Re-Accredited by NAAC with 'A' Grade
VEER NARMAD SOUTH GUJARAT UNIVERSITY
University Campus, Udhna-Magdalla Road, SURAT - 395 007, Gujarat, India.
વીર નર્મદ દક્ષિણ ગુજરાત યુનિવર્સિટી
યુનિવર્સિટી કેમ્પસ, ઉધના-મગદલ્લા રોડ, સુરત - ૩૯૫ ૦૦૭, ગુજરાત, ભારત.
Tel : +91 - 261 - 2227141 to 2227146, Toll Free : 1800 2333 011, Fax : +91 - 261 - 2227312
E-mail : info@vnsgu.ac.in, Website : www.vnsgu.ac.in

ક્રમાંક : એસ./સિલેબસ/પરિપત્ર/૬૫૩૬/૨૦૨૨
તા. ૩૧/૦૩/૨૦૨૨

પ્રતિ,
વડાશ્રી,
ડિપાર્ટમેન્ટ ઓફ કોમ્પ્યુટર સાયન્સ,
વીર નર્મદ દક્ષિણ ગુજરાત યુનિવર્સિટી,
સુરત.

વિષય :– એમ.સી.એ. (M.C.A.1st Year) સેમેસ્ટર – ૧ અને ૨ અભ્યાસક્રમ રિવાઇઝડ અભ્યાસક્રમ બાબત.

મહાશયશ્રી,

સવિનય જણાવવાનું કે, શૈક્ષણિક વર્ષ–૨૦૨૨–૨૩ થી અમલમાં આવનાર M.C.A.(1st year),નો પેટાસમિતિએ તૈયાર કરેલ અભ્યાસક્રમ કોમ્પ્યુટર સાયન્સ અભ્યાસ સમિતિની તા.૨૧/૦૩/૨૦૨૨ની સભાનાં ઠરાવ ક્રમાંકઃ ૩ અન્વયે મંજૂર કરી કોમ્પ્યુટર સાયન્સ એન્ડ ઈન્ફોર્મેશન ટેકનોલોજી વિદ્યાશાખાને કરેલ ભલામણ કોમ્પ્યુટર સાયન્સ એન્ડ ઈન્ફોર્મેશન ટેકનોલોજી વિદ્યાશાખાની તા.૨૧/૦૩/૨૦૨૨ની સભાનાં ઠરાવ ક્રમાંકઃ૦૩ અન્વયે સ્વીકારી એકેડેમિક કાઉન્સિલને કરેલ ભલામણ એકેડેમિક કાઉન્સિલે તેની તા.૨૩/૦૩/૨૦૨૨ ની સભાના ઠરાવ ક્રમાંક : ૧૫૩ અન્વયે મંજૂર કરેલ છે. જેની આથી જાણ કરવામાં આવે છે.

કોમ્પ્યુટર સાયન્સ અભ્યાસસમિતિની તા.૨૧/૦૩/૨૦૨૨ની સભાનાં ઠરાવ ક્રમાંકઃ૩

:: આથી ઠરાવવામાં આવે છે કે, શૈક્ષણિક વર્ષ–૨૦૨૨–૨૩ થી અમલમાં આવનાર M.C.A. (1st year),નો પેટાસમિતિએ તૈયાર કરેલ અભ્યાસક્રમ મંજૂર કરી કોમ્પ્યુટર સાયન્સ એન્ડ ઈન્ફોર્મેશન ટેકનોલોજી વિદ્યાશાખાને ભલામણ કરવામાં આવે છે.

કોમ્પ્યુટર સાયન્સ એન્ડ ઈન્ફોર્મેશન ટેકનોલોજી વિદ્યાશાખાની તા.૨૧/૦૩/૨૦૨૨ની સભાનાં ઠરાવ ક્રમાંકઃ ૩

:: આથી ઠરાવવામાં આવે છે કે, કોમ્પ્યુટર સાયન્સ અભ્યાસ સમિતિની તા.૨૧/૩/૨૦૨૨ની સભાનાં ઠરાવ ક્રમાંકઃ૩ અન્વયે કરેલ ભલામણ સ્વીકારી શૈક્ષણિક વર્ષ ૨૦૨૨–૨૩ થી અમલમાં આવનાર M.C.A. (1st year), નો અભ્યાસક્રમ મંજૂર કરવા એકેડેમિક કાઉન્સિલને ભલામણ કરવામાં આવે છે.

એકેડેમિક કાઉન્સિલની તા.૨૩/૦૩/૨૦૨૨ની સભાનાં ઠરાવ ક્રમાંકઃ૧૫૩

:: આથી ઠરાવવામાં આવે છે કે, કોમ્પ્યુટર સાયન્સ એન્ડ ઈન્ફોર્મેશન ટેકનોલોજી વિદ્યાશાખાની તા.૨૧/૦૩/૨૦૨૨ની સભાનાં ઠરાવ ક્રમાંકઃ ૩ અન્વયે કરેલ ભલામણ સ્વીકારી શૈક્ષણિક વર્ષ ૨૦૨૨–૨૩ થી અમલમાં આવનાર M.C.A. (1st year),નો અભ્યાસક્રમ મંજૂર કરવામાં આવે છે.

ઈ.ચા.કુલસચિવ

બિડાણ : ઉપર મુજબ

પ્રતિ,

૧) અધ્યક્ષશ્રી, કોમ્પ્યુટર સાયન્સ એન્ડ ઈન્ફોર્મેશન ટેકનોલોજી વિદ્યાશાખા
૨) પરીક્ષા નિયામકશ્રી, પરીક્ષા વિભાગ, વીર નર્મદ દ. ગુ. યુનિવર્સિટી, સુરત.

...તરફ જાણ તેમજ અમલ સારૂ.

Master of Computer Application

Name of Program	**Master of Computer Application**
Abbreviation	**MCA**
Duration	**2 Years**
Eligibility Criteria	Passed BCA/ Bachelor Degree in Computer Science Engineering or equivalent Degree. OR Passed B.Sc./ B.Com./ B.A. with Mathematics at 10+2 Level or at Graduation Level (with additional bridge Courses as per the norms of the concerned University).
Objective of Program	The core objective of the MCA programme is to prepare the students for productive career in software industry and academia by providing an outstanding environment of teaching and research in the core and emerging areas of the discipline.
Program Outcome	**PO1 : Fundamental Knowledge Enrichment** Program trains students with the core computer science and Information Technology (IT) knowledge domains. It also makes students capable of using core concepts in the conceptualization of domain specific application development. **PO2 : Critical Thinking Development** The program develops the skills of critical thinking, problem solving, evaluative learning of various techniques, and understanding the essence of the problem. **PO3 : Advanced Emerging Technology Awareness** The program trains students with the latest technologies that is being used in the industry. The continuous syllabi review adds value to the program for the outgoing students and make them ready to face challenging demands of the industry. **PO4 : Advanced Tools Usage** The program teaches the students to apply the advanced tools to solve real world problems. **PO5 : Nurturing Project Planning and Management Capabilities** The program trains students for designing and conceptualizing the software architecture, planning and managing the product development process of complex and live software projects. It also makes students understand the decision making for selection of an appropriate project management capabilities. **PO6 : Real World Problem / Project Development** Real world project provides the candidates exposure to work in the challenging and demanding environment of the industry. The project development training makes students employable and industry ready. **PO7 : Team Work and Leadership Development** Trains students to work in a team and also to take leadership of the of the project management team.
Program Specific Outcomes	PSO1 : Develop and strengthen the fundamental core concepts that are required to solve complex problems PSO2 : Develop the professional and entrepreneurship skills that needs independent logical and analytical thinking, teamwork and leadership PSO3 : Nurture the students to investigate for the design and development of a workable solution for a real world problem PSO4 : Develop students for self-learning and practicing challenging

	problem solution PSO5 : Train students to apply managerial skills to develop business applications. PSO6 : Train students to use recent computer science and application domain specific knowledge PSO7 : Train students to take-up the real world challenges to develop workable solution to a domain specific problem PSO8 : Inculcate the passion for continuous learning and doing research for making a successful professional career

Mapping between POs and PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
PO1								
PO2								
PO3								
PO4								
PO5								
PO6								
PO7								

