



**MES Garware College of Commerce, Pune, India  
(Autonomous)**

**Affiliated to  
Savitribai Phule Pune University, Pune**

**AUTONOMY HANDBOOK**

**Choice Based Credit System - CBCS  
(2021 Pattern)  
With effect from Academic Year 2022-23**

**Degree Programme of  
Bachelor of Business Administration – Computer Application (BBA-CA)**

**Course Contents**

<b>Sr. No.</b>	<b>Course Code</b>	<b>Name of the Course (Paper / Subject)</b>	<b>Pg. No.</b>
1	B4-21/401	C++ programming	3
2	B4-21/402	Object Oriented Software Engineering	9
3	B4-21/403A	Dot Net Programming	13
3	B4-21/403B	Python Programming	18
4	B4-21/404A	Big Data	22
5	B4-21/404B	Block Chain	26
6	B4-21/405	Project based on HTML, CSS, JS	32
7	B4-21/407	jQuery	34

## Course Contents

<b>Course Code:</b> <b>B4-21/401</b>	<b>Subject / Course: C++ programming</b>	<b>Marks: 100</b> <b>Credits: 3</b>
<p><b>Course Objectives:</b></p> <ol style="list-style-type: none"> <li>1. Acquire an understanding of basic object-oriented concepts and the issues involved in effective class design.</li> <li>2. Enable students to write programs using C++ features like operator overloading, constructor and destructor.</li> <li>3. To understand managing console I/O operations in C++.</li> <li>4. Enable students to write programs using C++: inheritance, polymorphism and exception handling.</li> </ol>		
<p><b>Course Outcome:</b></p> <p>After completing the course, the student shall be able to</p> <p><b>CO1:</b> Ability to visualize the representation of object-oriented concepts.</p> <p><b>CO2:</b> Practical Implementation of constructor, inline function, friend function concepts of C++ Language.</p> <p><b>CO3:</b> Basic knowledge of input, output operations and practical implementation in coding.</p> <p><b>CO4:</b> Practical implementation of inheritance, polymorphism and exception handling.</p>		

Unit	Unit Title	Contents	No. of Lectures
I	Introduction to C++	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	10
II	Beginning with C++	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	9

Unit	Unit Title	Contents	No. of Lectures
		2.5 Functions: 2.5.1 Function prototyping, call by reference and return by reference 2.5.2 Inline functions 2.6 Default arguments	
III	Classes and Objects	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	7
IV	Constructors and Destructors	4.1 Constructors 4.2 Types of constructors: 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	7
V	Inheritance	5.1 Introduction 5.2 Defining Base class and Derived class 5.3 Types of Inheritance 5.4 Virtual Base Class 5.5 Abstract class 5.6 Constructors in derived class	7
VI	Polymorphism	6.1 Compile Time Polymorphism 6.1.1 Introduction, rules for overloading operators 6.1.2 Function overloading 6.1.3 Operator Overloading unary and binary 6.1.4 Operator Overloading using friend function 6.1.5 Overloading insertion and extraction	8

Unit	Unit Title	Contents	No. of Lectures
		operators 6.1.6 String manipulation using operator overloading 6.2 Runtime Polymorphism 6.2.1 this Pointer, pointers to objects, pointer to derived classes 6.2.2 Virtual functions and pure virtual functions	
<b>Total No of Lectures</b>			<b>48</b>

**Teaching Methodology:**

Unit	Unit Title	Suggestive teaching methodology	Practical	Outcome expected		Weightage of Marks (%)
				Conceptual understanding Knowledge / Skills / Attributes etc.		
I	Introduction to C++	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	To understand concepts and features of CPP and input / output Operators of CPP	Critical thinking and problem-solving skills	15%
II	Beginning with C++	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	To understand Data type and Keywords, Declaration of variables, dynamic initialization of variables, reference variable, Function prototyping, Inline functions, Default arguments.	Information Literacy, critical thinking, problem solving, analytical reasoning	20%
III	Classes and Objects	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	To understand basic Structure and class, Class, Object, Access specifies,	Critical thinking, problem solving, analytical	15%

Unit	Unit Title	Suggestive teaching methodology	Practical	Outcome expected		Weightage of Marks (%)
				Conceptual understanding	Knowledge / Skills / Attributes etc.	
				defining data member, Defining member functions inside and outside class definition. Simple C++ program using class, Memory allocation for objects, Static data members and static member function, Array of objects, objects as a function argument, Friend function and Friend class, Function returning objects	reasoning, Life long Learning, Application Skills	
IV	Constructors and Destructors	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	To understand constructor, destructor concept in C++	Critical thinking, problem solving, analytical reasoning, Life long Learning, Experimental Learning	25%
V	Inheritance	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	To understand inheritance and its practical implementation.	Critical thinking, Problem solving, Analytical reasoning, Life long Learning, Experimental	13%

Unit	Unit Title	Suggestive teaching methodology	Practical	Outcome expected		Weightage of Marks (%)
				Conceptual understanding	Knowledge / Skills / Attributes etc.	
					Learning	
VI	Polymorphism	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	To understand Polymorphism and its practical implementation.	Critical thinking, Problem solving, Analytical reasoning, Life long Learning, Experimental Learning	12%

