BMS INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(Autonomous Institute affiliated to VTU, Belagavi, Approved by AICTE New Delhi)

Yelahanka, Bengaluru 560119

Bachelor of Engineering

Department of Computer Science and Engineering

VII and VIII Semester Scheme & Syllabus
2022 Scheme
Effective from the AY 2025-26

Vision and Mission of the Department

Vision

To be a centre of excellence in Computer Science and Engineering education and research, nurturing technically competent, ethically responsible, and socially conscious professionals to meet global challenges and drive sustainable innovation.

Mission

M1. To impart quality education in Computer Science and Engineering by integrating fundamental knowledge with emerging technologies and industry practices.

M2. To foster innovation, problem-solving, and research aptitude through a curriculum enriched with project-based learning, professional activities, and collaborative initiatives.

M3. To develop graduates with strong ethical values, leadership qualities, and a commitment to lifelong learning through co-curricular and extra-curricular activities.

Program Educational Objectives (PEOs)

PEOs	
PEO1	Professional Excellence: Pursue successful careers in industry, academia, and entrepreneurship by applying the foundational knowledge of Computer Science and Engineering with professional competence.
PEO2	Higher Education and Lifelong Learning: Engage in higher studies, research, or professional development programs, demonstrating a commitment to lifelong learning in a rapidly evolving technological landscape.
PEO3	Ethics and Social Responsibility: Exhibit ethical behaviour, effective communication, teamwork, and leadership qualities, with a strong sense of responsibility toward society and the environment.

Program Specific Outcomes (PSOs)

PSOs	
PSO-1	Apply theoretical foundations, Algorithmic principles and software engineering practices to develop efficient and scalable IT solutions.
PSO-2	Design effective systems by leveraging principles of computing and communication technologies.

ಬಿ.ಎಂ.ಎಸ್. ತಾಂತ್ರಿಕ ಮತ್ತು ವ್ಯವಸ್ಥಾಪನಾ ಮಹಾವಿದ್ಯಾಲಯ
BMS Institute of Technology and Management
(An Autonomous Institution, Affiliated to VTU Belagavi)
Avalahalli, Doddaballapur Main Road, Bengaluru, Karnataka – 560064

REVISED

Date: 18-12-2024

CONTINUOUS INTERNAL EVALUATION (CIE)
AND
SEMESTER END EXAMINATION (SEE) PATTERN

(Applicable to UG students admitted from the 2022 batch, effective from the Academic year 2024-25 onwards)

The UG students admitted from the 2022 batch onwards are hereby informed to note the following regarding Continuous Internal Evaluation and Semester End Examination pattern:

- The Weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Examination (SEE) is 50%.
- The Minimum passing mark for the CIE is 40% of the Maximum marks (i.e. 20 marks out of 50) and for the SEE minimum passing mark is 35% of the Maximum marks (i.e. 18 out of 50 marks).
- A student will be declared to have passed the course if they secure a minimum of 40% (i.e. 40 marks out of 100) in the combined total of the CIE and SEE.

The following tables summarize the CIE and SEE Patterns for the courses of various credits:

IPCC COURSES: 4 CREDITS OR 3 CREDITS						
Evaluation Type		**Internal Assessments (IAs)**	**Test/ Exam Marks Conducted for**	**Marks to be scaled down to**	**Min. Marks to be Scored**	**Evaluation Details**
Theory Component	CIE – IA Tests	CIE – Test 1 (1.5 hr)	40	20	-	The sum of the two internal assessment tests will be **80 Marks** and the same will be scaled down to **20 Marks**.
		CIE – Test 2 (1.5 hr)	40			

	CIE – CCA (Comprehensive Continuous Assessment)	CCA	10	05	-	Any one assessment method can be used from the list appended below.
	Total CIE Theory			**25**	**10**	
Practical Component	CIE - Practical		30	15	-	Each laboratory experiment is to be evaluated for **30 Marks** using appropriate rubrics.
	CIE Practical Test		20	10	-	One test after all experiments to be conducted for **20 Marks**
	Total CIE Practical			**25**	**10**	
Total CIE Theory + Practical				**50**	**20**	
SEE			100	50	18	SEE exam is a theory exam, conducted for **100 Marks**, scored marks are scaled down to **50 Marks.**
CIE + SEE				**100**	**40**	
The laboratory component of the IPCC shall be for CIE only.						