Medium of Instruction: English

Program Structure: Semester 1

Course Code	Title	Teaching per week		Course Credits	University Examination		Internal Marks	Total Marks
		Theory	Practical		Duration	Marks		
101	Database Management System	4	0	4	3 Hrs	70	30	100
102	Object Oriented Programming Methodology	4	0	4	3 Hrs	70	30	100
103	Cloud Computing	4	0	4	3 Hrs	70	30	100
104	Computer Network	4	0	4	3 Hrs	70	30	100
105	Data Structures and Design and Analysis of Algorithms	4	0	4	3 Hrs	70	30	100
106	Programming Skills - I	0	3	3	2 Hrs	70	30	100
107	Programming Skills – II	0	3	3	2 Hrs	70	30	100
108	Programming Skills – III	0	4	4	2 Hrs	70	30	100
	Total	20	10	30	21 Hrs	560	240	800

Program Structure: Semester 2

Course Code	Title	Teaching per week		Course Credits	University Examination		Internal Marks	Total Marks
		Theory	Practical		Duration	Marks		
201	Artificial Intelligence	4	0	4	3 Hrs	70	30	100
202	Front End Technologies	4	0	4	3 Hrs	70	30	100
203	Programming in .NET	4	0	4	3 Hrs	70	30	100
204	Python Programming Language Blockchain Technology	4	0	4	3 Hrs	70	30	100
205	iOS Development using Swift Android Application Programming	4	0	4	3 Hrs	70	30	100
206	Programming Skills – IV	0	3	3	2 Hrs	70	30	100
207	Programming Skills – V	0	2	2	2 Hrs	70	30	100
208	Programming Skills – VI	0	2	2	2 Hrs	70	30	100
209	Programming Skills – VII	0	3	3	2 Hrs	70	30	100
	Total	20	10	30	23 Hrs	630	270	900

MCA 1st Semester

Course: 101: Database Management System

Course Code	**101**
Course Title	**Database Management System**
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	Give fundamental knowledge of Database Fundamentals like Keys & Normalisation, Oracle Database Server Architecture and Working knowledge of SQL & PL/SQL in Oracle.
Course Objective	To acquaint the students with Database Fundamentals like Keys & Normalisation in general and Oracle Architecture in particular. Also, to get working knowledge of SQL and PL/SQL programming
Course Outcome	CO1: Students will be able to understand and evaluate the role of database management systems in IT applications within an organization. CO2: Students will be able to normalise any real-life database and hence they will be able to design and implement properly structured databases of real-world applications. CO3: Students will be able to write queries using Structured Query Language (SQL) and programs using PL/SQL. CO4: Students will understand how Oracle Database works and the importance of various components of Oracle Database. CO5: Students will understand administration and security enforcement of Oracle Database. This will help them in pursuing higher studies and career in Database Administration.
Mapping between COs with PSOs	
Pre-requisite	Nil
Course Content	**Unit 1: Database Computing Models** 1.1. Introduction to Data Models 1.1.1. Conceptual Data Model-E-R Model 1.1.2. Record-based Data Models-Hierarchical, Relation, Network 1.2. Introduction to NoSQL Data Models 1.3. Comparison of SQL and NoSQL 1.4. Important terminologies in DBMS 1.4.1. Notion of Keys (Super Key, Candidate Key, Primary Key, Foreign Key) 1.4.2. Referential Integrity Constraint 1.4.3. Functional Dependencies **Unit 2: Overview of Oracle Database Server Architecture** 2.1. Architecture of Oracle Database and Oracle Instance 2.2. Overview of Physical and Logical Structures 2.3. Oracle Server Start-up and Shutdown 2.4. Creating Database **Unit 3: Oracle Tools and Utilities** 3.1. SQL 3.2. Pl/SQL Procedural Extension. 3.2.1. Overview, PL/SQL data types & Control Structures

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■		■		■		■	
CO2	■	■	■		■		■	
CO3	■		■	■		■	■	
CO4			■	■	■	■	■	■
CO5		■	■	■	■	■	■	■

	3.2.2. Cursors 3.2.3. Stored Procedures & Functions 3.2.4. Database Triggers 3.2.5. Package Creation **Unit 4: Database Administration** 4.1. Managing Users 4.1.1. User Authentication Methods 4.1.1.1. Password Authentication 4.1.1.2. O.S Authentication 4.1.2. User Configuration Setup 4.1.2.1. Profiles 4.1.2.2. Default Tablespace 4.1.2.3. Temporary Tablespace 4.1.3. Resource Management 4.1.3.1. Quotas 4.1.4. Working with user database account 4.1.4.1. Creating, Modifying, and deleting user account 4.1.4.2. Changing password 4.2. Backup & Recovery **Unit 5: Database Security** 5.1. Authentication 5.2. Privileged Accounts & Privileges 5.3. Object Security 5.4. System security 5.5. Database Roles 5.6. Database Auditing [Self-Study] Export & Import Tools, Overview of Grid Based Database **Computing, Calling External Routines from PL/SQL
Reference Books	1. Oracle 9i PL/SQL Programming -Scott Urman- Oracle Press 2. Oracle DBA Fundamentals-I - Oracle Press 3. Effective PL/SQL: - Apress 4. Expert Oracle Database Architecture 9i and 10g-Tom Kyte- Apress 5. Effective Oracle by Design - Peter Norton - Tom Kyte-Oracle Press 6. Oracle 24 X 7 Tips and Techniques - Venkat Devraj–Oracle Press 7. Expert Oracle Database 11g Administration – Alpati- Wiley Student Edition 8. Fundamentals of Database Management System- Gilleneon-Wiley Student Edition 9. SQL & PL/SQL for Oracle 11g Black Book - Deshpande-McGraw Hill 10. Beginning Oracle Database 11g Administration from novice to professional- Iggy Fernandez - Apress/Springer 11. Oracle PL/SQL-Benjamin Rosenweig & Elena Silvestrova-4/e, Pearson 12. Database Systems Using Oracle: A simplified guide to SQL & PL/SQL- Shah Nilesh- PHI 13. Learning Oracle SQL & PL/SQL: A Simplified Guide- Chatterjee, Rajeeb C- PHI 14. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Pramod J Sadalage, Martin Fowler, Wiley 15. Professional NoSQL, Shashank Tiwari, Wiley
Teaching Methodology	Class Work, Discussion, Self Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment 70% External Assessment

Course: 102: Object Oriented Programming Methodology

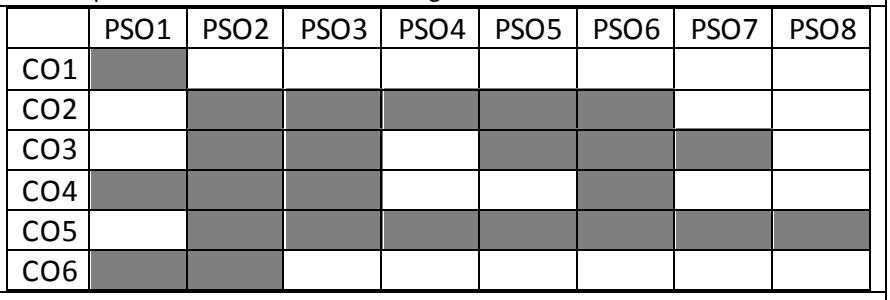
<table>
<tr><td>Course Code</td><td>102</td></tr>
<tr><td>Course Title</td><td>Object Oriented Programming Methodology</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs</td></tr>
<tr><td>Minimum weeks per Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Review / Revision</td><td>June 2022</td></tr>
<tr><td>Purpose of Course</td><td>This course introduces the concepts of object-oriented programming and skills necessary for developing programs in C++.</td></tr>
<tr><td>Course Objective</td><td>1. To make students understand concepts of object-oriented paradigm
2. To make students develop C++ programs
3. To make students learn capabilities of an object-oriented programming language</td></tr>
<tr><td>Course Outcome</td><td>CO1- Articulate the principles of Object Oriented Problem solving and programming.
CO-2-To demonstrate the differences between traditional imperative design and object Oriented Design
CO-3-Outline the essential features and elements of C++ programming language.
CO-4- To grasp and apply the concepts of class, method, constructor, abstraction, inheritance and Static Polymorphism.
CO-5- To understand and apply Dynamic Polymorphism in real world applications.
CO-6- To implement Genericity through the usage of Templates.
Mapping between Cos and PSOs</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><th></th><th>PSO1</th><th>PSO2</th><th>PSO3</th><th>PSO4</th><th>PSO5</th><th>PSO6</th><th>PSO7</th><th>PSO8</th></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO6</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>C Programming</td></tr>
<tr><td>Course Content</td><td>Unit 1: C++ Basics
1.1 Data Types
1.2 Pointers
1.2.1 Pointer Arithmetic
1.2.2 Array of Pointers
1.2.3 Dynamic Array
1.3 ios Class
1.4 Input and Output
1.5 Manipulators