#### Evaluation Method:

Unit	Evaluation Method	Marks (100)			Project / Practical (If any)
		Formative Assessment		Summative Assessment	
		CCE I (20)	CCE II (20)	SEMESTER (60)	
1	Test and lab course work	MCQ	Assignment		Practical in Computer Laboratory
2	Assignment and Quiz	MCQ	Assignment		Practical in Computer Laboratory
3	Test and Lab course work	MCQ	Assignment		Practical in Computer Laboratory
4	Test, Quiz or Lab course work.	MCQ	Assignment		Practical in Computer Laboratory
5	Assignment and Quiz	MCQ	Assignment		Practical in Computer Laboratory
6	Assignment and Quiz	MCQ	Assignment		Practical in Computer Laboratory

#### Suggested Books:

Sr. No.	Name of Book	Author	Publication	Edition	Place
1	Object Oriented programming with C++	E Balagurusamy	McGraw Hill Education (India).	Eighth	New Delhi
2	Object Oriented Programming with C++	Robert Lafore	PEARSON	Third	New Delhi
3	The Complete Reference C++	Herbert Schildt	McGraw Hill Education (India)	Fourth	New Delhi

### Suggested Web/E-Learning Resources

<b>Sr. No.</b>	<b>Topic of the course</b>	<b>Lectures (Available on Youtube / Swayam / MOOCS etc.)</b>	<b>Link</b>	<b>Journals / Articles / Case studies</b>
1	Advanced C++	Swayam	<a href="https://onlinecourses.swayam2.aic.in/aic20_sp01/preview">https://onlinecourses.swayam2.aic.in/aic20_sp01/preview</a>	online course
2	Introduction to Programming in C++	MOOC	<a href="https://www.edx.org/course/introduction-to-programming-in-c">https://www.edx.org/course/introduction-to-programming-in-c</a>	online course
3	Advanced Programming in C++	MOOC	<a href="https://www.edx.org/course/advanced-programming-in-c">https://www.edx.org/course/advanced-programming-in-c</a>	online course

●



<b>Course Code:</b> <b>B4-21/402</b>	<b>Subject / Course : Object Oriented Software Engineering</b>	<b>Marks: 100</b> <b>Credits: 3</b>
<b>Course Objectives :</b>		
<ol style="list-style-type: none"> <li>1. To understand the fundamentals of object modelling and Unified Process.</li> <li>2. To learn designing with static and dynamic UML diagrams and its implementation.</li> <li>3. To learn Behavioural Modeling and Architectural Modeling techniques</li> <li>4. To Design the software based on Object Oriented Analysis and Design</li> </ol>		
<b>Course Outcome :</b>		
After completing the course, the student shall be able to		
<b>CO1:</b> Students will be able to learn fundamentals of object modelling and Unified Process		
<b>CO2:</b> Students will acquire Knowledge of UML diagrams and its implementation.		
<b>CO3:</b> Students will learn Behavioural Modeling and Architectural Modeling techniques		
<b>CO4:</b> Students will be able to learn software design with design patterns using Object Oriented Analysis and Design		

<b>Unit</b>	<b>Unit Title</b>	<b>Contents</b>	<b>No. of lectures</b>
I	Introduction and basics of Software Modeling	1.1 Software Life Cycle Models (Revision of SE) 1.2 System Concepts 1.3 Project Organization 1.4 Communication in Project Management 1.5 Risk management in Project Management	4
II	SRS Documentation	2.1 SRS Specification 2.2 Requirement Elicitation 2.3 Business Engineering	4
III	Introduction to UML	3.1 Concept of UML 3.2 Advantages of UML	2
IV	Structural Modeling	4.1 Classes 4.2 Relationship 4.3 Common Mechanism 4.4 Class Diagram 4.5 Advanced Classes 4.6 Advanced Relationship	11

Unit	Unit Title	Contents	No. of lectures
		4.7 Interface 4.8 Types and Roles 4.9 Packages 4.10 Object Diagram	
V	Basic Behavioural Modeling	5.1 Interactions 5.2 Use Cases and Use Case Diagram with stereo types 5.3 Interaction Diagram 5.4 Sequence Diagram 5.5 Activity Diagram 5.6 State Chart Diagram (Case studies for all types of Behaviour Modeling)	11
VI	Architectural Modeling	6.1 Component 6.2 Components Diagram 6.3 Deployment Diagram 6.4 Collaboration Diagram (Case studies for all types of Architectural Modeling)	7
VII	Object Oriented Analysis	7.1 Iterative Development and the Rational Unified Process 7.2 Inception 7.3 Understanding Requirements 7.4 Use Case Model From Inception to Elaboration 7.5 Elaboration	5
VIII	Object Oriented Design	8.1 The Booch Method, The Coad and Yourdon Method and Jacobson Method and Raumbaugh Method 8.2 The Generic Components of the OO Design Model 8.3 The System Design Process - Partitioning the Analysis Model, Concurrency and Sub System Allocation, Task Management Component, The Data Management Component, The Resource Management Component, Inter Sub System Communication	4
	<b>Total</b>		<b>48</b>

**Evaluation Method:**

Unit	Evaluation Method	Marks (100)			Project / Practical (If any)
		Formative Assessment		Summative Assessment	
		CCE I (20)	CCE II (20)	SEMESTER (60)	
I	Assignments				NA
II	Assignments				NA
III	MCQs / Presentations on Object Oriented Software Engineering				NA
IV	MCQs / Presentations on Object Oriented Software Engineering				NA
V	UML Case study submission				NA
VI	UML Case study submission				NA
VII	UML Case study submission				NA
VIII	MCQs/ Presentations on Object Oriented Software Engineering				NA