Professional Core Courses (PCC) / Engineering Science Courses (ESC): 03 and 02 Credit						
Evaluation Type		**Internal Assessments (IAs)**	**Test/ Exam Marks Conducted for**	**Marks to be scaled down to**	**Min. Marks to be Scored**	**Evaluation Details**
Theory Component	CIE – IA Tests	CIE – Test 1 (1.5 hr)	40	30	-	The sum of the two internal assessment tests will be **80 Marks** and the same will be scaled down to **30 Marks**.
		CIE – Test 2 (1.5 hr)	40			
	CIE - CCAs	CCA	20	20	-	Any Two assessment methods can be used from the list. If it is project-based, one CCA shall be given.
	Total CIE Theory			**50**	**20**	
SEE			100	50	18	SEE is a theory exam, conducted for **100 Marks**, scored marks are scaled down to **50 Marks.**
CIE + SEE				**100**	**40**	

`

NON-IPCC COURSES: 01 Credit Course - MCQ						
Evaluation Type		**Internal Assessments (IAs)**	**Test/ Exam Marks Conducted for**	**Marks to be scaled down to**	**Min. Marks to be Scored**	**Evaluation Details**
Continuous Internal Evaluation Component	CIE – IA Tests (MCQs)	CIE – Test 1 (1 hr)	40	40	-	The question paper pattern for this course shall be an **MCQ** of **1 or 2 Marks (s).** The questions with 2 Marks can be framed based on a higher Bloom's level. The sum of the two internal assessment tests will be **80 Marks**, and the same will be scaled down to **40 Marks**.
		CIE – Test 2 (1 hr)	40			
	CIE - CCAs	CCA	10	10	-	Any One Assessment method can be used from the list provided below.
	Total CIE			**50**	**20**	
SEE (MCQ Type)				50	18	The question paper pattern for this course shall be an **MCQ** of **1 or 2 Marks (s).** The questions with 2 Marks can be framed based on higher Bloom's level. MCQ-type question papers of 50 questions with each question of a **01 Mark**, examination duration is 01 hour.
CIE + SEE				**100**	**40**	

Professional Core Course Laboratory (PCCL) /Ability Enhancement Course Laboratory (AEC) - 01 Credit					
Evaluation Type	**Internal Assessments (IAs)**	**Test/ Exam Marks Conducted for**	**Marks to be scaled down to**	**Min. Marks to be Scored**	**Evaluation Details**
Continuous Internal Evaluation	CIE - Practical	30	30		Each laboratory experiment is to be evaluated for **30 Marks** using appropriate rubrics.
	CIE - Practical Test	50	20		One test after all experiments is to be conducted for **50 Marks** and to be scaled down to **20 Marks**.
	Total CIE	-	**50**	**20**	
Semester End Examination		100	50	18	SEE to be conducted for **100 Marks**.
CIE+SEE		**100**		**40**	

Learning Activities for CCAs:
A faculty member may choose the following CCAs based on the needs of the course:

1. Course project
2. Literature review
3. MOOC
4. Case studies
5. Tool exploration
6. GATE-based aptitude test
7. Open book tests
8. Industry integrated learning
9. Analysis of Industry / Technical / Business reports
10. Programming assignments with higher Bloom level
11. Group discussions
12. Industrial / Social / Rural projects

CoE 18/12/2024

Dean AA 18.12.24

Principal 18/12/24

Copy To:
1. The Vice-Principal, Deans, HoDs, and Associate HoDs
2. All faculty members and students of 2022, 2023, and 2024 batch.
3. Examination Section

`

Scheme of VII Semester

BMS INSTITUTE OF TECHNOLOGY &MANAGEMENT

(Autonomous Institution Affiliated to VTU, Belagavi)

B. E. in Computer Science and Engineering

Scheme of Teaching and Examinations – 2022 Scheme

Outcome-Based Education (OBE) and Choice Based Credit System (CBCS)

(Effective from the academic year 2025-26 onwards)

Common to CSE/ISE

UG PROGRAM: CSE/ISE **Semester : VII**

Sl. No.	Course Category	Course Code	Course Title	Teaching Department (TD) and Question Paper Setting Board (PSB)	Teaching Hours /Week					Examination				Credits
					L	T	P	S	Total	CIE Marks	SEE Marks	Total Marks	SEE Duration (H)	
1	IPCC	BCS701	Information and Network Security	TD: CSE PSB: CSE/ISE	3	0	1		4	50	50	100	3	4
2	PCC	BCS702	Parallel Computing		3	0	0		3	50	50	100	3	3
3	PEC	BCS703X	Professional Elective Course III		3	0	0		3	50	50	100	3	3
4	OEC	BCS704X	Open Elective Course II		3	0	0		3	50	50	100	3	3
5	PW	BCSP705	Major Project Phase II		0	0	7		7	100	100	200	3	7
6	PCCL	BCSL706	Parallel Computing Laboratory		0	0	1		1	50	50	100	3	1
TOTAL										**350**	**350**	**700**	**-**	**21**

