GH raisoni COLLEGE
Engineering and Management
Nagpur

Department of InformationTechnology

B. Tech. in InformationTechnology

with Multidisciplinary Minor

UG Scheme & Curriculum - 2023
(NEP 2020)

UG Scheme-2023 (NEP-2020)

Course Code	Name of Course	Course Category	Teaching Scheme				Credits	Evaluation Scheme					
								Theory			Practical		Total Marks
			L	T	P	Total Hours		TAE	CAE	ESE	INT	EXT	
SEMESTER-IV													
UITL201 UITP201	Object Oriented Programming	PCC4	3		2	5	**4**	10	15	50	25	-	100
UCSL205 UCSP205	Design and Analysis of Algorithms	PCC5	3	-	2	5	**4**	10	15	50	25	25	125
UCSL306/ UNXXLXXX	Multi-disciplinary Minor-II Software Engineering and Project Management	MDM2	2	-	-	2	**2**	10	15	50	-	-	75
23BTITL211	Open Elective-II Basics of Game Programming	OE 2	2	-	-	2	**2**	10	15	50	-	-	75
UCSL307/ UCSP307	Web Development	VEC2	2		4	6	4	10	15	50	25	-	100
TOTAL			**12**	**0**	**8**	**20**	**16**	**50**	**75**	**250**	**75**	**25**	**475**

Course Title: **Object Oriented Programming**										
Semester	IV	Teaching Scheme				Evaluation Scheme				
						Theory			Practical	
Term	EVEN	Th	Tu	Pr	Credits	TAE	CAE	ESE	INT	EXT
Course Category	C	3 Hrs.	-	2 Hrs.	4	10	15	50	25	-
Course Code	23BTITL203									
Teaching Mode	Online	5 Hrs.			TOTAL	75			25	
Duration of ESE	2.5 Hrs.					100				

Course Objectives	1. This course introduces student's general idea and concepts of object-oriented programming.
	2. It is also aimed at developing skills to implement these concepts.
	3. The course provide carrier opportunities in design of some applications as object-oriented concepts plays dominant role in software development
Course Outcomes	Up on successful completion of this course, student will be able to:
	CO1: Articulate the principles of object-oriented programming using C++
	CO2: Apply function overloading, constructor overloading, operator overloading & its uses in programming
	CO3: Implement inheritance and polymorphism concepts and its use for application development
	CO4: Implement static and dynamic memory allocation for software development
	CO5: Develop generic programming applications using templates

Mapping of Course Outcomes with Program Outcomes and Program Specific Outcomes:

Course Outcomes	Program Outcomes and Program Specific Outcomes														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO 2	PSO3
CO1	--	3	--	--	--	--	--	--	--	--	--	--	2	--	--
CO2	--	--	3	--	--	--	--	--	--	--	--	--	3	--	--
CO3	--	--	2	2	--	--	--	--	--	--	--	--	3	--	--
CO4	--	--		2	--	--	--	--	--	--	--	--	2	--	--
CO5	--	--		2	--	--	--	--	--	--	--	--	2	--	--

Course Contents:

Unit	Contents	Hours
I	**Principles Of Object-Oriented Programming:** Differences between C and C++.A look at procedure Oriented programming, object-oriented programming paradigm, basic concepts of OOP, Benefits of OOP, OO languages, A sample program, structure of C++ program. Introduction to OOPS: The origins of C++, what is Object Oriented Programming? Some C++ fundamentals, Headers & Name Spaces, Introducing C++ Classes, Function overloading, Operator overloading, Inheritance, Constructors & Destructors, Function & Operator Overloading:	8
II	**Overloading** constructor functions, Localizing variables, Function overloading & Ambiguity, Finding the address of an overloaded function, this Pointer, Operator overloading, References, Using reference to overload a unary operator, Overloading [], overloading ( ), Applying operator overloading.	8
III	**Inheritance, Virtual Functions and polymorphism** Inheritance and the access specifies, Constructors and Destructors in derived classes, Multiple Inheritance, Passing parameters to a basic class, Pointers and references to derived types, Virtual Functions, Why virtual functions?, Pure virtual functions and abstract types, Early Vs Late binding.	8
IV	**Static & Dynamic allocation** Static & Dynamic allocation using new and delete, static class members, Virtual base classes, const member functions and mutable, volatile member functions, Using the asm keyword, linkage specification, The. * And ->* operators, creating conversion functions, copy constructors, Granting access, namespaces, Explicit constructors, type name and export.	8
V	**Templates & Exception Handling** Class templates, class templates with multiple parameters, function templates, function templates with multiple parameters, File Handling, Exception Handling, fundamentals, options the uncaught exception ( ), Applying exception Handling, and RTTI, casting operators, Recent trends in Object Oriented Programming in C++ , Advanced topics & its Application	8