**Suggested Readings:**

Sr. No.	Title of the Book	Author/s	Publication
1.	The Unified Modeling Language User/Reference Guide,	Grady Booch, James Rumbaugh	Pearson Education Inc
2.	The Unified software development Process	Ivar Jacobson, Grady Booch	Pearson Education
3.	Agile Software development	Alistair Cockbair	Pearson Education

**Suggested Web/E-Learning Resources:**

<b>Sr. No.</b>	<b>Topic of the Lecture</b>	<b>Lectures (Available on Youtube / Swayam / MOOCS etc.)</b>	<b>Films</b>	<b>Journals / Articles / Case studies</b>
1.	Software Engineering	<a href="https://onlinecourses.nptel.ac.in/noc19_cs69/preview">https://onlinecourses.nptel.ac.in/noc19_cs69/preview</a>		
2.	Object Oriented Design	<a href="https://www.mooc-list.com/tags/object-oriented-design">https://www.mooc-list.com/tags/object-oriented-design</a>		

•

<b>Course Code :</b> <b>B4-21/403A</b>	<b>Subject / Course : Dot Net Programming</b>	<b>Total Marks: 100</b> <b>Credits: 3</b>
<b>Course Objectives :</b>		
<ol style="list-style-type: none"> <li>1. To gain Knowledge of the .NET technologies framework .</li> <li>2. To understand and implement various controls for Creating a web Application using ASP.net and c#</li> <li>3. To learn data access with ADO.net.</li> </ol>		
<b>Course Outcome :</b>		
After completing the Course, the student shall be able to:		
<b>CO1:</b> Understand the .NET technologies for web programming		
<b>CO2:</b> Understand how to design and develop interactive and responsive web applications.		
<b>CO3:</b> Design and Implement database connectivity using ADO.net for C# and ASP		

<b>Unit</b>	<b>Unit Title</b>	<b>Contents</b>	<b>No. of Lectures</b>
I	Introduction to .net framework	1.1 What is the framework? 1.2 Architecture of .net framework 1.3 IDE (Integrated Development Environment) 1.4 Event Driven Programming 1.5 Windows Application 1.5.1 Controls 1.5.2 Menus and Popup Menu 1.5.3 Predefined Dialog Controls 1.5.4 DialogBox	10
II	Introduction to C#	2.1 C# Language fundamentals 2.1.1 Data type and Control Constructs 2.1.2 Value and Reference Types,Boxing 2.1.3 Arrays 2.1.4 String class and its various operations 2.1.5 Functions 2.2 Object Oriented Concepts 2.2.1 Defining classes and Objects 2.2.2 Access modifiers 2.2.3 Constructors 2.2.4 Inheritance	14

<b>Unit</b>	<b>Unit Title</b>	<b>Contents</b>	<b>No. of Lectures</b>
		2.2.5 Interface 2.2.6 Abstract Class 2.2.7 Method Overloading and Overriding 2.2.8 Delegates	
III	Creating Web Forms Applications	Introduction to ASP.NET 3.1 What is ASP.NET? 3.2 ASP.NET Page Life Cycle 3.3 Architecture of ASP.NET 3.4 Forms, WebPages, HTML forms, 3.5 Request & Response in Non-ASP.NET pages 3.6 Using ASP.NET Server Controls 3.7 Overview of Control structures 3.8 Functions 3.9 HTML events 3.9.1 ASP.NET Web control events 3.9.2 Event driven programming and postback 3.10 Introduction to Web forms 3.10.1 Web Controls 3.10.2 Server Controls 3.10.3 Client Controls 3.10.4 Navigation Controls 3.10.5 Validations 3.10.6 Master Page 3.10.7 State Management Techniques	12
IV	Storing and Retrieving Data with ADO.NET	4.1 Basics of ADO.net 4.1.1 Connection Object 4.1.2 Command Object 4.1.3 Dataset 4.1.4 Data Table 4.1.5 Data Reader Object 4.1.6 Data Adapter Object 4.2 Data grid view Data Binding: Insert, Update, Delete records 4.3 Navigation Using Data Source	12
<b>Total Number of Lectures</b>			<b>48</b>

**Teaching Methodology:**

Unit	Unit Title	Teaching methodology	Project (If any)	Outcome expected- Conceptual understanding Knowledge / Skills / Attributes etc.		Weightage of Marks (%)
				Course Outcome (CO)	Learning Outcome (LO)	
1	Introduction to .net framework	Lectures - Demonstration and Practical Implementation in Lab	Practical	To understand .net framework	<ul style="list-style-type: none"> <li>➤ .Net Architecture</li> <li>➤ Fundamental knowledge of .net</li> <li>➤ To get knowledge of various windows application controls to develop web forms</li> </ul>	20%
2	Introduction to C#	Lectures - Demonstration and Practical Implementation in Lab	Lab	Understand the Fundamentals C# Language	<ul style="list-style-type: none"> <li>➤ Basics of C#</li> <li>➤ To get knowledge of Object oriented language</li> </ul>	30%
3	Creating Web Forms Applications	Lectures - Demonstration and Practical Implementation in Lab	Lab	To study how to create web forms in ASP.Net.	<ul style="list-style-type: none"> <li>➤ Create web forms</li> <li>➤ Understand the controls</li> <li>➤ ASP.net tools</li> </ul>	30%
4	Storing and Retrieving Data with ADO.NET	Lectures - Demonstration and Practical Implementation in Lab	Lab	To study how to access data with ADO.Net.	<ul style="list-style-type: none"> <li>➤ Database connectivity</li> </ul>	20%