Unit 2: Introduction to Object Oriented Programming
2.1 Structure, classes and Objects
2.2 Encapsulation & Data Hiding
2.3 Constructors
2.4 Friend Functions
2.5 Inline Functions
2.6 Dynamic Object Creation & Destruction
2.7 Static Members
2.8 this Pointer
2.9 Destructors

Unit 3: Object Oriented Properties
3.1 Introduction to Object Oriented Properties
3.2 Abstraction</td></tr>
</table>

	3.3 Polymorphism 3.3.1 Operator Overloading 3.3.2 Function Overloading & Type Conversions 3.4 Inheritance 3.4.1 Types of Inheritance 3.4.2 Constructor & Destructor calls during Inheritance 3.5 Dynamic Polymorphism 3.5.1 Overriding 3.5.2 Virtual Functions 3.5.3 Abstract Class **Unit 4: Data Files and Exception Handling** 4.1 Streams 4.2 File Types and Modes 4.3 File Pointers & their manipulations 4.4 Sequential Input & Output operations 4.5 Random access 4.6 Error handling during File operations 4.7 Exception Handling **Unit 5: Generic Programming and C++ Standard Template Library (STL)** 5.1 Template Classes 5.2 Template Functions 5.3 Implementation of Object-Oriented Properties on Template Classes 5.4 STL 5.4.1 Algorithms 5.4.2 Containers 5.4.3 Functions 5.4.4 Iterators
Reference Books	1. The C++ Programming Language, Stroustrup, Addison Wesley 2. The Complete Reference C++, Schildt, Tata McGraw Hill 3. OOP in Turbo C++, Robert Lafore, Galgotia Publication 4. C++ Primer, Lippman, Addition Wesley 5. Object Oriented Programming with ANSI and Turbo C++, Kamthane, Pearson Education 6. Thinking in C++, Bruce Eckel, Pearson 7. Object Oriented Modelling & Design, Rumbaugh, PHI 8. Object Oriented Analysis & Design with Application, Grady Booch, LPE 9. Standard C++ with Object Oriented Programming, Paul S. Wang, Thomson 10. C++ Primer Plus, Stephan Prata, Addison Wesley 12. Programming with ANSI C++, Bhushan Trivedi, Oxford University Press
Teaching Methodology	Class Work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment 70% External Assessment

Course: 103: Cloud Computing

Course Code	**103**
Course Title	**Cloud Computing**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	The purpose of the course is to make student capable of implementing the concepts, methods and tools of Cloud Computing
Course Objective	The objective of the course is to provide comprehensive and in-depth knowledge of Cloud Computing Concepts, technologies, architecture, applications and implementation.
Course Outcome	CO1: Explain students about the cloud and cloud computing, History & Evolution, Properties & Characteristics, Advantages & Disadvantages of cloud computing. CO2: Explain students about various service models and deployment models CO3: To provide students a foundation of different cloud service models IAAS,PAAS and SAAS so that they are able to use Cloud Computing Services in real world problem CO4: Understanding the concepts of cloud infrastructure security, data security and storage, Access control and authentication in cloud. CO5: Train students to use AWS and Microsoft Azure CO6: Explain students in brief about BigTable and Firebase
Mapping between COs with PSOs	
Pre-requisite	Basics of DBMS, Web Development & HTML, Networking
Course Out come	To give basic knowledge of cloud computing, its architecture and its benefits and how to deploy applications on well-known cloud platforms
Course Content	**Unit 1: Introduction to Cloud & its architecture** 1.1 Introduction & Definitions 1.2 Cloud Computing (NIST) 1.2.1 History & Evolution 1.2.2 Properties & Characteristics 1.2.3 Advantages & Disadvantages 1.3 Cloud Architecture overview **Unit 2: Cloud Computing Models** 2.1 Cloud computing Stack 2.1.1 Comparison with traditional architecture 2.2 Service Models 2.2.1 Infrastructure as a Service (IaaS) 2.2.2 Platform as a Service (PaaS) 2.2.3 Software as a Service (SaaS) 2.3 Deployment Models 2.3.1 Public Cloud 2.3.2 Private Cloud 2.3.3 Hybrid Cloud

	2.3.4 Community Cloud **Unit 3: Cloud Service Models** 3.1 Infrastructure as a Service (IAAS) 3.1.1 Introduction to Virtualization 3.1.1.1 Hypervisors, Virtual Machine, Machine Image 3.1.2 Resource Virtualization 3.1.2.1 Server, Storage, Network 3.1.3 Amazon EC2, Eucalyptus 3.2 Platform as a Service (PAAS) 3.2.1 Introduction to SOA 3.2.2 Cloud Platform 3.2.2.1 Computing 3.2.2.2 Storage 3.2.3 Introduction to Microsoft Azure 3.2.4 Introduction to Salesforce's Force.com 3.3 Software as a Service (SAAS) 3.3.1 Introduction 3.3.2 Web Service & Web OS **Unit 4: Cloud Security** 4.1 Infrastructure Security 4.2 Data Security and Storage 4.3 Identity and Access Management (IAM) 4.4 Access Control 4.5 Authentication in Cloud **Unit 5: Cloud Databases (DBaaS)** 5.1 AWS SimpleDB & RDS 5.2 AzureTable Service & SQL Azure 5.3 Introduction to BigTable 5.4 Introduction to Firebase
Reference Books	1. Cloud Computing Principles and Paradigms (Wiley), Rajkumar Buyya, James Broberg, Andrzej M. Goscinski 2. Cloud Computing: Principles, Systems and Applications, Nikos Antonopoulos, Lee Gillam (Springer) 3. Enterprise Cloud Computing: Technology, Architecture, Applications Gautam Shroff - Cambridge University Press 4. Cloud and Virtual Data Storage Networking, Greg Schulz - Auerbach 5. Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L Krutz, Russel Dean Vines (John Wiley & Sons) 6. Cloud Computing , (David Crookes - TMH Education) 7. Cloud Computing Bible, Barrie Sosinsky (Wiley India) 8. Cloud Computing: Implementation, Management and Security, (James F Ransome, John W Rittinghouse - CRC Press) 9. Amazon Cloud Computing with Java, (Aditya Yadav - Lulu.com) 10. Grid and Cloud Database Management, Fiore, Sandro, Aloisio, Giovanni - Springer 11. Building a Database Cloud for Dummies, Michael Wessler John Wiley & Sons
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 104: Computer Network

Course Code	**104**
Course Title	**Computer Network**
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	This course aims towards learning fundamentals of computer network. The course teaches students about the various network technologies and popular network protocols
Course Objective	1.To make students learn about computer network fundamentals 2.To make students familiar with services offered at each layer of the network protocol stack 3.To make students learn various protocols at data link layer, network layer, and transport layer of network.
Course Outcome	CO1: Understand students the fundamental aspects of the computer networks. CO2: Explain and help students to learn fundamentals network protocols at data link layer, network layer and transport layer. CO3: Explore students the services offered at each layer of the network protocol stack. CO4: Train students to implement various error control, flow control, routing algorithms and security algorithms fall under data link layer, network layer and transport layer. CO5: Explore students the concepts of Security, digital certificate, Public key Infrastructure, and similar security schemes.
Mapping between COs with PSOs	
Pre-requisite	Nil
Course Content	**Unit 1: Introduction to Data Communication** 1.1 Introduction to networks, Internet and its application 1.2 Network Structure 1.3 Network Architecture 1.4 The OSI Reference model & services 1.5 The TCP/IP Reference model and Comparison with OSI Model 1.6 Concepts of data transmission 1.6.1 Guided and unguided Transmission media. PSTN 1.7 Multiplexing & switching techniques 1.8 ISDN (Integrated Service Digital Network) **Unit 2: Data Link Layer** 2.1 MAC Sub layer 2.1.1 Multiple Access Protocols 2.1.2 Ethernet 2.1.3 LAN protocols & IEEE standards for LAN 2.1.4 Fibre Optic & Satellite networks 2.2 Data Link Layer protocols 2.3 Error detection & correction **Unit 3: Upper Layers** 3.1 Network 3.1.1 Routing Algorithms

