Programming with C++ by Robert Lafore
3. The Complete Reference C++ by Herbert Schildt

Title of the Course: Advanced Web Development								
Year: I				Semester: III				
Course Type	Course Code	Credit Distribution		Credits	Allotted Hours	Allotted Marks		
		Theory	Practical			CI E	ES E	Total
DSC-9	BBACA242T	03	00	03	45	30	70	100

Learning Objectives:

- Understand client server architecture and able to use the skills for web project development
- Create job opportunities as a web developer..

Course Outcomes (Cos)

On completion of this course, students will be able to:

1. To know & understand concepts of internet programming.
2. Understand how server-side programming works on the web.
3. Understanding How to use PHP Framework (Joomla / Druple)

Unit I Introduction to Object Oriented Programming in PHP (08)

- 2.1 Server information
- 2.2 Processing forms
- 2.3 Sticky forms
- 2.4 Setting response headers

Unit I I Web Techniques (06)

- 2.1 Server information
- 2.2 Processing forms
- 2.3 Sticky forms
- 2.4 Setting response headers

Unit I II XML (10)

- 3.1 Introduction XML
- 3.2 XML document Structure
- 3.3 PHP and XML
- 3.4 XML parser
- 3.5 The document object model
- 3.6 The simple XML extension
- 3.7 Changing a value with simple XML

Unit I V Ajax with PHP (08)

- 4.1 Understanding java scripts for AJAX
- 4.2 AJAX web application model
- 4.3 AJAX –PHP framework
- 4.4 Performing AJAX validation
- 4.5 Handling XML data using php and AJAX
- 4.6 Connecting database using php and AJAX

Unit V Introduction to Web Services (12)

- 5.1 Definition of web services
- 5.2 Basic operational model of web services, tools and technologies enabling web services
- 5.3 Benefits and challenges of using web services.
- 5.4 Web services Architecture and its characteristics
- 5.5 Core building blocks of web services
- 5.6 Standards and technologies available for implementing web services
- 5.7 Web services communication models
- 5.8 Basic steps of implementing web services.

Unit VI PHP Framework (Joomla / Druple)

(16)

- 6.1 Introduction to Joomla/Druple
 - 6.1.1 Introduction
 - 6.1.2 Joomla/Druple features
 - 6.1.3 How Joomla/Druple works ?
 - 6.1.4 The platform Components, Modules and Plugins
- 6.2 Administering Joomla/Druple
 - 6.2.1 Presentation Administration
 - 6.2.2 Content Administration
 - 6.2.3 System Administration
- 6.3 Working with Joomla/Druple
 - 6.3.1 Adding articles
 - 6.3.2 Adding menus to point to content
 - 6.3.3 Installing new templates
 - 6.3.4 Creating templates
 - 6.3.5 Adding a Module and Component
 - 6.3.6 Modifying the existing templates
 - 6.3.7 Creating templates with web editors

Suggested reading :

Php: A Beginner's Guide 1st Edition McGraw-Hill Osborne Media; 1 edition by Vikram Vaswani

Murach's PHP and MySQL (2nd Edition) by Joel Murach and Ray Harris

PHP: The Complete Reference Paperback – 1 Jul 2017 by Steven Holzner (Author)

Building Web Services with Java, 2nd Edition, S. Graham and others, Pearson Edn., 2008..

Title of the Course: Software Engineering								
Year: I				Semester: III				
Course Type	Course Code	Credit Distribution		Credits	Allotted Hours	Allotted Marks		
		Theory	Practical			CI E	ES E	Total
DSC-10	BBACA243T	02	00	02	30	15	35	50

Learning Objectives:

- Apply information technology principles and practices to real-world solutions
- Demonstrate effective use of written, verbal, and non-verbal communication, employing relevant knowledge, skills, and judgment in a business setting
- Manage a simple project and be able to contribute to a more complex project as a team member
- Work as a professional maintaining high standards of practice, making ethical/legal judgments and decisions, and sustaining a professional standing through a commitment to life-long learning
- Develop and apply personal management and team member skills as a professional software developer

Course Outcomes:

1. To understand System concepts.
2. To understand Software Engineering concepts.
3. To understand the applications of Software Engineering concepts and Design in Software development.