**Evaluation Method:**

Unit	Evaluation Method	Marks (100)			Project / Practical (If any)
		Formative Assessment		Summative Assessment	
		CCE I (20)	CCE II (20)	SEMESTER (60)	
I	Assignment and Lab work	MCQ	Assignment		Practicals in Computer Lab
II	Assignment and Lab work	MCQ	Assignment		Practicals in Computer Lab
III	Assignment and Lab work	MCQ	Assignment		Practicals in Computer Lab
IV	Assignment and Lab work	MCQ	Assignment		Practicals in Computer Lab

**Suggested Readings:**

Sr. No.	Title of the Book	Author/s	Publication	Edition	Place
1.	Inside C#	Tom Archer	Microsoft Press	-	-
2.	Microsoft ASP.NET 4.0 Step by Step	George Shepherd	Microsoft Press	-	-
3.	ASP.net – The Complete Reference	MacDonald	Tata McGraw Hill	-	-
4.	Murach's ASP.NET 4.6 Web Programming in C#2015	Mary Delamater and Anne Bohem	SPD	Sixth	-

**Suggested Web/E-learning Resources:**

SR NO	Topic	Lectures (Available on Youtube / Swayam / MOOCS etc)	Films	Journals / Articles / Case studies
1	Introduction to .net framework	<a href="https://www.javatpoint.com/net-framework">https://www.javatpoint.com/net-framework</a> <a href="https://dotnet.microsoft.com/en-us/learn/dotnet/what-is-dotnet">https://dotnet.microsoft.com/en-us/learn/dotnet/what-is-dotnet</a> <a href="https://www.c-sharpcorner.com/article/what-is-net/">https://www.c-sharpcorner.com/article/what-is-net/</a>		



<b>SR NO</b>	<b>Topic</b>	<b>Lectures (Available on Youtube / Swayam / MOOCS etc)</b>	<b>Films</b>	<b>Journals / Articles / Case studies</b>
2	Introduction to C#	<a href="https://www.w3schools.com/cs/index.php">https://www.w3schools.com/cs/index.php</a> <a href="https://www.javatpoint.com/c-sharp-tutorial">https://www.javatpoint.com/c-sharp-tutorial</a>		
3	Creating Web Forms Applications	<a href="https://www.javatpoint.com/asp-net-tutorial">https://www.javatpoint.com/asp-net-tutorial</a>		
4	Storing and Retrieving Data with ADO.NET	<a href="https://www.c-sharpcorner.com/UploadFile/18fc30/understanding-the-basics-of-ado-net/">https://www.c-sharpcorner.com/UploadFile/18fc30/understanding-the-basics-of-ado-net/</a> <a href="https://www.javatpoint.com/ado-net-tutorial">https://www.javatpoint.com/ado-net-tutorial</a>		

●

<b>Course Code : B4-21/403B</b>	<b>Subject / Course : Python Programming</b>	<b>Total Marks: 100 Credits: 3</b>
<p><b>Course Objectives:</b></p> <ul style="list-style-type: none"> <li>• To learn and understand Python programming basics and paradigm.</li> <li>• To learn and understand python looping, control statements and string manipulations.</li> <li>• Students should be made familiar with the concepts of GUI controls and designing GUI applications.</li> <li>• To learn and know the concepts of file handling, exception handling.</li> </ul>		
<p><b>Course Outcome:</b></p> <p>On completion of the course, student will be able</p> <p><b>CO1:</b> Define and demonstrate the use of built-in data structures “lists” and “dictionary”.</p> <p><b>CO2:</b> Design and implement a program to solve a real world problem.</p> <p><b>CO3:</b> Design and implement GUI application.</p> <p><b>CO4:</b> Define and demonstrate how to handle exceptions and files.</p>		

<b>Unit</b>	<b>Unit Title</b>	<b>Contents</b>	<b>No. of Lectures</b>
I	Introduction to Python	1.1 History, feature of Python, setting up path, working with python Interpreter, basic syntax, variable and data types, operators 1.2 Conditional statements-If, If-Else, nested if-else, Examples. 1.3 Looping-For, While, Nested loops, Examples 1.4 Control Statements-Break, Continue, Pass. 1.5 String Manipulation-Accessing String, Basic Operations, String Slices, Function and Methods, Examples. 1.6 Lists-Introduction, accessing list, operations, working with lists, function & methods. 1.7 Tuple-Introduction, Accessing tuples, operations working, function & methods, Examples. 1.8 Dictionaries-Introduction, Accessing values in dictionaries, working with dictionaries, properties, function, Examples. 1.9 Functions-Defining a function, calling a function, types of function, function arguments, anonymous function, global & local variable, Examples.	12