Text Books	1.	Object Oriented Programming in C++ -Robert Lafore, edition, Galgotia publications
	2.	The Complete Reference C++, Herbert Schildt, 4th Edition, TMH
E--Books	1.	Cryptography & network security by atul kahate https://www.pdfdrive.com/cryptography-network-security-by-atul-kahate-e124796757.html
Reference Books	1.	Let's C++ by Y. Kanetkar, BPB publications
	2	Object oriented programming with C++, E Balagurusamy, 4th edition, TMH

Sr. No.	Name of Experiments / Mini Projects/ Case Studies
1	Write a program to compute the area of triangle and circle by overloading the area () function.
2	Define a class to represent a bank account. Include the following members: Data members: - Name of depositor, Account number, Type of account, Balance amount in the account Member functions: - To assign initial values, To deposit an amount, To withdraw an amount after checking the balance, To display name & balance Write a main program to test program using class and object.
3	Create two classes DM and DB which stores values of distances. DM stores distances in meters and centimeters and DB in feet and inches. Write a program that can read values for the class objects and add one object of DM with another object of DB. Use a friend function to carry out addition operation.
4	Create a class MAT of size m * n. Define all possible matrix operations for MAT type objects
5	Create Stud class to display student information using constructor and destructor. (Default constructor, Multiple constructors, Copy constructor, Overloaded constructor)
6	Consider class network of given figure. The class master derives information from both account and admin classes which in turn derive information from the class person. Define all the four classes and write a program to create, update and display the information contained in master objects.
7	A book shop sells both books and video tapes. Create a class media that stores the title and price of the publication. Create two derived classes, one for storing number of pages in the book and another for storing playing time of tape. A function display () must be defined in all classes to display class contents. Write a program using polymorphism and virtual function.
8	Write a program to show use of this pointer, new and delete.
9	Write a function template for finding the minimum value contained in an array
10	Write a program containing a possible exception. Use a try block to throw it and catch block to handle it properly.
Open Ended Experiments / New Experiments	
1	Write a class template to represent a generic vector. Include member functions to perform following tasks -To create a vector -To modify the value of given element -To multiply by scalar value. -To display vector.
2	Write a C++ program to design a simple calculator

Course Title: **Design and Analysis of Algorithms**										
Semester	IV	Teaching Scheme				Evaluation Scheme				
						Theory			Practical	
Term	EVEN	Th	Tu	Pr	Credits	TAE	CAE	ESE	INT	EXT
Course Category	C	3 Hrs.	-	2 Hrs.	4	10	15	50	25	25
Course Code	23BTCSL206									
Teaching Mode	Offline	5 Hrs.			Total	75			50	
Duration of ESE	2.5 Hrs.					100				

Course Objectives	This course introduces students the general idea of analysis and design of algorithms while making them aware of basic methods of algorithm analysis and design.
	It is also aimed at developing skills to solve real life applications which involve algorithm development
	The course also provides career opportunities in analysis, design and optimization technique in algorithms
Course Outcomes	CO1: Recall basic concepts of algorithm in analysis and Design of algorithms.
	CO2: Examine Recurrence relations, solutions of recurrence of searching sorting methods
	CO3: Apply Greedy methods used for analysis and Design of Algorithm
	CO4: Analyze Dynamic Programming concepts in designing algorithm
	CO5: Evaluate advanced techniques and tools available for algorithm analysis and development

Mapping of Course Outcomes with Program Outcomes and Program Specific Outcomes:

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO 2
CO1	--	3	--	--	--	--	--	--	--	--	--	--	2	--
CO2	--	--	3	--	--	--	--	--	--	--	--	--	1	--
CO3	--	--	--	3	--	--	--	--	--	--	--	--	2	--
CO4	--	--	3	--	--	--	--	--	--	--	--	--	2	--
CO5	--	--	3	--	--	--	--	--	--	--	--	--	2	--

Course Contents:

Unit	Contents	Hours
I	**Unit – I: Mathematical foundations & Asymptotic notations** summation of arithmetic and geometric series, n, n2,, Complexity calculation of various standard functions, Principles of designing algorithms, ,Asymptotic notations of analysis of algorithms, analyzing control structures, worst case and average case analysis, Iterative Algorithm analysis.	9
II	**Unit – II: Divide and Conquer Method** Recurrence relations, solutions of recurrence relations using technique of characteristic equation and generating functions Divide and conquer basic strategy, matrix operation, Strassens matrix multiplication, binary search, quick sort, merge sort.	9
III	**Unit – III: Greedy Method** Greedy method – basic strategy, application to job sequencing with deadlines problem, minimum cost spanning trees, single source shortest path etc.	9
IV	**Unit IV: Dynamic Programming** Dynamic Programming basic strategy, multistage graphs, all pairs shortest path, single source shortest paths, optimal binary search trees, traveling salesman, Knapsack, sum of subset problems	9
V	**Unit V: Traversal and Search Techniques** Basic Traversal and Search Techniques, breadth first search and depth first search, Backtracking basic strategy, 8-Queen's problem, graph coloring, Hamiltonian cycles , NP-HARD AND NP-COMPLETE PROBLEMS: Basic concepts, non-deterministic algorithms, NP-hard and NP-complete classes.	9

Text Books	1.	Thomas H. Cormenet. al. "Introduction to Algorithms", Prentice Hall of India.
	2.	Design & Analysis of Computer Algorithms by Aho,. Horowitz, Sahani, Rajsekharam, Pearson education
Reference Books	1.	"Computer Algorithms", Galgotia Publications Pvt. Ltd. Brassard, Bratley, "Fundamentals of Algorithms", Prentice Hall
	2.	Computer Algorithms: Introduction to Design and analysis, 3rd Edition, By Sara Baase & A. V. Gelder Pearson Education.
On-line TL Material	1.	NPTEL course on Design and Analysis of Algorithms: https://www.class-central.com/course/nptel-design-and-analysis-of-algorithms-3984

Sr. No.	Title of Practical	Hrs. Allotted	Relevance to CO
1.	To Implement and analyze time complexity of Searching Algorithm.	2	CO1
2.	To Implement and analyze time complexity of Sorting Algorithm.	2	CO1
3.	To Implement and analyze time complexity of algorithm using Divide and Conquer Technique.	2	CO2
4.	To Implement and analyze time complexity of Greedy Algorithm.	2	CO2
5.	To Implement and analyze time complexity of Algorithm of Dynamic Programming Technique.	2	CO3
6.	To implement and analyze time complexity of Minimum Cost spanning Tree Algorithm.	2	CO3
7.	To implement and analyze time complexity of Single source shortest path Algorithm.	2	CO4
8.	To implement and analyze time complexity of All pair shortest path Algorithm.	2	CO4
9.	To implement and analyze time complexity of Graph Traversing Algorithm.	2	CO5
10.	Open-Source Practical	2	CO5

Course Title: **SOFTWARE ENGINEERING AND PROJECT MANAGEMENT MD-II**										
Semester	IV	Teaching Scheme				Evaluation Scheme				
						Theory			Practical	
Term		Th	Tu	Pr	Credits	TAE	CAE	ESE	INT	EXT
Course Category	C	2Hrs	--	2Hrs	3	10	15	50	25	--
Course Code	23BTITL204									
Teaching Mode	Offline	4 Hrs			Total	75			25	
Duration of ESE	2.5 Hrs					100				

Course Objectives	The student should be made to: 1. Understand the phases in a software project
	2. Understand fundamental concepts of requirements engineering and Analysis Modeling.
	3. Understand the major considerations for enterprise integration and deployment.
	4.Learn various testing and maintenance measures
	Up on successful completion of this course, student will be able to:

Course Outcomes	**CO1:** Identify the key activities in managing a software project
	CO2: Compare different process models and design techniques.
	CO3: Understand the concepts of requirements engineering and Analysis.
	CO4: Apply quality management and project management procedure for software design and deployment.
	CO5: Remember various software testing and planning approaches.