IPCC: Integrated Professional Core Course, **PCC**: Professional Core Courses, **PCCL**: Professional Core Course laboratory, **UHV**: Universal Human Value Course, **NCMC**: Non Credit Mandatory Course, **ESC**: Engineering Science Course **AEC**: Ability Enhancement Course, **SEC**: Skill Enhancement Course, **L:** Lecture, **T**: Tutorial, **P**: Practical, **S:** Self Study, **CIE**: Continuous Internal Evaluation, **SEE**: Semester End Evaluation.

Integrated Professional Core Course (IPCC): Refers to Professional Core Course Theory Integrated with practical of the same course. Credit for IPCC can be 04 and its Teaching–Learning hours (L: T: P: S) can be considered as (3: 0: 2: 0) or (3: 2: 0:0). The theory part of the IPCC shall be evaluated both by CIE and SEE. The practical part shall be evaluated by only CIE (no SEE). However, questions from the practical part of IPCC shall be included in the SEE question paper.

Professional Elective Courses (PEC): A professional elective (PEC) course is intended to enhance the depth and breadth of educational experience in the Engineering and Technology curriculum. Multidisciplinary courses that are added supplement the latest trend and advanced technology in the selected stream of engineering. Each group will provide an option to select one course. The minimum number of students' strengths for offering a professional elective is 10. However, this condition shall not be applicable to cases where the admission to the program is less than 10.

Professional Elective Course III		**Open Elective Course II**	
Course Code	**Course Name**	**Course Code**	**Course Name**
BCS703A	Natural Language Processing	BCS704A	Introduction to Algorithms
BCS703B	Building Enterprise Applications	BCS704B	Introduction to Augmented Reality and Virtual Reality
BCS703C	Deep Learning	BCS704C	Introduction to Cyber Security
BCS703D	Quantum Computing		

National Service Scheme /Physical Education/Yoga/NCC/Music: All students have to register for any one of the courses namely National Service Scheme (NSS), Physical Education (PE) (Sports and Athletics), and Yoga (YOG) with the concerned coordinator of the course during the first week of III semesters. Activities shall be carried out between III semester to the VI semester (for 4 semesters). Successful completion of the registered course and requisite CIE score is mandatory for the award of the degree. The events shall be appropriately scheduled by the colleges and the same shall be reflected in the calendar prepared for the NSS, PE, and Yoga activities. These courses shall not be considered for vertical progression as well as for the calculation of SGPA and CGPA, but completion of the course is mandatory for the award of degree.

Open Elective Courses (OEC): Students belonging to a particular stream of Engineering and Technology are not entitled to the open electives offered by their parent Department. However, they can opt for an elective offered by other Departments, provided they satisfy the prerequisite condition if any. Registration to open electives shall be documented under the guidance of the Program Coordinator/ Advisor/Mentor.

Selection of an open elective shall not be allowed if,

- The candidate has studied the same course during the previous semesters of the program.
- The syllabus content of open electives is similar to that of the Departmental core courses or professional electives.
- A similar course, under any category, is prescribed in the higher semesters of the program.
- The minimum students' strength for offering open electives is 10. However, this condition shall not be applicable to cases where the admission to the program is less than 10.

Major Project Phase II: The objective of the Project work is (i) To encourage independent learning and the innovative attitude of the students. (ii) To develop interactive attitude, communication skills, organization, time management, and presentation skills. (iii) To impart flexibility and adaptability. (iv) To inspire teamwork. (v) To expand intellectual capacity, credibility, judgment and intuition. (vi) To adhere to punctuality, setting and meeting deadlines. (vii) To instill responsibilities to oneself and others. (viii)To train students to present the topic of project work in a seminar without any fear, face the audience confidently, enhance communication skills, involve in group discussion to present and exchange ideas.