<b>Unit</b>	<b>Unit Title</b>	<b>Contents</b>	<b>No. of Lectures</b>
II	Modules and Packages	<p><b>2.1 Built in Modules</b>  2.1.1 Importing modules in python program.  2.1.2 Working with Random Modules.  2.1.3 E.g. - built-ins, time, date time, calendar, sys, etc.</p> <p><b>2.2 User Defined functions</b>  2.2.1 Structure of Python Modules.</p> <p><b>2.3 Packages</b>  2.3.1 Predefined Packages.  2.3.2 User defined Packages.</p>	8
III	Classes , Objects and Inheritance	<p><b>3.1 Classes and Objects</b>  3.1.1 Classes as User Defined Data Type  3.1.2 Objects as Instances of Classes  3.1.3 Creating Class and Objects  3.1.4 Creating Objects By Passing Values  3.1.5 Variables &amp; Methods in a Class</p> <p><b>3.2 Inheritance</b>  3.2.1 Single Inheritance  3.2.2 Multilevel Inheritance  3.2.3 Multiple Inheritance  3.2.4 Hybrid Inheritance  3.2.5 Hierarchical Inheritance  3.2.6 IS-A Relationship and HAS-A Relationship</p>	10
IV	Exception Handling	4.1 Python Exception 4.2 Common Exception 4.3 Exception handling in Python (try-except-else) 4.4 The except statement with no exception 4.5 Multiple Exception 4.6 The try-finally clause 4.7 Custom Exception and assert statement	8
V	GUI Programming	5.1 Introduction 5.2 Tkinter programming 5.4 Tkinter widgets 5.5 Frame 5.6 Button 5.7 Label 5.8 Entry	10
<b>Total No of Lectures</b>			<b>48</b>

**Teaching Methodology:**

Unit	Unit Title	Suggestive teaching methodology	Practical	Outcome expected		Weightage of Marks (%)
				Conceptual understanding Knowledge / Skills / Attributes etc.		
I	Introduction to Python	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	To understand Conditional statements, Looping, Control Statements, String Manipulation, Lists, Tuple, Dictionaries and Functions.	critical thinking and problem-solving skills	15%
II	Modules and Packages	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	To understand Built in Modules, User Defined functions, Packages.	critical thinking, problem solving, analytical reasoning	20%
III	Classes, Objects and Inheritance	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	To understand Classes and Objects and Inheritance.	Life long Learning, Application Skills	15%
IV	Exception Handling	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	To understand Exception and try-finally clause.	Problem solving, Experimental Learning	25%
V	GUI Programming	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	To understand Tkinter programming, Tkinter widgets, Frame, Button, Label, Entry.	Life long Learning, Experimental Learning, Application Skills	25%

**Evaluation Method:**

Unit	Evaluation Method	Marks (100)			Project / Practical (If any)
		Formative Assessment		Summative Assessment	
		CCE I (20)	CCE II (20)	SEMESTER (60)	
1	Test and lab course work	MCQ	Assignment		Practical in Computer Laboratory
2	Assignment and Quiz	MCQ	Assignment		Practical in Computer Laboratory
3	Test and Lab course work	MCQ	Assignment		Practical in Computer Laboratory
4	Test, Quiz or Lab course work.	MCQ	Assignment		Practical in Computer Laboratory
5	Assignment and Quiz	MCQ	Assignment		Practical in Computer Laboratory

**Suggested Readings:**

Sr. No.	Name of Book	Author	Publication	Edition
1	Python Programming: An introduction to computer science	John Zelle	Independent publication.	Third
2	Learning Python	Mark Lutz	O'Reilly	Fourth
3	Programming Python	Mark Lutz	O'Reilly	Fourth

**Suggested Web/E-Learning Resources:**

Sr. No.	Topic of the course	Lectures (Available on Youtube / Swayam / MOOCS etc.)	Link	Journals / Articles / Case studies
1	Python 3.4.3	Swayam	<a href="https://onlinecourses.swayam2.ac.in/aic20_sp33/preview">https://onlinecourses.swayam2.ac.in/aic20_sp33/preview</a>	online course
2	Programming for Everybody (Getting Started with Python)	edX	<a href="https://www.edx.org/course/programming-for-everybody-getting-started">https://www.edx.org/course/programming-for-everybody-getting-started</a>	online course

<b>Course Code:</b> <b>B4-21/404A</b>	<b>Subject / Course : Big Data</b>	<b>Marks: 100</b> <b>Credits: 3</b>
<p><b>Course Objectives:</b></p> <ol style="list-style-type: none"> <li>1. To understand big data and big data analytics techniques..</li> <li>2. To enable learners to develop expert knowledge and analytical skills in current and developing areas of analysis statistics, and machine learning</li> <li>3. To enable the learner to identify, develop and apply detailed analytical, creative, problem-solving skills.</li> <li>4. To enable the learner to understand data analytics with R programming.</li> </ol>		
<p><b>Course Outcome:</b></p> <p>After completing the course, the student shall be able to</p> <p><b>CO1:</b> To understand big data and big data analytics techniques..</p> <p><b>CO2:</b> To develop expert knowledge and analytical skills in current and developing areas of analysis statistics, and machine learning</p> <p><b>CO3:</b> To identify, develop and apply detailed analytical, creative, problem solving skills.</p>		

<b>Unit</b>	<b>Unit Title</b>	<b>Contents</b>	<b>No. of Lectures</b>
I	Introduction To Big Data	1.1 Introduction to Big Data 1.2 Types of Digital Data 1.3 Big Data Analytics 1.4 Challenges of Big Data 1.5 Applications of Big data	4
II	Introduction to Data Science	2.1 Basics of Data Analytics 2.1.1 Lifecycle of Data Analytics 2.2 Types of Analytics– 2.2.1 Descriptive, 2.2.2 Predictive, 2.2.3 Prescriptive 2.2.4 Statistical Inference 2.3 Populations and Sample 2.3.1 Statistical modeling, 2.3.2 Probability, 2.3.3 Distribution	10