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2	■							
CO3	■							
CO4	■	■	■	■				■
CO5		■	■	■		■	■	■

	3.1.2 Congestion Control Algorithm 3.1.3 Internetworking 3.2 Transport Layer 3.2.1 Connection Management 3.3 Concepts of Session Layer **Unit 4: The Presentation Layer** 4.1 Data Compression Technique 4.2 Cryptography 4.3 Symmetric Key Algorithms 4.4 Public – Key Algorithms & management of Public Keys 4.5 Digital Signatures and Communications security **Unit 5: The Application Layer** 5.1 Electronic Mail 5.2 Virtual Terminals General Purpose Applications [Self Study] Virtual LAN
Reference Books	1. Networking Complete- 1st Edition 2002, BPB Publication (Text Book) 2. Data Communication and Networking: Forouzan, TMH 3. Computer Networks - A. S. Tanenbaum - Prentice-Hall 4. Computer Networks and Distributed Processing - Martin J. - Pretice-Hall 5. Local Area Networks: An Introduction - Stalling, William - Mc-Millan Publishing Co. 6. Computer Networks: Protocols, Standards and Interfaces - Black – Prentice-Hall 7. Data Networks: Concepts Theory and Practices - Black - PHI 8. N/W Architecture - Comer - Prentice-Hall
Teaching Methodology	Class Work, Discussion, Self Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment 70% External Assessment

Course: 105: Data Structures and Design and Analysis of Algorithms

Course Code	**105**
Course Title	**Data Structures and Design and Analysis of Algorithms**
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	This course introduces the various data structures and algorithms involving these data structures and their logical implementation. Students also will be able to understand complex data structures like trees and their use in various applications
Course Objective	1. To learn fundamental data structures like arrays, stacks, lists. 2. To learn complex data structures like trees. 3. To learn and compare various sorting techniques. 4. To learn analysis of algorithms
Course Outcome	CO1. Explain the futures of various non-primitive data structure and their applications. Train students for algorithms to create, insert, delete and traversing various data structure. CO2. Explain the basics of computational complexity analysis and big O & theta notation that help for fundamental research in algorithm analysis. Students can select the efficient algorithm after analyzing a problem and identifying it's the computing requirements. CO3. Explain insight of basic technique for algorithm such as Divide & Conquer, Greedy method, Backtracking & Branch & Bound CO4. Students will be able to work with various searching and sorting techniques and compare its computational complexity. CO5. Train students with to work with Hash structure, proper hash method and collision detection techniques.
Mapping between COs with PSOs	
Pre-requisite	C programming Language
Course Content	**Unit 1: Non-Primitive (Linear) Data structures** 1.1 Arrays - Storage structures & operations 1.2 Stacks - operations and its applications in Recursion, Polish expressions etc. 1.3 Queues - Types of queues, operations and applications. 1.4 Linked lists - Types of linked list, operations and applications. **Unit 2: Non-Primitive (Non - Linear) Data structures** 2.1 Trees - Concept and Definitions, Operations, linked & threaded storage Representation of Binary Trees. 2.2 Applications of Trees - The manipulation of Arithmetic expressions, Symbol-table construction, Syntax Analysis etc. **Unit 3: Algorithm Analysis & Basic techniques for algorithms** 3.1 Analysis of Algorithms - Asymptotic notation - Big-O 3.2 Basic techniques of algorithms 3.2.1 Divide & Conquer method 3.2.2 Greedy method

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■	■	■	■			■	
CO2		■			■			■
CO3	■	■	■	■		■	■	■
CO4	■	■	■	■			■	
CO5	■	■						

	3.2.3 Backtracking 3.2.4 Branch & Bound **Unit 4: Searching and Sorting** 4.1 Searching techniques 4.1.1 Sequential search 4.1.2 Binary Search 4.2 Search Trees – 4.2.1 Binary Search Tree 4.2.2 Height Balanced Tree – AVL tree, 2:3 tree 4.2.3 weight-balanced tree 4.3 Sorting 4.3.1 Internal sorting - Insertion, Selection, Quick, 2-way merge and Heap 4.3.2 External sorting - k-way merging, Balanced merge and poly phase Merge **Unit 5: Hashing** 5.1 Introduction to Hash Tables 5.2 Hash functions 5.2.1 Division method 5.2.2 Multiplication method 5.3 Collision resolution techniques 5.3.1 Open Hashing (Separate chaining) 5.3.2 Closed Hashing (Open Addressing) [Self Study] Graphs – Creation and Traversal
Reference Books	1. An Introduction to Data Structures with applications - Trembley – McGraw Hill 2. Theory and Problems of Data Structure – Lipschutz Semour – McGraw Hill 3. Algorithms + Data Structure Programs - Wirth, Niclaus - PHI. 4. Fundamentals of Data Structures,Horwitz, E. and Sahni S. - Computer Science Press. 5. The Art of Computer Programming, Vols. 1-2, Knuth D. - Addison Wesley. 6. Data Structures and Algorithms - Aho A.V., Hopcroft and Ullman - Addison Wesley 7. Data Structure & "C" Programming - Vanwyte C J - Addison Wesley. 8. Data Structures, Algorithms And Object Oriented Programming – Tata McGraw Hill edition Geogory L. Heileman. 9. Data Structures and the Standard Template Library - William J. Collins, Tata McGraw Hill edition. 10. Programming with C++ and Data Structures - Maria Litvin & Gary Litvin, Vikas Publishing House Pvt. ltd. 11. Data Structures using C & C++ - Y. Langsam Moshe J. Angensterin & A.M. Tenenbaum 12. Data Structures and Algorithms in C++ - Adam Drozdek, Thomson Learning 13. Data Structures & Program Design in C - Robert Kruse, C.L. Tondo, Brnceleing PHI Pvt Ltd. 14. Data Structures and Algorithms in Java, Lafore, Pearson 15. Data Structures and Algorithm Analysis in Java, Mark Allen Weiss, Pearson 16. Data Structures and Algorithms in Java, Micheal T Goodrich, Roberto Tamassia, Wiley
Teaching Methodology	Class Work, Discussion, Self Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment 70% External Assessment

Course: 106: Programming Skills - I

Course Code	**106**
Course Title	**Programming Skills – I**
Credit	5
Teaching per Week	5 Hrs
Minimum weeks per Semester	15 (Including lab work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	Give fundamental knowledge of Database Models, Oracle Database Server Architecture and Working knowledge of SQL & PL/SQL in Oracle.
Course Objective	To acquaint the students with Client Server Architecture in general and Oracle Architecture in particular. Also, to get working knowledge of SQL and PL/SQL programming
Pre-requisite	None
Course Outcome	After studying the course, students will be able to understand how Oracle Database works and the importance of various components of Oracle. This course will also help students to appreciate the role of a database administrator. After successful completion, students will be able to manage Oracle database and will be able to write codes in SQL & PL/SQL necessary for an application.
Course Content	Practical based on paper no: 101 (DBMS)
Reference Books	None
Teaching Methodology	Lab. Work
Evaluation Method	30% Internal assessment and 70% External Assessment

Course: 107: Programming Skills - II

Course Code	**107**
Course Title	**Programming Skill – II**
Credit	3
Teaching per Week	3 Hrs
Minimum weeks per Semester	15 (Including lab work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	This course helps students practically implement the concepts of object oriented programming using C++.
Course Objective	1. To make students practically learn concepts of object-oriented paradigm 2. To make students develop and code C++ programs.
Pre-requisite	C Programming
Course Outcome	After studying the course, students will be able to practically solve common and complex programming problems using object-oriented paradigm. This course also helps students learn practical implementation of data files and operations upon them using object-oriented approach.
Course Content	Practical based on paper no: 102 (OOPM)
Reference Books	None
Teaching Methodology	Lab. Work
Evaluation Method	30% Internal assessment and 70% External Assessment