Unit I: Introduction to System Concepts (03)

1. Definition & Characteristic of System
2. Basic Components of the System
3. Types of System

Unit II: Introduction to Software Engineering (05)

1. Definition & Characteristics of Software
2. Definition of Software Engineering
3. Need for Software Engineering
4. Mc Call's Quality factors
5. Software Product and Process
6. V & V Model

Unit III: Software Development Life Cycle (06)

1. Introduction
2. SDLC
3. Waterfall Model
4. Incremental Process Models
5. Prototyping Model
6. Spiral Model

Unit IV: Requirement Engineering (06)

1. Introduction
2. Requirement Elicitation
3. Requirement Elaboration
4. Requirement Gathering
5. Feasibility study
6. Fact Finding Techniques

7. SRS Format

Unit V: Analysis And Design Tools (08)

1. Decision Tree and Decision Table
2. Data Flow Diagrams (DFD) (Up to 2nd level)
3. Data Dictionary
4. Input and Output Design
5. Structured Design Concepts
6. Structure Chart
7. Coupling and Cohesion
8. Three Case Studies on above topics

Unit VI: Software Maintenance and Software Re-Engineering (02)

1. Maintenance definition and types
1. Software reengineering
2. Reverse Engineering

Suggested Readings:

1. Software Engineering: A Practitioner's Approach- Roger S. Pressman, McGraw hill International Editions 2010(Seventh Edition)
2. System Analysis, Design and Introduction to Software Engineering (SADSE) - S. Parthsarthy, B.W. Khalkar
3. Analysis and Design of Information Systems(Second Edition) - James A. Senn, McGraw Hill
4. System Analysis and Design- Elias Awad, Galgotia Publication, Second Edition

Title of the Course: Practical (C++)								
Year: I				Semester: III				
Course Type	Course Code	Credit Distribution		Credits	Allotted Hours	Allotted Marks		
		Theory	Practical			CI E	ES E	Total
VSC-2	BBACA244P	00	02	02	30	15	35	50

Ass no	Week	Assignment
1.	First	First CPP Program - Print Hello World Program for Use of cin and cout Program of Arithmetic Operations Program to find Largest and Smallest Between Numbers Program using conditional operators and Manipulators.
2.	Second	Programs Using Control Structure, Loops. Program to Accept Marks in Subject and Print Total and Percentage. Program for the illustration of switch case Program of finding Maximum Number, Product of Digits, Prime, Factorial, Sum of Digits, Palindrome, Armstrong using Loop Pattern Programs.
3.	Third	Programs of Function, Function Declaration, Definition, Inline Function - to calculate square and cube of number and Inline function for arithmetic operations. Friend Function - to check number is even or odd, to calculate multiplication of two numbers using forward declaration, Function to check a given number is : Armstrong or not, Palindrome or not, Prime or not, Perfect or not, Leap Year or not, etc.
4.	Fourth	C Class and Objects, Declare, Define and Access Data Members and Member Functions, C++ Program Using Class to accept and display Student data calculate Marks, Employee Data and calculate DA, HRA, Gross Salary.
5.	Fifth	Program to define function outside the class definition using scope resolution operator. Program to demonstrate Static Data Members and Member Function
6.	Sixth	Programs of Constructor - Default Constructor - Calculate area of Circle, Square, Rectangle, Roots of Quadratic Equation. Parameterized Constructor - Check Leap Year, Calculate Base and Power, Overloaded Constructor for the addition of numbers Program for the demonstration of Copy Constructor

7.	Seventh	Programs of Constructor with Default Arguments, Dynamic Constructor, Multiple Constructor in class, Destructor
8.	Eighth	Programs of Inheritance - Demonstration of Single Inheritance - Accept Number and perform Mathematical Operations, Demonstration of Multilevel Inheritance - get Child, Father and GrandFather details and display it, Demonstration of Multiple Inheritance - Accept data of Teaching and Non Teaching and derive in Record class
9.	Ninth	Programs of Hierarchical Inheritance - Accept Base class Fields and derive classes as Arts and Science, accept and display Data Hybrid Inheritance - Program Using Student class Regular Subject Marks and Add On Courses Marks, Calculate and Display Result. Virtual Base Class, Abstract Class, Constructor in Derived Class
10.	Tenth	Programs of Polymorphism - Function Overloading - Overloaded function to calculate Area, Perimeter, Volume, Addition, Multiplication, Operator Overloading - Overloading Unary and Binary Operators, Operator Overloading using Friend Function Program for the illustration of this pointer
11.	Eleven	Programs of Managing Console Input/Output Operations, Unformatted Input/Output Operations - programs using get() function, getline() function, put() function, read() function, write function(). Formatted Input/Output Operations - programs using width() function, precision() function, fill() function, setf() function, unsetf() function
12.	Twelve	Programs of File Handling and File Operations - Open File Close File Read data from file write data into file Deleting Data from file Detecting end of file Update file
13.	Thirteen	Programs of Exception Handling using try, catch, throw, throws. Program using User defined exception
14.	Fourteen	Programs of Function Template Programs of Class Template