Mapping of Course Outcomes with Program Outcomes and Program Specific Outcomes:

Course Outcomes	Program Outcomes and Program Specific Outcomes														
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	3	2	3	-	-	-	-	-	-	-	3	-	-	-	-
CO2	3	2	3	-	-	-	-	-	-	-	3	-	-	-	-
CO3	2	3	3	2	1	-	-	-	-	-	3	-	3	2	-
CO4	2	3	3	-	-	-	-	-	-	-	3	-	3	3	-
CO5	3	-	3	1	-	-	-	-	-	-	3	-	-	3	-

Course Contents:

Unit	Contents	Hours
I	**SOFTWARE PROCESS** Introduction to Software Engineering, Software Process, Perspective and Specialized Process Models, Agile models, Software Project Management, Project Scheduling – Scheduling, Earned Value Analysis.	7
II	**REQUIREMENTS ANALYSIS AND SPECIFICATION** Software Requirements: Functional and Non-Functional, User requirements, System requirements, Software Requirements Document – Requirement Engineering Process: Feasibility Studies, Requirements elicitation and analysis, requirements validation, requirements management-Classical analysis: Structured system Analysis, Petri Nets- Data Dictionary	7
III	**SOFTWARE DESIGN** Design process – Design Concepts-Design Model– Design Heuristic – Architectural Design – Architectural styles, Architectural Design, Architectural Mapping using Data Flow- User Interface Design: Interface analysis, Interface Design –Component level Design: Designing Class based components, traditional Components.	7
IV	**PROJECT MANAGEMENT** Estimation – FP Based, LOC Based, Make/Buy Decision, COCOMO II – Planning – Project Plan, Planning Process, RFP Risk Management – Identification, Projection, RMMM – Scheduling and Tracking –	7
V	**TESTING** Software testing fundamentals-Internal and external views of Testing-white box testing – basis path testing-control structure testing-black box testing Debugging	7

Text Books	1.	Roger S. Pressman, "Software Engineering – A Practitioner"s Approach", Seventh Edition, McGraw-Hill International Edition, 2010.
	2.	Ian Sommerville, "Software Engineering", 9th Edition, Pearson Education Asia, 2011.
E--Books	1.	Foundations of Software Engineering by Kevin Amaratunga
	2.	The New Software Engineering by Sue Conger
Reference Books	1.	Rajib Mall, "Fundamentals of Software Engineering", Third Edition, PHI Learning Private Limited, 2009.
	2.	PankajJalote, "Software Engineering, A Precise Approach", Wiley India, 2010.
on line TL Material	1.	Certification Courses available: https://nptel.ac.in/downloads/106105087/
	2.	Free Courses available: https://swayam.gov.in/course/236-software-enginnering

Practical List

Sr. No.	Practical List	Mapping with CO	Mapping with PO & PSO
1	To understand the importance of SDLC approach & various processes models.	CO1	PO1, PSO1
2	To identify the requirements from the problem statement.	CO1	PO1, PO2, PSO2
3	To identify the suitable selection criteria of software development model for the given problem.	CO2	PO3, PSO2
4	To identify the various requirement development activities viz. elicitation, analysis, specification and verification for the given scenarios.	CO3	PO3, PO4, PSO2, PSO3
5	To identify the elements in Software Requirements Specification document for a given scenario.	CO3	PO1, PO2, PSO2
6	To understand E-R Modeling from the problem statement.	CO4	PO4, PO5, PSO2, PSO3
7	To study and illustrate the various software testing techniques.	CO4	PO3, PSO1, PSO2
8	To study and illustrate various UML diagrams and the use of class diagrams.	CO4	PO3, PSO1, PSO2
Open Ended Practical			
9	To design test cases for given problem statement.	CO5	PO2, PO3, PSO1, PSO2
Content Beyond Practical			
10	To study and illustrate the various automation tool like selenium.	CO5	PO2, PO3, PSO1, PSO2