`BMS INSTITUTE OF TECHNOLOGY &MANAGEMENT

(Autonomous Institution Affiliated to VTU, Belagavi)

B. E. in Computer Science and Engineering

Scheme of Teaching and Examinations – 2022 Scheme

Outcome-Based Education (OBE) and Choice Based Credit System (CBCS)
(Effective from the academic year 2025-26 onwards)

Common to CSE/ISE

UG PROGRAM: CSE/ISE **Semester : VIII**

Sl. No.	Course Category	Course Code	Course Title	Teaching Department (TD) and Question Paper Setting Board (PSB)	Teaching Hours /Week					Examination				Credits
					L	T	P	S	Total	CIE Marks	SEE Marks	Total Marks	SEE Duration (H)	
1	PEC	BCS801X	Professional Elective Course IV (Online courses)	TD: ISE PSB: CSE/ISE	3	0	0		3	50	50	100	3	3
2	OEC	BCS802X	Open Elective Course III (Online courses)		3	0	0		3	50	50	100	3	3
3	INT	BCSI803	Industry Internship / Research Internship	-	0	0	10		10	100	100	200	3	10
TOTAL										**200**	**200**	**400**		**16**

Professional Elective Course IV (Online courses)

Course Code	NPTEL Course ID	Course Name
noc25-cs65	https://onlinecourses.nptel.ac.in/noc25_cs65/preview	Social Networks
noc25-cs37	https://onlinecourses.nptel.ac.in/noc25_cs37/preview	GPU Architectures and Programming
noc25-cs09	https://onlinecourses.nptel.ac.in/noc25_cs43/preview	Business Intelligence & Analytics
noc25-cs62	https://onlinecourses.nptel.ac.in/noc25_cs62/preview	Reinforcement Learning
noc25-cs02	https://onlinecourses.nptel.ac.in/noc25_cs02/preview	Advanced Computer Networks

Open Elective Course III (Online courses)

Course Code	NPTEL Course ID	Course Name
noc25-cs61	https://onlinecourses.nptel.ac.in/noc25_cs61/preview	Quantum Algorithms and Cryptography
noc25-cs32	https://onlinecourses.nptel.ac.in/noc25_cs32/preview	Foundations of Cyber Physical Systems
noc25-cs38	https://onlinecourses.nptel.ac.in/noc25_cs38/preview	Human Computer Interaction
noc25-cs35	https://onlinecourses.nptel.ac.in/noc25_cs35/preview	Games and Information
noc25-cs45	https://onlinecourses.nptel.ac.in/noc25_cs45/preview	Introduction to Large Language Models (LLMs)

Professional Elective / Open Elective Online Courses: These are ONLINE courses suggested by the respective Board of Studies. Details of these courses shall be made available for students during the V semester only. Students are required to choose only the courses which are suggested by the respective BoS. Duration of the online course should be of a minimum of 12 weeks. Students can able to complete the specified online courses with a qualifying certificate issued by the competent authority. The online courses can be registered and completed from the vacation of V semester onwards and the credits will be considered during the VIII semester only. The courses are to be chosen from the SWAYAM – NPTEL platform only. The credits earned for these courses will not be considered for claiming the credits under the Honors Degree programme.
Industry Internship / Research Internship: The mandatory Industry / Research internship is for 14 to 20 weeks. The internship shall be considered as a head of passing and shall be considered for the award of a degree. Those who do not take up/complete the internship shall be declared to fail and shall have to complete it during the subsequent examination after satisfying the internship requirements. Research internship: A research internship is intended to offer the flavor of current research going on in the research field. It helps students get familiarized with the field and imparts the skill required for carrying out research. Industry internship: Is an extended period of work experience undertaken by students to supplement their degree for professional development. It also helps them learn to overcome unexpected obstacles and successfully navigate organizations, perspectives, and cultures. Dealing with contingencies helps students recognize, appreciate, and adapt to organizational realities by tempering their knowledge with practical constraints.

B.E. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS) applicable for 2022 Scheme

SEMESTER – VII

Information and Network Security (3:0:1:0) 4

(Effective from the academic year 2025-26)

Course Code	BCS701	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	3:0:1:0	SEE Marks	50
Total Number of Contact Hours	40(Theory)+26(Practical)	Exam Hours	03

Course Objectives:

This course will enable students to:

1. Understand the fundamentals of Cryptography and Hash functions.
2. Demonstrate concepts of Random number generation and key managements.
3. Design Cryptographic applications using any key generation algorithms.