Course: 108: Programming Skills - III

Course Code	**108**
Course Title	**Programming Skill – III**
Credit	3
Teaching per Week	3 Hrs
Minimum weeks per Semester	15 (Including lab work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	This course introduces the various data structures and algorithms involving these data structures and their practical implementation using JAVA programming language. Students also will be able to understand and write JAVA programs using complex data structures like trees.
Course Objective	1. To practically learn implementation of fundamental data structures like arrays, stacks, lists using JAVA 2. To learn implementing complex data structures like trees using JAVA 3. To learn and compare various sorting techniques practically 4. To learn analysis of algorithms practically.
Pre-requisite	C programming Language
Course Outcome	After studying the course, students will be able to use data structures and their application in sorting, searching and comparison of algorithms. Students will also learn analysis of the algorithms.
Course Content	Practical based on paper no: 105 (Data Structures and Design and Analysis of Algorithms)
Reference Books	None
Teaching Methodology	Lab. Work
Evaluation Method	30% Internal assessment and 70% External Assessment

MCA 2nd Semester

Course: 201: Artificial Intelligence

Course Code	**201**
Course Title	**Artificial Intelligence**
Credit	4
Teaching per Week	4 Hrs
Medium of Instruction	English
Minimum weeks per Semester	15 (Including Classwork, examination, preparation, holidays etc.)
Effective From	June 2022
Purpose of Course	The purpose of the course is to make the student capable of implementing the concepts, methods, and tools of Artificial Intelligence and learn their implementation in Knowledge-Based Systems Course Objective To acquaint students with concepts of Artificial Intelligence and its applications.
Course Objective	To make students acquainted with concepts of Artificial Intelligence and its applications.
Course Outcome	CO1: Explain students the insight of the historical and fundamental aspects the artificial intelligence. CO2: Train students to represent declarative knowledge in the form of symbolic knowledge through various Knowledge Representation (KR) techniques like First Order Predicate Logic (FOPL), Semantic Network, Conceptual Graphs, Scripts, and Frames. CO3: Train students to apply various searching algorithms fall under informed and uninformed search methods to solve complex problem of AI domain. CO4: Explain and train students to deal with the uncertainty that inherently lies within many AI problem. CO5: Expose the students with the analysis and development process of the knowledge based system development. CO6: Explain students to utilize the AI problem solving techniques in the advanced AI problem domain like Natural Language Processing (NLP) and Computer Vision (CV)
Mapping between COs with PSOs	
Pre-requisite	Basics of Mathematics, Data Structures
Course Content	**Unit 1: Introduction to AI** 1.1 Introduction to Artificial Intelligence 1.1.1 Definition of Artificial Intelligence 1.1.2 History of Artificial Intelligence 1.1.3 Application of Artificial Intelligence 1.1.4 Introduction to Knowledge-Based System 1.2 Turing Problem 1.3 Knowledge Representation 1.3.1 knowledge and Knowledge Base 1.3.2 First Order Predicate Logic (FOPL) 1.3.3 Inference Rules **Unit 2: Knowledge Representation and Searching Techniques** 2.1 Structured Knowledge Representation 2.1.1 Associative network and Conceptual graphs 2.1.2 Frames and Scripts 2.1.3 Conceptual Dependencies 2.2 Searching

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2	■		■					
CO3	■	■	■		■			
CO4		■	■	■		■	■	■
CO5	■		■		■	■		
CO6	■	■		■		■	■	■

	2.2.1 Search Problem 2.2.2 Initial State, action, transition model, goal test, the cost function 2.2.3 Uninformed Search 2.2.3.1 Depth First Search 2.2.3.2 Breadth-First Search 2.2.3.3 Iterative Deepening Search 2.2.4 Informed Search 2.2.4.1 Heuristics 2.2.4.2 A* Search 2.2.4.3 Minimax 2.2.4.5 Hill-Climbing Method 2.2.4.6 Constraint Satisfaction Search **Unit 3: Probability using AI** 3.1 Uncertainty 3.1.1 Probability 3.1.2 Conditional Probability 3.1.3 Baye's Rule 3.1.4 Joint Probability 3.1.5 Probability Rules 3.2 Introduction to Hidden Markov Model **Unit 4: Knowledge Acquisition** 4.1 Introduction to Knowledge Acquisition 4.1.1 Knowledge gathering 4.1.2 Learning Models 4.1.2.1 Introduction to Supervised Learning 4.1.2.2 Introduction to Unsupervised Learning 4.1.2.3 Reinforcement Learning 4.1.3 Performance of Learning Model **Unit 5: Expert System** 5.1 Introduction to Expert System 5.2 Characteristics of Expert System 5.3 Architecture of Expert System 5.4 Application of AI in Natural Language Processing 5.5 Application of AI in Computer Vision
Reference Books	1. Artificial intelligence, 3rd Edition, Kevin Knight, Elaine Rich, B. Shivashankar Nair, McGraw Hill 2. Russell Stuart Jonathan and Norvig Peter, Artificial Intelligence: A Modern Approach, 3rd Edition, Prentice-Hall, 2010 3. A First Course in Artificial Intelligence, Deepak Khemani, McGraw Hill 4. Introduction to artificial intelligence, Akerkar, Rajendra, PHI Learning 5. Foundation of Artificial Intelligence and Expert Systems by V.S. Janakiraman, K. Sarukesi, P. Gopalakrishnan, Mc Millan 6. Expert Systems Principles and Programming (3rd Edition) by Giarratano & Riley, Thomson (Vikas Publishing House)
Teaching Methodology	Classwork, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment based on class attendance, participation, class test, quiz, assignment, seminar, internal examination, etc. 70% External - based on semester end University examination

Course: 202: Front-End Technologies

Course Code	**202**
Course Title	**Front-End Technologies**
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	To understand the concepts of HTML, CSS, Front-end Scripting technologies
Course Objective	To teach the concepts of Front-end Scripting and its practical applications.
Course Outcome	CO1: Explain students the insight of the fundamental aspects of the static web technology like HTML, CSS and JavaScript CO2: Train students to represent declarative knowledge in the Bootstrap and its components, with help of this students can learn static web designing. CO3: Train students to understand various JavaScript based technologies like JQuery, JSON and AJAX. CO4: Explain and train students to understand basic concepts of Angular JS. CO5: Expose the students to the various UI technology and real world use of it. CO6: After completion of this course, the student will be able to design and develop web pages and Interactive UI for Web Applications
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2	■		■					
CO3	■	■	■		■			
CO4		■	■	■		■	■	■
CO5	■		■		■	■		
CO6	■	■		■		■	■	■

Pre-requisite	Basic concepts of HTML, Web & Programming skills
Course Content	**Unit 1: Fundamentals of Web Technology** 1.1 HTML 1.1.1 Basic HTML tags 1.1.2 HTML Forms 1.2 HTML 5 1.2.1 HTML5 new elements & semantic elements 1.2.2 HTML5 Form Input elements 1.2.3 HTML5 Attributes 1.2.5 Video and Audio 1.3 CSS 3 1.3.1 Introduction to CSS 3 1.3.2 Selectors and Classes, Pseudo Classes, Specificity Rules 1.3.3 Font and Text effect, Shadows, Transition & Transformation 1.3.4 Colours, Gradients, Background Images, and Masks 1.3.5 Box Model, Positioning, Float, Border and Box effects etc. 1.3.6 Embedding Media, Flex-box, CSS Grid **Unit 2: JavaScript** 1.4 JavaScript 1.4.1 Fundamentals of JavaScript, Syntactic Structure 1.4.2 Use of JavaScript in HTML, Basic Validations 1.4.3 Objects, Functions, Call-backs 1.4.4 DOM Events 1.4.5 JS Functions : Bind, Closures 1.4.6 JS Async: Callback, Promises, Async/ Await 1.4.7 Web storage, Geolocation 1.4.8 Canvas 1.4.9 ajax introduction