Course Title: **BASICS OF GAME PROGRAMMING Open Elective-II**										
Semester	IV	Teaching Scheme				Evaluation Scheme				
						Theory			Practical	
Term		Th	Tu	P	Credits	TAE	CAE	ESE	INT	EXT
Course Category	OE-II	2 Hrs.	--	--	2	10	15	50	--	--
Course Code	23BTITL206									
Teaching Mode	Offline	2 Hr			Total	75			--	
Duration of ESE	2Hrs					75				

Course Objectives	The objectives of the course is to let students understand the concept of Game design and development, learning different processes, mechanics and the issues in Game Design.
	Another objective is to expose students to the Core architectures of Game Programming, Knowing about Game programming platforms, frame works and engines. Learn to develop games.
Course Outcomes	CO1: Discuss the concepts of Game design and development.
	CO2: Design the processes, and use mechanics for game development.
	CO3: Explain the Core architectures of Game Programming.
	CO4: Use Game programming platforms, frame works and engines.
	CO5: Create interactive Games.

Mapping of Course Outcomes with Program Outcomes and Program Specific Outcomes:

Course Outcomes	Program Outcomes and Program Specific Outcomes													
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	2	2	--	--	1	--	--	--	2	--	1	--	--
CO2	2	2	3	--	3	--	--	--	--	--	--	1	--	3
CO3	--	--	--	--	--	--	--	--	--	2	--	2	--	--
CO4	--	--	3	--	3	--	--	--	1	--	2	--	2	2
CO5	--	--	2	--	--	1	--	--	2	2	2	2	1	3

Course Contents:

Unit	Contents	Hours
I	**3D Graphics for Game Programming:** 3D Transformations, Quaternions, 3D Modeling and Rendering, Ray Tracing, Shader Models, Lighting, Color, Texturing, Camera and Projections, Culling and Clipping, Character Animation, Physics-based Simulation, Scene Graphs.	6
II	**Game Engine Design:** Game engine architecture, Engine support systems, Resources and File systems, Game loop and real-time simulation, Human Interface devices, Collision and rigid body dynamics, Game profiling.	8

III	**Game Programming**: Application layer, Game logic, Game views, managing memory, controlling the main loop, loading and caching game data, User Interface management, Game event management.	8
IV	**Gaming Platforms and Frameworks:** 2D and 3D Game development using Flash, DirectX, Java, Python, Game engines - Unity. DX Studio.	6
V	**Game Development**: Developing 2D and 3D interactive games using DirectX or Python – Isometric and Tile Based Games, Puzzle games, Single Player games, and Multi-Player games.	6

Text Books	1.	Mike Mc Shaffrfy and David Graham, "Game Coding Complete", Fourth Edition, Cengage Learning PTR.
	2.	Jason Gregory, "Game Engine Architecture", CRC Press / A K Peters
	3.	David H. Eberly, "3D Game Engine Design: A Practical Approach to Real-Time Computer Graphics", Morgan Kaufmann

Reference Books	1.	Ernest Adams and Andrew Rollings, "Fundamentals of Game Design", Prentice Hall / New Riders
	2.	Eric Lengyel, "Mathematics for 3D Game Programming and Computer Graphics", Course Technology PTR.
	3.	Jesse Schell, The Art of Game Design: A book of lenses, 1st Edition, CRC Press

Course Title: **Web Development :Vocational &Skill Enhancement elective_III**										
Semester	IV	Teaching Scheme				Evaluation Scheme				
						Theory			Practical	
Term		Th	Tu	Pr	Credits	TAE	CAE	ESE	INT	EXT
Course Category	C	1Hrs.	--	2Hrs.	2	10	15		25	-
Course Code	23BTCSL207									
Teaching Mode	Offline	3 Hrs			Total	25			25	
Duration of ESE	NA					50				