Preamble: This course introduces the students to understand the concept of cryptography and cipher text and how to maintain data integrity using Hash functions. Exploring fundamentals of entity authentication passwords, dynamic passwords and fundamentals of Key management. Establishing Cryptographic applications in wireless networks.

Module – 1

Introduction: How to Speak Crypto. Classic Crypto. Simple Substitution Cipher. Cryptanalysis of a Simple Substitution. Definition of Secure. Double Transposition Cipher. One-time Pad. Project VENONA. Codebook Cipher. Ciphers of the Election of 1876. Modern Crypto History. Taxonomy of Cryptography. Taxonomy of Cryptanalysis.

Textbook1: Chapter 2 **(8 Hours)**

Module – 2

What is a Hash Function: The Birthday Problem. Non-cryptographic Hashes. Tiger Hash. HMAC. Uses of Hash Functions. Online Bids. Spam Reduction. Other Crypto-Related Topics. Secret Sharing. Key Escrow. Random Numbers. Texas Hold 'em Poker. Generating Random Bits. Information Hiding.

Textbook1: Chapter 5 (**8 Hours)**

Module – 3

Random number generation: Providing freshness Fundamentals of entity authentication Passwords Dynamic password schemes Zero-knowledge mechanisms Further reading Cryptographic Protocols Protocol Basics from objectives to a protocol Analysing a simple protocol Authentication and key establishment protocols.

Textbook2: Chapter 8,9 (**8 Hours)**

Module – 4

Key management fundamentals: Key lengths and lifetimes Key Generation Key Establishment Key Storage Key Usage Governing key management Public-Key Management Certification of public keys The certificate lifecycle Public-key management models Alternative approaches

Textbook2: Chapter 10, 11 (**8 Hours)**

Module – 5

Cryptographic Applications: Cryptography on the Internet Cryptography for wireless local area networks Cryptography for mobile telecommunications Cryptography for secure payment card transactions Cryptography for video broadcasting Cryptography for identity cards Cryptography for home users.

Textbook2: Chapter 12 **(8 Hours)**

Course Outcomes:

The students will be able to:

CO1: Apply the cryptographic algorithm to solve the give problem.

CO2: Apply the concepts of Hash functions and random number generation in network security.

CO3: Use the fundamentals of Key management in maintaining Security.

CO4: Examine Network cryptographic concepts for the given scenario.

Textbooks:

1. Information Security: Principles and Practice, 2nd Edition by Mark Stamp Wiley,2011
2. Everyday Cryptography: Fundamental Principles and Applications Keith M. Martin Oxford Scholarship Online: December 2013.

References:

1. Applied Cryptography Protocols, Algorithms, and Source Code in C by Bruce Schneier, 2015.
2. Cryptography and Network Security: Principles and Practice by William Stallings, 8th edition, Pearson Publication, 2023.

Continuous Course Assessment suggested:

1. Demonstration of simple projects on Cryptography applications

Lab Programs:

1. Write a C Program to perform Caesar cipher encryption and decryption algorithm by taking security key.
2. Write a C Program to implement Polyalphabetic (Vigenere) cipher encryption-decryption which uses two or more cipher alphabets to encrypt the data.
3. Write a program to find out the original text using frequency analysis. It is known that the original text is an English article.
4. Write a C Program to implement Transposition cipher which is a simple data encryption scheme in which plaintext characters are shifted in some regular pattern to form cipher text.
5. Write a program to demonstrate the Birthday Attack on a hash function.
   Assume a hash function that produces values between 0 and 999.
   Randomly generate hash values until a duplicate (collision) is found.
   Display how many random values were needed before the first collision occurred.

B.E. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS)

SEMESTER - VII

Parallel Computing (3:0:0:0)3

Common to all Branches

(Effective from the academic year 2025-26 for 2022 Scheme)

Course Code	BCS702	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	3:0:0:0	SEE Marks	50
Total Number of Lecture Hours	40(Theory)	Exam Hours	03

CREDITS : 03

Prerequisites: C programming language, Data structures and algorithms.

Course objectives:

This course will enable students to

1. To understand the concepts of Parallel Computers, Data and Temporal Parallelism.
2. To learn Structures of Parallel Computers.
3. To understand the concepts of Operating Systems for Parallel Computers.
4. To acquire knowledge on CUDA.
5. To learn Parallel Programming with CUDA C.