	Unit 3: Introduction to Bootstrap 2.1 What is Bootstrap 2.1.1 What is Responsive web page 2.1.2 Advantages and features of Bootstrap 2.1.3 Setup Environment 2.1.4 Apply bootstrap to Application 2.2 Bootstrap with CSS 2.2.1 Grid system 2.2.2 Typography 2.2.3 Code, table, forms, buttons, image, responsive utilities etc. 2.3 Bootstrap components 2.3.1 What is Bootstrap components 2.3.2 Advantages of components 2.3.3 Types of Bootstrap components 2.3.3.1 Glyphicons, Drop downs, button group, input groups navigation, pagination etc. **Unit 4: JQuery** 3.1 Introduction to JQuery 3.1.1 Syntax, Attributes, Selectors, Events 3.2 JQuery Effects 3.2.1 Hide/Show, Fade, Slide, Animation etc. 3.2.2 JQuery with HTML 3.3 Traversing 3.4 JQuery and AJAX 3.5 JSON 4.2.1 JSON Basics 4.2.2 Syntax 4.2.3 Datatype, Parse, Stringify, Objects, Array 4.2.4 Use of JSON using any one server site scripting **Unit 5: Introduction to Angular** 5.1 Introduction to Angular JS/ angular 5.2 Directives, Expressions, Controllers, Filters, Tables, Html DOM 5.3 Modules, Forms, Includes, Views 5.4 Angular SQL 5.5 AngularJS Applications
Reference Books	1. Responsive Web Design with HTML5 and CSS3 By Ben Frain - Packt Publishing Ltd. 2. HTML, CSS & JavaScript Web Publishing in One Hour a Day, Sams Teach Yourself by Laura Lemay, Rafe Colburn, Jennifer Kyrnin – Sams Publication 3. Training Guide Programming in HTML5 with JavaScript and CSS3 (MCSD): 70-480 by Glenn Johnson - Pearson Education 4. Learning Bootstrap by Aravind Shenoy, Ulrich Sossou - Packt Publishing Ltd. 5. Professional AngularJS by Valeri Karpov, Diego Netto - John Wiley & Sons 6. Ajax: Creating Web Pages with Asynchronous JavaScript and XML - Edmond Woychowsky - Prentice Hall
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 203: Programming in .NET

Course Code	**203**
Course Title	**Programming in .NET**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	This course is an introduction to students to understand fundamentals of .NET technology. The course also gives students an idea about C#.NET Programming.
Course Objective	1. To make students understand .NET Technology 2. To make students understand C#.NET Programming 3. To make students understand the importance of ASP.NET
Course Outcome	CO-1 Study .Net Architecture. CO-2- Design and develop console and window based .NET application. CO-3-To learn basic syntax and implement small applications in C# programming language. CO-4-Create and manipulate GUI components in C#. CO-5-Create applications in C# using Object Oriented Properties. CO-6-Design and implement Database connectivity using ADO.Net. CO-7-Identify and resolver problems in C# applications.
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■		■	■			■	
CO2	■		■	■		■	■	
CO3	■	■		■		■	■	
CO4	■			■		■		
CO5				■	■	■	■	
CO6			■	■		■	■	
CO7		■	■	■	■		■	■

Pre-requisite	Nil
Course Content	**Unit1: Overview of Microsoft .NET Platform** 1.1 Introduction to Building Blocks of .Net Platform 1.2 Overview of .Net Assemblies 1.3 Common Type System 1.4 Common Language Specification 1.5 Common Language Runtime 1.6 Exploring an Assembly (ildasm) 1.7 Platform Independent Nature of .Net 1.8 Base Class Libraries **Unit2: Overview of C#** 2.1 Literals, Variables, Data Types 2.2 Operators 2.3 Expressions and Looping 2.4 Constants, Arrays, Array Class, List 2.5 String, String Builder 2.6 Boxing and UnBoxing 2.7 Events, Errors and Exceptions **Unit3: Object Oriented Aspects of C#** 3.1 Creating Classes, Encapsulation, Object Construction & Destruction 3.2 Inheritance 3.3 Polymorphism 3.4 Abstraction 3.4 Interfaces and Abstract Classes 3.5 Delegates **Unit4: Application Development**

	4.1 Creating Windows Forms with Events and Controls 4.2 Menu Creation 4.3 Inheriting Windows Forms 4.4 SDI and MDI Application 4.5 Dialog Boxes (Modal and Modeless) 4.6 Validating Controls **Unit5: Accessing Data** 5.1 ADO.Net 5.1.1 Data Adapter 5.1.2 Data Set 5.1.3 Typed Data Set 5.2 Using Stored Procedures 5.3 Handling Exceptions 5.4 LINQ [Self Study] Report Generation, Deployment
Reference Books	1. .NET Framework Essentials, Hoand Lam, Thuan L. Thai, O'REILLY 2. Microsoft .NET Framework 4.5 Quickstart Cookbook, Jose Luis Latorre Millas, PACKT Publishing 3. Pro C# 5.0 and the .NET 4.5 Framework, Andrew Troelsen, Apress 4. C# IN DEPTH, Jon Skeet, Manning Publications 5. Beginning C# 7 Programming with Visual Studio 2017, Benjamin Perkins, wrox 6. Illustrated C#, Daniel Solis, Cal Schrotenboer, Apress 7. The C# Programmer's Study Guide, Ali Asad, Hamza Ali, Apress
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 204: Python Programming Language

Course Code	**204**
Course Title	**Python Programming Language**
Credit	4
Teaching per Week	4 Hrs
Medium of Instruction	English
Minimum weeks per Semester	15 (Including Classwork, examination, preparation, holidays, etc.)
Effective From	June 2022
Purpose of Course	The Python language is used popularly among the people working in the area of Machine Learning (ML), Data Analytics, Artificial Intelligence, Web Application, and even the people working on Desktop Applications. This course imparts to the students understanding of Python programming language.
Course Objective	1. To make students understand Python Language 2. To make students understand various components of language and its Working 3. To prepare students to understand the use of language in the area of AI, ML, Data Analytics etc.
Course Outcome	CO1: Students will be able to Write, Test and Debug Python Programs. CO2: Students will be able to Implement Conditionals and Loops, use functions and represent Compound data using Lists, Tuples and Dictionaries in Python programs. CO3: Students will be able to Read and write data from & to files in Python and develop Real World Application. CO4: Students will be able to Design and implement programs to solve real-world problems using Python Programming Language. CO5: Students will learn essential packages like NumPy and Matplotlib, which are necessary for Machine Learning, Data Analytics, and AI.
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2				■			■	
CO3			■	■		■	■	
CO4			■			■	■	■
CO5	■		■	■		■	■	

Pre-requisite	Basic programming skills
Course Content	**Unit 1: Fundamentals of Python** 1.1 Features of Python 1.2 Python's Integrated Development and Learning Environment (IDLE) 1.3 Python identifiers 1.4 Python Operators 1.5 Python Datatypes 1.5.1 Numeric: integer, float, complex 1.5.2 Sequence: list, tuple, range 1.5.3 Sets 1.5.4 Texts 1.5.5 Binary: bytes, bytearray 1.5.6 Iterator 1.5.7 Mapping: dictionary 1.6 Input/Output in Python **Unit 2: Program Logic** 2.1 Control Structures 2.1.1 If, if..else, nested if, shorthand if, shorthand if..else 2.1.2 while loop 2.1.3 for loop 2.1.4 break, continue, pass 2.2 Functions in Python 2.2.1 Function declaration