Title of the Course: Practical (AWT)								
Year: I				Semester: III				
Course Type	Course Code	Credit Distribution		Credits	Allotted Hours	Allotted Marks		
		Theory	Practical			CI E	ES E	Total
CEP-01	BBACA245P	00	02	02	30	15	35	50

Assignments: Advance PHP

Assignment 1

- 1) Write a PHP script to create a simple calculator that can accept two numbers and perform operations like addition, subtraction, multiplication etc.
- 2) Write a php script to create a student.xml file which contains student rollno, name, address, college and course. Print student details of specific courses in tabular format after accepting the course as input.
- 3) Write a simple PHP program which implements Ajax for subtraction of two numbers.
- 4) Write a script to solve following questions (Use "Student.xml" file)
 1. Create a DOM Document object and load this XML file.
 2. Get the output of this Document to the browser.
 3. Save this [. XML] document in another format i.e. in [.doc]

Write a XML Script to print the names of the students present in the "Student.xml "
- 5) Write a Calculator class that can accept two values, then add, subtract, multiply them or divide them on request.

Assignment 2

- 1) Write a script to create "cricket.xml" file with multiple elements as shown below:

```
<CricketTeam>
<Team country="Australia">
<player> ----- <runs>
</player>-----</ runs>
<wicket>-----</wicket>
</Team>
</CricketTeam >
```

Write a script to add multiple elements in the "cricket.xml" file of category, country=" India".

- 2) Define a class Employee having private members - id, name, department, salary. Define parameterized constructors. Create a subclass called "Manager"

with a private member bonus. Create 3 objects of the Manager class and display the details of the manager having the maximum total salary (salary + bonus).

3) Create an xml file which should comprise the following:

```
<cricket>
<player>Sachin Tendulkar</ player>
<runs>2000</runs>
<wickets> I00 </ wickets>
<noofnotout>20</noofnotout>
</cricket> For at least 5 players.
```

Write a PHP script to display the details of players who have scored more than 1200 runs and at least 50 wickets.

4) Create an abstract class Shape with methods area() and volume(). Derive three classes: rectangle (length, breath), Circle(radius) and Cylinder(radius, height) , Calculate area and volume of all. (Use Method overriding).

5) Create student table as follows: Student(sno, sname, standard, Marks, per).Write AJAX script to select the student name and print the student's details of a particular standard.

Assignment 3

1) Write a PHP script, which will return the following component of the URL (Using response header)http://www.college.com/Science/CS. php

2) Create an employee table as follows: EMP (eno, ename, designation, salary). Write an Ajax program to select the employees name and print the selected employee' s details.

3) Consider the following relational database: Project (P_Group No, Project_Tiltle)

Student (Sea t_no, Name, Class, P_Group_No)

Write an AJAX script to accept project titles and display a list of students who are working on a particular project.

4) Write a PHP Script to create class Shape and its subclass Triangle, Square, Circle and display area of selected shape (use concept of inheritance).

5) Write an Ajax script to get player details from an XML file when the user selects the player name. Create an XML file to store details of the player (name, country, wickets and runs).