Module – 1

Introduction: Why do we need High Speed Computing, how do we Increase the Speed of Computers, History of Parallel Computers. **Solving problems in parallel:** Utilizing Temporal Parallelism, Utilizing Data Parallelism, Comparison of Temporal and Data Parallel Processing, Data Parallel Processing with Specialized Processors. **(8 Hours)**

Module – 2

Structure of parallel computers: A Generalized Structure of a Parallel Computer, Classification of Parallel Computers, Vector Computers, A Typical Vector Super Computer, Array Processors, Shared Memory Parallel Computers, Distributed Shared Memory Parallel Computers, Message Passing Parallel Computers. **(8 Hours)**

Module – 3

Operating systems for parallel computers: Resource Management, Process Management, Process Synchronization, Inter-process Communication, Memory Management, Input/output (Disk Arrays), Basics of Performance Evaluation, Performance Measurement Tools.

(8 Hours)

Module – 4

Computer unified device architecture: The age of parallel processing, The rise of GPU computing, CUDA, Applications of CUDA, Development Environment-CUDA Enabled Graphics Processors, NVIDIA Device driver, CUDA Development Tool kit, Standard C compiler.

(8 Hours)

Module – 5

CUDA C: Introduction to CUDA C: First program, Querying Devices, Using Device Properties, Parallel Programming in CUDA C: CUDA Parallel Programming- Summing Vectors program.

(8 Hours)

Course outcomes:

The students will be able to:

CO1 : Apply data and temporal parallelism concepts to solve computational problems.
CO2: Apply the principles of parallel computer architectures and their structural models.
CO3: Analyze the performance metrics and evaluation methods of parallel computer systems.
CO4: Develop parallel programs using CUDA C.

Text Books

1. Parallel Computers Architecture and Programming, V. Rajaraman, C. Siva Ram Murthy, PHI, 2nd Edition, 2016.
2. CUDA by Example, Jason Sanders, Edward Kandrot, Addison Wesley, 2010.

References

3. Introduction to Parallel Computing, Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar, second edition, 2003 by Addison-Wesley, an imprint of Pearson Education.
4. Parallel Computing Theory and Practice, Michel j. Quinn, 2nd Edition, 2017.

Continuous Course Assessment(CCA) suggested:

1. Solving case study examples with CUDA.
2. Nvidia MOOC's certification.

B.E. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS) applicable for 2022 Scheme

SEMESTER – VII

Natural Language Processing (3:0:0:0) 3

Common to all Branches

(Effective from the academic year 2025-26 for 2022 Scheme)

Course Code	BCS703A	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	3:0:0:0	SEE Marks	50
Total Number of Lecture Hours	40 (Theory)	Exam Hours	03

Course objectives:

This course will enable students to

1. Apply the fundamental concept of NLP, grammar-based language model and statistical-based language model.
2. Model morphological analysis using Finite State Transducers and parsing using context-free grammar and different parsing approaches.
3. Develop the Naïve Bayes classifier and sentiment analysis for Natural language problems and text classifications.
4. Apply the concepts of information retrieval, lexical semantics, lexical dictionaries such as WordNet, lexical computational semantics, distributional word similarity.
5. Identify the Machine Translation applications of NLP using Encode and Decoder.

Module 1

Overview and language modeling: Overview: Origins and challenges of NLP-Language and Grammar-Processing Indian Languages- NLP Applications-Information Retrieval. Language Modeling: Various Grammar- based Language Models-Statistical Language Model.
Textbook 1: Ch. 1: 1.1-1.9, Ch. 2: 2.1-2.3 **(8 Hours)**

Module 2

Word level and syntactic analysis: Word Level Analysis: Regular Expressions-Finite-State Automata-Morphological Parsing-Spelling Error Detection and correction-Words and Word classes-Part-of Speech Tagging. Syntactic Analysis: Context-free Grammar-Constituency- Parsing-Probabilistic Parsing.
Textbook 1: Ch. 3: 3.1-3.7, Ch. 4: 4.1-4.5 **(8 Hours)**

Module 3

Naive Bayes, Text Classification and Sentiment: Naive Bayes Classifiers, Training the Naive Bayes Classifier, Worked Example, Optimizing for Sentiment Analysis, Naive Bayes for Other Text Classification Tasks, Naive Bayes as a Language Model.
Textbook 2: Ch. 4: 4.1-4.6 **(8 Hours)**