Course Objectives	1. To understand the basic principles of web designing
	2. To learn web page creation using HTML
	3. To learn dynamic website design using CSS & JAVA Script
	4. To learn content management system
	5. To learn web hosting
Course Outcomes	Upon successful completion of the course, students will be able to
	CO1: Explain the basic principles of web designing
	CO2: Implement all basic tags in HTML
	CO3: Design web page using HTML, CSS & JAVA Script
	CO4: Design & Understand content management system
	CO5: Publish & Host the website

Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1:	-	3	3	-	-	-	-	-	-	-	-	-	2	-	-
CO2:	-	1	3	-	-	-	-	-	-	-	-	-	3	-	-
CO3:	-	1	2	2	-	-	-	-	-	-	-	-	3	-	-
CO4:	-	1	2	2	-	-	-	-	-	-	-	-	2		
CO5:	-	1	1	2	-	-	-	-	-	-	-	-	2	-	-

Unit	Contents	Hours
I	**Web Design Principles:** Basic principles involved in developing a web site, Planning process, Designing navigation bar, Page design, Home Page Layout Design Concept., Brief History of Internet, What is World Wide Web, Why create a web site, Web Standards	8
II	**Introduction to HTML:** What is HTML, HTML Documents, Basic structure of an HTML document, Creating an HTML document, Mark up Tags, Heading-Paragraphs, Line Breaks, HTML Tags. Elements of HTML: Introduction to elements of HTML, Working with Text, Working with Lists, Tables and Frames, Working with Hyperlinks, Images and Multimedia, Working with Forms and controls	8

III	**Introduction to Cascading Style Sheets & JAVA Script:** Concept of CSS, Creating Style Sheet, CSS Properties, CSS Styling(Background, Text Format, Controlling Fonts), Working with block elements and objects, Working with Lists and Tables, CSS Id and Class, Box Model, Advanced CSS, JAVA Script Introduction, Application, Advantages, Popup Boxes, Programming details, Class & object.	8
IV	**Content Management System:** The differences between content, content management, and a content management system. Types of content management systems. Why we use CMS, what a CMS does, and what CMS doesn't do	8
V	**Introduction to Web Publishing or Hosting:** Creating the Web Site, Saving the site, Working on the web site, Creating web site structure, Creating Titles for web pages, Themes-Publishing web sites. Tomcat Installation and Initial Configuration, Tomcat Architecture – Understanding the directory structure.Understanding server.xml (deployment descriptor), Deploying Web Applications.	8

Text Books	1.	HTML 5 in simple steps by Kogent Learning Solutions Inc. Dreamtech Press
	2.	HTML, XHTML, and CSS Bible, 5ed Steven M. Schafer , Wiley India
E--Books	1.	HTML 5 in simple steps by Kogent Learning Solutions Inc. Dreamtech Press
	2.	HTML, XHTML, and CSS Bible, 5ed Steven M. Schafer , Wiley India
Reference Books	1.	Web Technologies: HTML, Javascript, Kogent Learning, Wiley India
Online TL Material	1.	NPTEL course: https://nptel.ac.in/courses/106/106/106106156/
	2.	Udemy course: Web Design for Beginners: Real World Coding in HTML & CSS

Experiment Number	Title of the Experiments	COs	Duration
Experiment No. 1	Implementing Basic tags in HTML	CO1	2 Hrs
Experiment No. 2	Design a web page using table tag exploring all attributes	CO1	2 Hrs
Experiment No. 3	Design a form in html considering all input types	CO2	2 Hrs
Experiment No. 4	Design a web page using inline & embedded CSS	CO2	2 Hrs

Experiment No. 5	Design webpage using external CSS	CO3	2 Hrs
Experiment No. 6	Design & implement all types of popup boxes using JAVA Script	CO3	2 Hrs
Experiment No. 7	Design a calculator in html using JAVA Script taking inputs from user	CO4	2 Hrs
Experiment No. 8	Design a dynamic website using combinations of HTML, CSS & JAVA Script	CO4	2 Hrs
Open Ended Practical			
Experiment No. 9	Web hosting	CO5	2 Hrs
Content Beyond Practical			
Experiment No. 10	Explore the different open source web designing tools	CO5	2 Hrs