Assignment 4

1) Write a PHP program to create a Class Calculator which will accept two values from user and pass as an argument through parameterized constructor and do the following task:

- a) Add
 - b) Subtract
 - c) Multiply them together or divide them on request.
- 2) Consider the following entities and their relationships Movie(movie no,movie_name,release_year) Actor(actor no,name) Relationship between movie and actor is many-many with attribute rate in Rs. Create a RDB in 3 NF. With using three radio buttons (accept, insert, update) Write an AJAX script to accept actor names and display names of movies in which he has acted.
- 3) Write a PHP Script to demonstrate the concept of Introspection for examining objects.(Using any 3 predefined functions)
- 4) Write a PHP script to perform the following stack related operations-insert, display.(Use concept of self processing form)
- 5) Write a PHP script to perform the following stack related operations-delete, display.(Use concept of self processing form)

Assignment 5

- 1) Write a PHP program to create a class circle having a radius data member and two member functions find_circumference() and find_area(). Display area and circumference depending on user's preference.
- 2) Write an Ajax code to print the content of "Myfile.dat" on clicking on fetch Button. The Data fetches from the server using Ajax Technique.
- 3) Write a PHP program to convert temperature Fahrenheit to Celsius using sticky form.
- 4) Write a PHP program to convert temperature Celsius to Fahrenheit using sticky form.
- 5) Write an AJAX script to read the contact.dat file and print the content of a file in a tabular form when the user clicks on the print button. Contact.dat file contains srno, name, residence number, mobile number (Enter at least 3 records in contact.dat file)

Assignment 6

- 1) Create a form to accept Employee details and display it in the next page. (Use sticky form concept).
- 2) Create a web Application that contains Voters details and check proper validation for (name, age, and nationality), as Name should be in upper letters only, Age should not be less than 18 yrs and Nationality should be Indian.(use HTML-AJAX-PHP)
- 3) Write a PHP script to accept a string from user and then display the accepted string in reverse order.(use concept of self processing form)
- 4) Write a PHP script to accept a string from user and then display the accepted string in Uppercase.(use concept of self processing form)

- 5) Write a PHP script using AJAX concept, to check if username and passwords are valid or Invalid (use a database to store username and password).

Assignment 7

- 1) Write a PHP program to select a list of subjects from the list box and display the selected subject information on the next page. (Use sticky Multi Value parameter).
- 2) Write a PHP script using AJAX concept, to give hints to the user when he/she types a city name in the text field.
- 3) Write an Ajax program to carry out validation for a username entered in the textbox. If the textbox is blank, print ' Enter username ' . If the number of characters is less than three, print ' Username is too short ' . If the value entered is appropriate the print ' Valid username ' .
- 4) Write a PHP Script to display Server information in table format (Use \$_SERVER).
- 5) Write a script to create an XML file "University.xml". The elements details of "University.xml" are as follows:

```
<Univ> <Uname>----- </Uname>
<City>----- </City>
<Rank>----- </Rank>
</Univ>
```

- 1) Store the details of at least 3 universities.
- 2) Link the "University.xml" file to css and get well formatted output as given below,...
 - i) Uname: Color : black;
Font-family: copperplate Gothic Light;
Font-size: 16 pt
Font: Bold;
 - ii) City and Rank : Color : Yellow;
Font-family: Arial;
Font-size: 12 pt
Font: Bold;

Assignment 8

- 1) Write a PHP program to create a simple distance calculator that can accept distance in meters from the user. Convert it into centimeters or kilometers according to user preference. (use radio buttons and Self Processing form)
- 2) Write a PHP script to generate an XML in the following format.


```
<?xml version = " 1.0" ?>
<BookStore>
<Books>
<PHP>
```



```

<title>Programming PHP</title>
<publication>O 'RELLY</ publication>
<price>800</price>
</PHP>
<PHP>

```

```

<title>Beginners PHP</title>
<publication>WROX</publication>
<price>900</price>
</PHP>
</Books>
</BookStore>

```

Add more than 5 book details . Display details of the book by selecting the publication of the book from the user.

3) Write an Ajax program to get book details from an XML file when the user selects a book name. Create an XML file to store details of the book(name, author, year and price).

4) Create a XML file which gives details of movies available in "Venus CD Store" from following categories a) Classical b) Action c) Horror Elements in each category are in the following format

```

<Category>
<Movie Name>----- </Movie Name>
<Release Year>----- </Release Year>
<Actor Name>----- </Actor Name>
</Category>

```

Save the file with the name "movies.xml".

5) Write a simple PHP program which implements Ajax to check armstrong number.

Assignment 9

1) Write a script to create XML file named "Course.xml"

```

<Course>
<Computer Science>
<Student name>..... </Student name>
<Class name >..... </Class name>
<percentage>..... </percentage>
</Computer Science>
</Course>

```

Store the details of 5 students who are in SYBBA(CA).