Module 4

INFORMATION RETRIEVAL AND LEXICAL RESOURCES: Information Retrieval:
Design features of Information Retrieval Systems-Classical, Non classical, Alternative Models of Information Retrieval valuation Lexical Resources: World Net-Frame Net-Stemmers-POS Tagger- Research Corpora.
Textbook 1: Ch. 9: 9.1-9.7, Ch. 12: 12.1-12.6 **(8 Hours)**

Module 5

Machine Translation: Language Divergences and Typology, Machine Translation using Encoder Decoder, Details of the Encoder-Decoder Model, Translating in Low-Resource Situations, MT Evaluation, Bias and Ethical Issues.
Textbook 2: Ch. 13: 13.1-13.7 **(8 Hours)**

Course outcomes:

The students will be able to:

CO1. Illustrate the foundational concepts of Natural Language Processing including language models, grammar rules, and NLP applications.

CO2. Apply morphological analysis and parsing techniques using tools like Finite State Transducers and Context-Free Grammars.

CO3. Analyze text classification methods and sentiment analysis by implementing the Naïve Bayes classifier on natural language datasets.

CO4. Evaluate the effectiveness of information retrieval systems, lexical resources (e.g., WordNet), and machine translation models like Encoder-Decoder.

Text Books:

1. Tanveer Siddiqui, U.S. Tiwary, "Natural Language Processing and Information Retrieval", Oxford University Press, 2018.
3. Daniel Jurafsky, James H. Martin, "Speech and Language Processing, An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition", Pearson Education, 2023.

Reference Books:

1. Akshay Kulkarni, Adarsha Shivananda, "Natural Language Processing Recipes - Unlocking Text Data with Machine Learning and Deep Learning using Python", Apress, 2019.
2. T V Geetha, "Understanding Natural Language Processing – Machine Learning and Deep Learning Perspectives", Pearson, 2024.
3. Steven Bird, Ewan Klein, and Edward Loper, "Natural Language Processing with Python' first edition, o'reilly Publication, 2009.
4. Lewis Tunstall, Leandro von Werra, and Thomas Wolf, "Natural Language Processing with Transformers" revised edition, o'reilly publication, 2022.

Continuous Course Assessment (CCA) Suggested:

1. Create a Chatbot with Google's Dialogflow
2. Building a Text Classification Model in Orange
3. Create Text Recommendation System with Gradio

B.E. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS) applicable for 2022 Scheme

SEMESTER – VII

Building Enterprise Applications (3:0:0:0) 3

(Effective from the academic year 2025-26)

Course Code	BCS703B	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	3:0:0:0	SEE Marks	50
Total Number of Contact Hours	40 (Theory)	Exam Hours	03

Course Objectives:

This course will enable students to:

1. To introduce the concepts of Enterprise applications and different issues related to their implementation.
2. To develop an understanding of enterprise application architectures and the design modeling techniques essential for their construction.
3. To provide knowledge on different testing techniques for Enterprise applications and methodologies used to roll out these applications.

Preamble:

In this course, we embark on a journey through the intricate landscape of modern business technology. We will uncover the foundational concepts, architectural paradigms, and strategic methodologies essential for designing, developing, and deploying the robust, large-scale software systems that drive today's enterprises.

Module – 1

Introduction to Enterprise application

Introduction to enterprise applications and their types, software engineering methodologies, life cycle of raising an enterprise application, key determinants of successful enterprise applications, and measuring the success of enterprise applications.

Textbook : Chapter-1: 1.1,1.2,1.3,1.4,1.5 **(8 Hours)**

Module – 2

Incepting enterprise application and business process modelling

Enterprise analysis, business modelling, Case study of EM Bank, requirements elicitation and analysis, use case modelling, User prototypes, Nonfunctional requirements, requirements validation, Planning and estimation.

Textbook: Chapter-2: 2.1, 2.2,2.3,2.4,2.5,2.6 **(8 Hours)**

Module – 3

Enterprise Architecture and designing enterprise application

Concept of architecture, views and viewpoints, enterprise architecture, logical architecture, technical architecture and design - different technical layers, data architecture and design – relational, XML, and other structured data representations, Infrastructure architecture and design elements - Networking, Internetworking, and Communication Protocols, IT Hardware and Software, Middleware, Policies for Infrastructure Management, Deployment Strategy.

Textbook: Chapter-3: 3.1, 3.2, 3.3, 3.4, 3.5, 3.6 **(8 Hours)**

Module – 4

Constructing enterprise application

Construction readiness of enterprise applications - defining a construction plan, defining a package structure, setting up a configuration management plan, setting up a development environment, introduction to the Software Construction Maps, construction of solution layers, Code review, static code analysis, build process and Unit testing, dynamic code

analysis.