Unit	Unit Title	Contents	No. of Lectures
		2.3.4 Correlation 2.3.5 Regression	
III	Machine Learning	1.1 Basics of Machine Learning 1.2 Supervised Machine Learning 3.2.1 K-Nearest-Neighbours 3.2.2 Naïve Bayes 3.2.3 Decision tree 3.2.4 Support Vector Machines 1.3 Unsupervised Machine Learning 3.3.1 Cluster analysis 3.3.2 K means 3.3.3 EM Algorithm 3.3.4 Association Rule Mining 3.3.5 Apriori algorithms 1.4 Regression Analysis 3.4.1 Linear Regression 3.4.2 Nonlinear Regression	20
IV	Data Analytics with R	4.1 Introduction 4.2 Data Manipulation 4.3 Data Visualization 4.4 Data Analysis	14
<b>Total No of Lectures</b>			<b>48</b>

#### Teaching Methodology:

Unit	Unit Title	Suggestive teaching methodology	Project (If any)	Outcome expected	Weightage of Marks (%)
				Conceptual understanding Knowledge / Skills / Attributes etc.	
I	Introduction To Big Data	Lecture - Demonstration and case study-based learning		1. To understand the concept of Big Data 2. To understand the applications of big data using case study	10%

Unit	Unit Title	Suggestive teaching methodology	Project (If any)	Outcome expected	Weightage of Marks (%)
				Conceptual understanding Knowledge / Skills / Attributes etc.	
II		Lecture - Demonstration and case study-based learning		1. To develop and apply detailed analytical, creative, problem-solving skills. 2. To understand the analytic techniques.	30%
III	Machine Learning	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	1. To develop and apply detailed analytical, creative, problem-solving skills. 2. To understand machine learning algorithm and implementation.	30%
IV	Data Analytics with R	Lecture - Demonstration and case study-based learning	Practical	1. To understand data manipulation, visualization and analysis with R programming.	30%

**Evaluation Method:**

Unit	Evaluation Method	Marks (100)			Project / Practical (If any)
		Formative Assessment		Summative Assessment	
		CCE I (20)	CCE II (20)	SEMESTER (60)	
1	Assignment, and case study	MCQ	Assignment		Practical in Computer Laboratory
2	Assignment, Quiz and Test	MCQ	Assignment		Practical in Computer Laboratory
3	Assignment and Lab Course work	MCQ	Assignment		Practical in Computer Laboratory
4	Assignment and Lab Course work	MCQ	Assignment		Practical in Computer Laboratory



**Suggested Books :**

<b>Sr. No.</b>	<b>Name of the Book</b>	<b>Author</b>	<b>Publication</b>	<b>Edition</b>	<b>Place</b>
1	"Big Data Analytics" Wiley 2015.	Seema Acharya, Subhasini Chellappan	Wiley Publication	--	--
2	Big Data and Business Analytics	Jay Liebowitz	Auerbach Publications, CRC press (2013)	--	--
3	BigDataAnalytics: Disruptive Technologies for Changing the Game	ArvindSathi	MC Press, 2012	--	--

**Suggested Web/E-Learning Resources:**

<b>Sr. No.</b>	<b>Topic of the course</b>	<b>Lectures (Available on Youtube / Swayam / MOOCS etc.)</b>	<b>Link</b>	<b>Journals / Articles / Case studies</b>
1	Introduction to big data	NPTEL	<a href="https://www.youtube.com/watch?v=rvJgArru8dI">https://www.youtube.com/watch?v=rvJgArru8dI</a>	online course
2	Introduction to machine learning	MOOC	<a href="https://www.edx.org/course/machine-learning-fundamentals-2">https://www.edx.org/course/machine-learning-fundamentals-2</a>	online course

●

<b>Course Code :</b> <b>B4-21/404B</b>	<b>Subject / Course : Block Chain</b>	<b>Marks : 100</b> <b>Credits: 3</b>
<b>Course Objectives :</b> <ol style="list-style-type: none"> <li>1. Understand how blockchain systems (mainly Bitcoin and Ethereum) work,</li> <li>2. To securely interact with them,</li> <li>3. Design, build, and deploy smart contracts and distributed applications,</li> <li>4. Integrate ideas from blockchain technology into their own projects.</li> </ol>		
<b>Course Outcome :</b> After completing the course, the student shall be able to <ol style="list-style-type: none"> <li><b>CO1:</b> To understand working of block chain system.</li> <li><b>CO2:</b> To understand secured interaction with system.</li> <li><b>CO3:</b> To understand designing, building and deployment of smart contracts and distributed applications.</li> <li><b>CO4:</b> To integrate ideas from blockchain technology into their own project.</li> </ol>		

<b>Unit</b>	<b>Unit Title</b>	<b>Contents</b>	<b>No. of Lectures</b>
I	Introduction To Block Chain	1.1 DigitalTrust 1.2 Asset 1.3 Transactions 1.4 DistributedLedgerTechnology 1.5 Types of network 1.6 Components of blockchainorDLT 1.7 Ledger 1.7.1 Blocks 1.7.2 Blockchain 1.8 PKI and Cryptography 1.8.1 Private keys 1.8.2 Publickeys 1.8.3 Hashing 1.8.4 DigitalSignature 1.9 Consensus 1.9.1 Byzantine Fault 1.9.2 Proof of Work 1.9.3 Poof of Stake 1.10 Security	12