2) Create an XML file which gives details of books available in " Pragati Bookstore" from the following categories.

1) Yoga 2) Story 3) Technical
and elements in each category are in the following format
<Book>

```
<Book Title> -----</Book Title>
<Book Author> ----- </Book Author>
<Book Price> ----- </Book Price>
```

```
</Book>
```

Save the file as " Bookcategory.xml"

3) Create an application that reads a "Sports.xml" file into a simple XML object. Display attributes and elements. (Hint: Use simple_xml_load_fileO function)

4) Write Ajax program to print Movie details by selecting an Actor's name. Create table MOVIE and ACTOR as follows with 1 : M cardinality MOVIE (mno, mname, release_yr) and ACTOR(ano, aname).

5) Write a script to create XML file named "College.xml" The element details of "College.xml" are:

```
<College >
<Cname>----- </Cname>
<City>----- </City>
<Rank>----- </Rank>
</College>Display at least 3 college Information.
```

Assignment 10

1) Write a script to create "vehicle.xml" file with multiple elements as given below <Vehicle>

```
<Type = Two Wheeler>
<Vehicler Name >----- </vehicle Name >
<Company >----- </Company>
<Color>----- </Color>
<Average>----- </Average>
</Type>
<N vehicle>
```

Also add Type = " Four Wheeler" and its elements

2) Write a PHP program to implement Create, Read, Update and Display operations on the Teacher table with attributes (tid, tname, address, subject). (UseRadioButtons)

3) Write a simple PHP program which implements Ajax for addition of two numbers.

4) Write a simple PHP program which implements Ajax for Factorial of a number.

5) Write a PHP script to read book.XML and print book details in tabular format using simple XML(Content of book.XML are book_code, book_name, author, year ,price).

Assignment 11

- 1) Write an Ajax program to display a list of games stored in an array on clicking the OK button.
- 2) Write a PHP script using AJAX concept, to develop user-friendly and interactive search engine (like a Google search engine)
- 3) Write a PHP Script to create a superclass Vehicle having members Company and price. Derive 2 different classes:
LightMotorVehicle (members - mileage)
HeavyMotorVehicle (members - capacity-in-tons).
- 4) Derive a class Rectangle from class Square. Create one more class Triangle. Create an interface with only one method called cal_area (). Implement this interface in all the classes. Include appropriate data members and constructors in all classes. Write a program to accept details of a Rectangle, Square and Triangle and display the area.
- 5) Write a simple PHP program which implements Ajax for multiplication of two numbers.

Assignment 12

- 1) write a Ajax program to display list of games stored in an array on clicking OK button
- 2) Create a student table. Write an AJAX program to select the student name and print the details of the selected student.
- 3) Write a PHP script to demonstrate the introspection for examining classes and objects.
- 4) Write a simple PHP program which implements Ajax to find leap year.
- 5) Write a PHP script for the following: Design a form to accept a number from the user. Perform the operations and show the results.
 - 1) Fibonacci Series.
 - 2) To find the sum of the digits of that number.
 (Use the concept of a self processing page.)

Assignment 13

- 1) Write a PHP Script to create a superclass School having members Students and Teachers. Derive 2 different classes from them.
- 2) Write a simple PHP program which implements Ajax to find fibonacci series.
- 3) Write a PHP program to select a list of students from the list box and display the selected students' information on the next page. (Use sticky Multi Value parameter).
- 4) Write a PHP script to accept a string from user and then display the

accepted string in Lowercase.(use concept of self processing form)

5) Write a PHP script using AJAX concept, to check if username and passwords are valid or Invalid (use a database to store username and password).
If valid then show users details(name,age,mobile num)

Assignment 14

- 1) Define a class Student having private members - id, name, department. Define parameterized constructors. Create a subclass called "Marks". Create 3 objects of the Marks class and display the details of the student having the maximum marks.
- 2) Write a php script to create a emp.xml file which contains emp id, name, address, designation and salary. Print employee details of specific designation in tabular format after accepting the designation as input.
- 3) Write an Ajax program to get emp details from an XML file when the user selects a emp name. Create an XML file to store details of the emp(name, dept, dob and salary).
- 4) Create a form to accept student details and display it in the next page. (Use sticky form concept).
- 5) Write a simple PHP program which implements Ajax to check palindrome.