	2.2.2 Passing arguments to function 2.2.3 Return values 2.2.4 Variable scope and name space 2.2.5 Lambda function 2.2.6 Recursive function 2.2.7 In-built function **Unit 3: Files and Modules** 3.1 File handling 3.1.1 Reading and writing to a file 3.1.2 Creation of new file 3.1.3 Deletion of a file 3.2 Python Modules 3.2.1 Creation of module 3.2.2 Importing a module 3.2.3 Date & time module **Unit 4: Exceptions, Class and Objects** 4.1 Exception Handling 4.1.1 try, catch, finally 4.2.2 Multiple error handling: except 4.2.3 Throwing a particular error: raise 4.2 Classes and Objects 4.2.1 Creation of class and object 4.2.2 The __init__() function 4.2.3 Self parameter 4.2.4 Modifying the property of a class 4.2.5 Inheritance & Encapsulation **Unit 5: Python Packages** 5.1 NumPy 5.1.1 Installing NumPy 5.1.2 Numpy Array: dtype, shape, reshape, ndim, itemsize, empty, zeros, ones, fromiter, arrange, linespace 5.1.3 Indexing and slicing, broadcasting 5.1.4 Array manipulation: changing shapes, transpose, changing dimension, joining and splitting arrays, adding and removing elements 5.1.5 Mathematical functions and matrix library 5.2 Introduction to Matplotlib 5.2.1 Installing Matplotlib 5.2.2 Components of a plot 5.2.3 Drawing a plot 5.2.4 Drawing scatter diagram
Reference Books	1. Python Programming, Anurag Gupta, G Biswas,, Mc Graw Hill 2. Exploring Python, Timothy A. Budd, McGraw Hill Publication 3. Core Python Programming, R. Nageswara Rao, Dreamtech Press 4. Learning Python, 5th Edition, Mark Lutz, O'Reilly Media 5. Python Projects, Laura Cassell, Alan Gauld, Wrox Publication 6. NumPy: Beginner's Guide, 3rd Edition, Ivan Idris, Packt Publishing 7. NumPy Essentials, Leo Chin, Tanmay Dutta, Packt Publishing 8. Matplotlib 2.x By Example, Allen Yu, Claire Chung, Aldrin Yim, Packt Publishing
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment based on class attendance, participation, class test, quiz, assignment, seminar, internal examination, etc. & 70% External based on semester end University examination

Course: 204: Blockchain Technology

Course Code	**204**
Course Title	**Blockchain Technology**
Credit	4
Teaching per Week	4 Hrs
Medium of Instruction	English
Minimum weeks per Semester	15 (Including Classwork, examination, preparation, holidays, etc.)
Effective From	June 2022
Purpose of Course	This course will help the students to understand the fundamental as well as advanced concepts of Blockchain Technology. The course also provides them the skills necessary to develop a Blockchain Application from scratch and make them understand its implementation and utility in various sectors as well.
Course Objective	To provide conceptual understanding of Blockchain Technology. To provide the knowledge of how Blockchain Technology can be used to innovate and improve in various domains. To acquire the skills to develop Blockchain base solutions. To Study basics of Hyperledger and Web 3.0.
Course Outcome	CO1: Students will be able to Understand Blockchain Technology. CO2: Students will learn Importance of Security and Consensus in Blockchain. CO3: Students will be able to Develop a Blockchain base applications for a specific domain. CO4: Expose the students to various sectors where Blockchain Technology can be implemented and how it will bring the transformation. CO5: Explain students about concept of Hyperledger , web 3.0 and Decentralised applications (Dapps) development.
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■					■		
CO2	■	■				■		
CO3		■	■	■			■	■
CO4		■	■		■		■	
CO5	■			■		■	■	■

Pre-requisite	Understanding of Basic Object Oriented Programming Languages like C++ / C#
Course Content	**Unit 1: Introduction to Blockchain** 1.1 History of Blockchain 1.2 Types of Blockchain 1.3 Blockchain Generations 1.4 Structure of Blockchain : Block, Hash, Nonce 1.5 Characteristics of Blockchain and its Advantages **Unit 2: Cryptography and Consensus** 2.1 PKI and Cryptography 2.1.1 Private key Cryptography 2.1.2 Public key Cryptography 2.1.3 Cryptographic Hash Function 2.1.4 SHA – 256 2.1.5 Digital Signature 2.2 Consensus 2.2.1 Byzantine Fault 2.2.2 Proof of Work 2.2.3 Proof of Stake 2.2.4 Double-Spending **Unit 3: Building the Blockchain** 3.1 Deciding on the Blockchain Architecture 3.2 Creating Blocks and Links 3.3 Inserting Hashes 3.4 Implementing Consensus Mechanism

	3.5 Looking forward to Smart Contracts **Unit 4: Blockchain Use Cases in** 4.1 Financial Services 4.2 Healthcare 4.3 Government 4.4 Retail and Consumer **Unit 5: Web 3.0 and Hyperledger** 5.1 Introduction to Web 3.0 5.1.1 Development Frameworks 5.1.2 Decentralize Applications (DApps) 5.2 Hyperledger as a protocol 5.2.1 Reference Architecture 5.2.2 Hyperledger Fabric 5.2.3 Distributed Ledger
Reference Books	1. The Blockchain Developer : A Practical Guide for Designing, Implementing, Publishing, Testing and Securing Distributed Blockchain-based Projects- Elad Elrom, Apress 2. Mastering Bitcoin : Programming the Open Blockchain, Andreas M. Antonopoulos, O'REILLY 3. The Book of Satoshi : The Collected Writings of Bitcoin Creator, Phil Champangne, E53 Publishing LLC 4. Blockchain : The Blockchain for beginners guide to Blockchain technology and leveraging Blockchain programming, Josh Thompson, Create Space independent Publishing Platform 5. "Mastering Blockchain: Distributed Ledger Technology, Decentralization, and Smart Contracts Explained", Imran Bashir, Second Edition, Packt Publishing, 2018 6. "Block Chain: The Block Chain for Beginners – Guide to Blockchain Technology and Leveraging Block Chain Programming", Josh Thompsons 7. "Block Chain: Blueprint for a New Economy", Melanie Swan, O'Reilly, 2015
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment based on class attendance, participation, class test, quiz, assignment, seminar, internal examination, etc. & 70% External based on semester end University examination

Course: 205: iOS Development using Swift

Course Code	**205**
Course Title	**iOS Development using Swift**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks/ Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	This course will help the students to understand the fundamental as well as advanced concepts of iOS Programming. The course also provides them the skills necessary to develop an iOS Application from scratch to deploying it on the App Store.
Course Objective	The objective of the course is - 1. To understand the iOS ecosystem and tools for creating iOS applications 2. To explain advanced level concepts in iOS application design and development 3. To impart knowledge of Swift programming language
Course Outcome	CO1: Understand the iOS ecosystem and Xcode IDE. Understand the life cycle of iOS application and how to implement it in MVC. Understand Foundation framework in iOS. CO2: Understand the syntax, and semantics of the Swift programming language. Expose the students to CLI applications with Swift. CO3: Understand the UIKit framework in iOS. Understand the usage and working of UI elements in iOS application. Understanding various types of design and their implementation. CO4: Understand data persistence in mobile application. Understand working with files in iOS. Expose students with implementation and usage of database in an iOS application. CO5: Understand the usage and data extraction of sensors in iPhone. Expose the students with Location and MapKit Framework in iOS to build map-based applications. Expose the students with ad-hoc and App Store application deployment.
Mapping between COs with PSOs	
Pre-requisite	Knowledge of C, C++ and SQL
Course Content	**Unit 1: Introduction to iOS ecosystem** 1.1. Introduction to Xcode IDE 1.1.1. Environment setup 1.1.2. Editors, Storyboard and Simulator 1.2. Application Life-Cycle 1.3. View Controller Life-Cycle 1.4. Info.plist and App Permissions 1.5. MVC in iOS 1.6. Introduction to iOS App Frameworks 1.6.1. Foundation Framework 1.6.2. UIKit Framework 1.6.3. Swift and SwiftUI **Unit 2: Introduction to Swift Programming Language** 2.1. Simple Values – Constant and Variable 2.2. Control Flow 2.3. Functions and Closures 2.4. Objects and Classes 2.5. Enumerations and Structures 2.6. Protocols and Extensions 2.7. Error Handling 2.8. Generics

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1								
CO2								
CO3								
CO4								
CO5								