Unit	Unit Title	Contents	No. of Lectures
		1.10.1 DDos 1.11 Crypto currency 1.12 Digital Token	
II	How Blockchain Works	2.1 How Blockchain Works 2.2 Structure of Blockchain 2.3 Block 2.4 Hash 2.5 Blockchain 2.6 Distributed 2.7 Lifecycle of Blockchain 2.8 Smart Contract 2.9 Consensus Algorithm 2.10 Proof of Work 2.11 Proof of Stake 2.12 Practical Byzantine 2.13 Fault Tolerance 2.14 Actors of Blockchain 2.15 Blockchain developer 2.16 Blockchain operator 2.17 Blockchain regulator 2.18 Blockchain user 2.19 Membership service provider 2.20 Building A SmallBlockchain Application	12
III	Introduction to Bitcoin	3.1 Currency 3.2 DoubleSpending 3.3 Cryptocurrency 3.4 P2PPaymentGateway 3.5 Wallet 3.6 Mining	8
IV	Ethereum	4.1 Ethereum network 4.2 EVM 4.3 Transaction fee 4.4 Mist 4.5 Ether,gas 4.6 Solidity-Smart contracts 4.7 Truffle 4.8 Web3 4.9 Design and issue Crypto currency 4.10 Mining 4.11 DApps 4.12 DAO	8

Unit	Unit Title	Contents	No. of Lectures
V	Introduction To HyperledgerFabricV1.1	5.1 Introduction to Hyperledger 5.2 What is Hyperledger 5.3 Why Hyperledger 5.4 Where can Hyperledger be used 5.5 Hyperledger Architecture 5.6 Membership 5.7 Blockchain 5.8 Transaction 5.9 Chaincode 5.10 Hyperledger Fabric 5.11 Features of Hyperledger	8
<b>Total No of Lectures</b>			<b>48</b>

**Teaching Methodology:**

Unit	Unit Title	Suggestive teaching methodology	Project (If any)	Outcome expected	Weightage of Marks (%)
				Conceptual understanding Knowledge / Skills / Attributes etc.	
I	Introduction To Blockchain	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	1. Understand what and why of Blockchain 2. Explore the major components of Blockchain 3. To understand security for block chain 4. To understand the creation of public and private keys.	10%

Unit	Unit Title	Suggestive teaching methodology	Project (If any)	Outcome expected	Weightage of Marks (%)
				Conceptual understanding Knowledge / Skills / Attributes etc.	
II	How Blockchain Works	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	<ol style="list-style-type: none"> <li>1. To understand working of block chain</li> <li>2. To understand structure of blockchain.</li> <li>3. To understand building of small blockchain application.</li> </ol>	20%
III	Introduction to Bitcoin	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	<ol style="list-style-type: none"> <li>2. To understand concept of cryptocurrency.</li> <li>3. To develop and apply detailed analytical, problem solving skills.</li> <li>4. To create blockchain application for bitcoin wallet.</li> </ol>	20%
IV	Ethereum	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	<ol style="list-style-type: none"> <li>1. To implement ethereum application.</li> <li>2. To create ethereum application for smart contracts.</li> <li>3. To understand how to transfer money from one account to other using ethereum.</li> </ol>	30%
V	Introduction To HyperledgerFabric V1.1	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	<ol style="list-style-type: none"> <li>1. To understand the concept of hyperledger.</li> </ol>	20%

**Evaluation Method:**

Unit	Evaluation Method	Marks (100)			Project / Practical (If any)
		Formative Assessment		Summative Assessment	
		CCE I (20)	CCE II (20)	SEMESTER (60)	
1	Assignment and Lab Course work	MCQ	Assignment		Practical in Computer Laboratory
2	Assignment and Lab Course work	MCQ	Assignment		Practical in Computer Laboratory
3	Assignment and Lab Course work	MCQ	Assignment		Practical in Computer Laboratory
4	Assignment and Lab Course work	MCQ	Assignment		Practical in Computer Laboratory
5	Assignment and Lab Course work	MCQ	Assignment		Assignment

**Suggested Readings :**

Sr. No	Title of the book	Author	Publication
1.	Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction	Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and StevenGoldfeder	Princeton University Press (July 19, 2016)
2.	Mastering Bitcoin: Unlocking Digital Cryptocurrencies	Antonopoulos	Shroff Publications
3.	“ETHEREUM: A Secure Decentralized Transaction Ledger,”	DR. Gavin Wood	

**Suggested Web/E-Learning Resources :**

<b>Sr. No.</b>	<b>Topic of the course</b>	<b>Lectures (Available on Youtube / Swayam / MOOCS etc.)</b>	<b>Link</b>	<b>Journals / Articles / Case studies</b>
1	“Introduction to Block Chain Technology & Applications”	NPTEL	<a href="https://nptel.ac.in/courses/106/104/106104220/">https://nptel.ac.in/courses/106/104/106104220/</a>	online course
2	“Blockchain Architecture & Use Cases”	NPTEL	<a href="https://nptel.ac.in/courses/106/105/106105184/">https://nptel.ac.in/courses/106/105/106105184/</a>	online course



<b>Course Code :</b> <b>B4-21/405</b>	<b>Subject / Course: Project based on HTML, CSS, JS</b>	<b>Total Marks : 100</b> <b>Credits : 3</b>
<p><b>Guidelines</b></p> <ol style="list-style-type: none"> <li>1. Students should work in a team of maximum 2 students.</li> <li>2. Students can choose a project topic HTML ,CSS,JS technology</li> <li>3. The student group will work independently throughout the project work including: problem identification, information searching, literature study, design and analysis, implementation, testing, and the final reporting.</li> <li>4. Project guide must conduct project presentations to monitor the progress of the project groups.</li> <li>5. At the end of the project, the group should prepare a report which should conform to international academic standards. The report should follow the style in academic journals and books, with clear elements such as: abstract, background, aim, design and implementation, testing, conclusion and full references, Tables and figures should be numbered and referenced to in the report.</li> <li>6. The final project presentation with demonstration will be evaluated by the project guide (appointed by the college) and one more examiner.</li> </ol>		

**Recommended Documentation contents:**

<b>Title</b>	<b>Contents</b>
Abstract	<p><b>Introduction</b></p> <ol style="list-style-type: none"> <li>1. motivation</li> <li>2. problem statement</li> <li>3. purpose/objective and goals</li> <li>4. literature survey</li> <li>5. project scope and limitations</li> </ol>
System analysis	<ol style="list-style-type: none"> <li>1. Existing systems scope and limitations of existing systems</li> <li>2. project perspective, features stakeholders</li> <li>3. Requirement analysis</li> <li>4. Functional requirements, performance requirements, security requirements etc.</li> </ol>
System Design	<ol style="list-style-type: none"> <li>1. Excise Payment, etc.</li> <li>2. Design constraints</li> <li>3. System Model: DFD</li> <li>4. Data Model</li> </ol>



<b>Title</b>	<b>Contents</b>
	5. User interfaces
Implementation details	1. Software/hardware specifications
Outputs	
Conclusion and Recommendations	
Future Scope	
Bibliography and References	

**Evaluation guidelines:**

<b>IA (30 marks)</b>			<b>EE (70 marks)</b>		
First presentation	Second presentation	Documentation	Project Logic / Presentation	Documentation	Viva
10	10	10	40	10	20



<b>Course Code:</b> B4-21/407	<b>Subject / Course : jQuery</b>	<b>Marks: 50</b> <b>Credits : 2</b>
<p><b>Course Objectives:</b></p> <ol style="list-style-type: none"> <li>1. To get hands-on experience on JavaScript and jQuery.</li> <li>2. To learn how to work with binding events to the controls in JavaScript.</li> <li>3. To learn how to download jQuery library and refer it to the Html page.</li> <li>4. To Learn Traversing of Html elements.</li> <li>5. To learn handling different events for different Controls.</li> <li>6. To learn how to provide effects to the elements or sections in the Html page.</li> </ol>		
<p><b>Course Outcome:</b></p> <p>After completing the course, the student shall be able to</p> <p><b>CO1:</b> Understand the practical on JavaScript and jQuery</p> <p><b>CO2:</b> Understand the working with binding events to the controls in JavaScript.</p> <p><b>CO3:</b> Understand the downloading jQuery library and refer it to the Html page.</p> <p><b>CO4:</b> Understand the Traversing of Html elements.</p> <p><b>CO5:</b> Understand the handling of different events for different controls.</p> <p><b>CO6:</b> Understand the giving effects to the elements or sections in the Html page.</p>		

<b>Unit</b>	<b>Unit Title</b>	<b>Contents</b>	<b>No. of Lectures</b>
I	Introduction to jQuery	1.1 jQuery Introduction 1.2 Install and Use jQuery Library 1.3 Un-Obstructive JavaScript 1.4 First jQuery Example 1.5 jQuery Syntax 1.6 How to escape a special character 1.7 Basic Selectors 1.8 Traversal Functions	5
II	HTML Manipulation	2.1 Getting Setting values from elements 2.2 Handling attributes 2.3 Inserting New elements 2.4 Deleting/Removing elements	5

Unit	Unit Title	Contents	No. of Lectures
		2.5 CSS manipulations 2.6 Dimensions 2.7 Positioning	
III	Effects and Events Effects	3.1 Showing/Hiding elements 3.2 Sliding elements 3.3 Fading elements 3.4 Deleting animation elements 3.5 Custom animation 3.6 Working with events.	5
<b>Total No of Lectures</b>			<b>15</b>

**Teaching Methodology:**

Unit	Unit Title	Suggestive teaching methodology	Project (If any)	Outcome expected	Weightage of Marks (%)
				Conceptual understanding Knowledge / Skills / Attributes etc.	
I	Introduction to jQuery	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	1. To understand how to download jQuery library and refer it to the Html page. 2. To understand and implement simple jQuery example	20%
II	HTML Manipulation	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	1. To understand and implement HTML manipulations and CSS manipulations.	40%
III	Effects and Events Effects	Lecture - Demonstration and Practical Implementation in Laboratory	Practical	1. To understand and implement effects and animation.	40%

**Evaluation Method:**

Unit	Evaluation Method	Marks (50)			Project / Practical (If any)
		Formative Assessment		Summative Assessment	
		CCE I (25)	CCE II (25)	SEMESTER	
1	Assignment and Lab Course work	Written	Assignment	-	Practical in Computer Laboratory
2	Assignment and Lab Course work	Written	Assignment	-	Practical in Computer Laboratory
3	Assignment and Lab Course work	Written	Assignment	-	Practical in Computer Laboratory

**Suggested Books:**

Sr. No.	Name of the Book	Author	Publication	Edition	Place
1	jQuery pocket reference	David Flanagan	O'Reilly Media, Inc.	--	--
2	Learning jQuery	Jonathan Chaffer	Packt Publisher	--	--
3	JavaScript and jQuery	David Sawyer McFarland	Shroff Publisher	--	--

●●●