	Unit 3: UIKit: View Controllers, Views and Controls 3.1. Text Views: UILabel, UITextField, UITextView 3.2. Controls: UIButton, UIDatePicker, UIPageControl, UISegmentedControl, UISlider, UIStepper, UISwitch 3.3. Content Views: UIActivityIndicatorView, UIImageView, UIPickerView, UIProgressView 3.4. Bars: UINavigationBar, UISearchBar, UIToolbar, UITabBar 3.5. Images and Video: UIImagePickerController 3.6. Container View Controllers: UINavigationController, UITabBarController 3.7. Container Views: Table Views, Collection Views 3.8. Alerts: UIAlertController 3.9. Gestures: UITapGestureRecognizer, UIPinchGestureRecognizer, UIRotationGestureRecognizer, UISwipeGestureRecognizer, UIPanGestureRecognizer **Unit 4: Data Persistence and Networking** 4.1. UserDefaults 4.2. FileManager 4.3. SQLite Framework 4.4. Core Data Framework 4.5. JSON Parsing 4.6. Working with URL and URL classes **Unit 5: App Services and App Deployment** 5.1. Core Motion – Accelerometer, Gyroscope, Pedometer, Magnetometer, Altitude 5.2. Core Location – CLLocationManager, CLLocation, Authorization 5.3. MapKit – Map Fundamentals, Map Coordinates, Annotations and Overlays 5.4. How to deploy an Ad-Hoc app – (diawi) 5.5. Publishing an app to the AppStore
Reference Books	1. Apple Documentation [developer.apple.com/documentation] 2. The Swift Programming Language by Apple Inc. [swift.org/documentation] 3. Hacking with Swift by Paul Hudson [hackingwithswift.com] 4. iOS 13 Programming Fundamental with Swift by Matt Neuberg, O'Reilly 5. Programming iOS 13 by Matt Neuberg, O'Reilly 6. Mastering Swift 5: Deep dive into the latest edition of the Swift programming language, 5th Edition, Packt Publishing Limited 7. Swiftui Essentials - IOS Edition: Learn to Develop IOS Apps Using Swiftui, Swift 5 and Xcode 11 by Neil Smyth, Payload Media, Inc. 8. Beginning iOS 13 & Swift App Development: Develop iOS Apps with Xcode 11, Swift 5, Core ML, ARKit and more by Greg Lim 9. Pro iPhone Development with Swift 5: Design and Manage Top Quality Apps by Wallace Wang, Apress
Teaching Methodology	Class work, Discussion, Self-study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 205: Android Application Programming

Course Code	**205**
Course Title	**Android Application Programming**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	The purpose of the course is to make student capable of implementing the concepts, methods and tools of mobile applications development using Android.
Course Objective	To provide a thorough introduction to the Android environment and tools for creating Android applications.
Pre-requisite	Basic concepts of Operating Systems, Programming skills in core Java and Knowledge of object-oriented programming is desirable. Knowledge of XML format is helpful.
Course Outcome	CO1: Train students for installing and using the Android Developer's Toolkit such as SDK Manager, Android Virtual Device, Dalvik Debug Monitor Service (DDMS), Android Debug Bridge (ADB) and make them capable to develop, manage and maintain application(Apps) using Android and publish Apps on Google Play. CO2: Understand the Android software stack & program building blocks like activities, services and notifications to use them effectively to develop Android applications. CO3: Explain working with AndroidManifest, xml resources like layout and values and incorporate xml resources with Java code. CO4: Explain the use of java library for views, controls, menus, dialogs, graphics, media, storage, SQLiteDatabase etc… to make interactive applications. CO5: Train students to build Android app that access the database & other resources on web server using web services. CO6: Train students for managing Telephony and Message services. Make them capable of using Location Manager and Google MAP related APIs for App and create the professional applications.
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1								
CO2								
CO3								
CO4								
CO5								
CO6								

Course Content

Unit 1: Introduction to Android

1.1 Evolution of Android and OHA
1.2 Architecture of Android OS
1.3 Introduction to Android SDK
1.4 Android Development tools: SDK Manager, Android Emulator, Android Virtual Device, Dalvik Debug Monitor Service (DDMS), Android Debug Bridge (ADB)
1.5 Anatomy of Android App: AndroidManifest.xml, Resources & R.java, Assets, Layouts & Drawable Resources

Unit 2: Working with User Interface in with Android Activity

2.1 Widgets: Button, ImageButton, EditText, CheckBox, ToggleButton, RadioButton, RadioGroup, ProgressBar, Text Fields, ListView, Spinner
2.2 Designing UI Layouts: LinearLayout, RelativeLayout, TableLayout
2.3 Toast and Dialogs: AlertDialogs, TimePicker, DatePicker
2.4 Menus: Option menu, Context menu
2.5 Adapters for data binding: Array adapter, Cursor adapter
2.6 Event callback methods: onClick(), onLongClick() and onTouch()

Unit 3: Android Application Components

	3.1 Activity and Activity lifecycle 3.2 Intents and Intent Filters 3.3 Implicit Intent and Explicit Intent 3.4 Linking of Activity using Intent: startActivity() & startActivityForResult() 3.5 Fragments 3.6 Introduction to Service: life cycle, creation and destroy 3.7 Broadcast receiver & notification **Unit 4: Data Persistency in Android** 4.1 Shared preferences 4.2 File I/O Access: internal and external files 4.3 Working with SQLite Database - Performing insert, update, delete and query operations 4.4 Data access through web services (external databases) 4.5 Working with inbuilt Content Provider: CallLogs, Contacts **Unit 5: Telephony APIs, Sensors and Leveraging Google APIs** 5.1 Telephony APIs 5.1.1 Working with Telephony utilities 5.1.2 Sending and receiving SMS 5.2 Location and Map 5.2.1 Incorporating Location APIs 5.2.2 Incorporating Google map 5.2.3 Geocoding and reverse Geocoding
Reference Books	1. Beginning Android 4 Application Development, WEI-MENG LEE, WROX Publication-Wiley-India 2. Professional Android 4 Application Development by Reto Meier WROX Publication-Wiley-India 3. Android Programming Unleashed, B.M. Harwani, Sams Publishing 4. Beginning Android 4 Onur Cinar Apress Publication 5. Advanced Android Application Development, Fourth Edition, By Shane Conder, Lauren Darcey, Joseph Annuzzi Jr., Pearson
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 206: Programming Skills IV

Course Code	**206**
Course Title	**Programming Skills IV**
Credit	2
Teaching per Week	2 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	To understand the concepts of HTML, CSS, Front-end Scripting technologies
Course Objective	To teach the concepts of Front-end Scripting and its practical applications
Pre-requisite	Basic concepts of HTML, Web & Programming skills
Course Outcome	After completion of this course, the student will be able to design and develop web pages and Interactive UI for Web Applications
Course Content	Practical based on paper no 202 (Front End Technologies)
Reference Books	None
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment & 70% External Assessment

Course: 207: Programming Skills V

Course Code	**207**
Course Title	**Programming Skills V**
Credit	2
Teaching per Week	2 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	This course is an introduction to students to understand fundamentals of .NET technology. The course also gives students an idea about C#.NET Programming. The course also explains the concept of C#.NET
Course Objective	1. To make students understand .NET Technology 2. To make students understand C#.NET Programming 3. To make students understand the importance of C#.NET
Pre-requisite	Object Oriented Programming methodology
Course Outcome	After studying the course, students will be able to understand how .NET Technology works and the importance of object-oriented programming. This course will also help students to appreciate the C#.NET programming.
Course Content	Practical based on paper no 203 (Programming in .NET)
Reference Books	None
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment & 70% External Assessment

Course: 208: Programming Skills VI

Course Code	**208**
Course Title	**Programming Skills VI**
Credit	2
Teaching per Week	2 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	The purpose of the course is: • To make student capable of implementing the concepts, methods and tools of Python programming language OR • To make student capable of implementing the concepts and algorithms apart from developing Blockchain from scratch.
Course Objective	To provide a thorough introduction to the Python Programming Language/Blockchain Technology and application development.
Pre-requisite	Basic object-oriented programming skills
Course Outcome	After completion of this course, the student will be capable to develop applications using Python Programming Language/Blockchain Technology.
Course Content	Practical based on paper no 204 (Python Programming Language) OR Practical based on paper no 204 (Blockchain Technology – practical implementation in C++ or C#.Net)
Reference Books	None
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment & 70% External Assessment

Course: 209: Programming Skills VII

Course Code	**209**
Course Title	**Programming Skills VII**
Credit	2
Teaching per Week	2 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2022
Purpose of Course	The purpose of the course is to make student capable of implementing the concepts, methods and tools of mobile applications development using Android/iOS.
Course Objective	To provide a thorough introduction to the Android/iOS environment and tools for creating Android/iOS applications.
Pre-requisite	Basic concepts of Operating Systems, Programming skills and knowledge of object-oriented programming is desirable
Course Outcome	After completion of this course, the student will be capable to develop, manage and maintain mobile device-based application using Android/iOS.
Course Content	Practical based on paper no. 205 (iOS Programming using Swift / Android Application Programming)
Reference Books	None
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment & 70% External Assessment