Textbook: Chapter-4: 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7 **(8 Hours)**

Module – 5

Testing and rolling out enterprise application

Types and methods of testing an enterprise application, testing levels and approaches, testing environments, integration testing, System testing: performance testing, penetration testing, usability testing, globalization testing and interface testing, user acceptance testing, rolling out an enterprise application.

Textbook: Chapter-5: 5.1, 5.2, 5.3, 5.4, 5.5, 5.6 **(8 Hours)**

Course Outcomes:

1. Illustrate the fundamentals of enterprise applications and evaluation metrics.
2. Assess the inception of enterprise applications along with prototyping and estimation techniques.
3. Apply the concepts of enterprise architecture to design logical, technical, data, and infrastructure components of an enterprise application.
4. Analyze the construction and testing processes of enterprise applications, including development readiness, code quality practices, testing methods, and rollout strategies.

Textbook:

1. Raising Enterprise Applications: A Software Engineering Perspective, by Anubhav Pradhan, Satheesha B. Nanjappa, Senthil K. Nallasamy, Veerakumar Esakimuthu, Wiley-India Publication, Reprint Version 2011, Publication date 2010, ISBN: 9788126519460.

References:

1. Patterns of Enterprise Application Architecture, Martin Fowler, With Contributions from David Rice, Matthew Foemmel, Edward Hieatt, Robert Mee and Randy Stafford, Addison Wesley Publication, Reprint Version - 2016. ISBN 0-321-12742-0.
2. A systematic perspective to managing complexity with enterprise architecture by Pallab Saha, ISBN:9781466645189, 2013.

Continuous Course Assessment (CCA) suggested:

The assignment requires a two-part submission: a detailed project plan and a seminar presentation based on that plan. “The Enterprise Application Lifecycle: A Comprehensive Project Plan from Architectural Design to Final Deployment”

B.E. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS) applicable for 2022 Scheme

SEMESTER – VII

Deep Learning (3:0:0:0) 3

(Effective from the academic year 2025-26)

Course Code	BCS703C	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	3:0:0:0	SEE Marks	50
Total Number of Contact Hours	40(Theory)	Exam Hours	03

Course Objectives:

This course will enable the students to:

1. Understand the fundamental concepts of deep learning and their applications in various domains.
2. Implement and train deep learning models using different techniques.
3. Analyze the structure and working of convolutional neural networks.
4. Apply the neural networks to tasks like image classification and object detection.

Preamble: Deep learning is a sub-field of Machine learning, it is a key enabler of AI powered technologies being developed across the globe. In this course, students will learn an intuitive approach to build complex models that help machines to solve real-world problems with human-like intelligence. Deep learning is an aspect of data science that drives many applications and services that improve automation, performing analytical and physical tasks without human intervention. This enables development of products and services such as digital assistants, voice-enabled devices, self-driving cars, Generative AI and GANs.

Module – I

Introduction: What is a Neural Network, The Human Brain, Models of a Neuron, Neural Networks Viewed As Directed Graphs, Feedback, Network Architectures, Rosenblatt‟s Perceptron: Introduction, Perceptron, The Perceptron Convergence Theorem, Relation Between the Perceptron and Bayes Classifier for a Gaussian Environment.

Textbook 2: Ch 1.1 - 1.4 **(8 Hours)**

Module – II

Feedforward Networks: Introduction to feedforward neural networks, Gradient Based Learning:(Learning Distributions with Maximum Likelihood, Learning Conditional Statistics, Sigmoid units for Bernoulli Output Distributions, Softmax units for Multinoulli Output Distributions), Hidden Units, Architecture Design, Backpropagation and Computational graph with chaining rule, Backpropagation computation model.

Textbook 1: Ch 6, 6.2.1.1, 6.2.2.2, 6.2.2.3, 6.3.1, 6.3.2, 6.4, 6.5 (**8 Hours)**

Module – III

Optimization for Training Deep Models: Empirical Risk Minimization, Challenges in Neural Network Optimization, Basic Algorithms: Stochastic Gradient Descent, Parameter Initialization Strategies, Algorithms with Adaptive Learning Rates: The AdaGrad algorithm, The RMSProp algorithm, Choosing the right optimization algorithm.

Textbook1: Ch 8.1.1, 8.2 to 8.5. **(8 Hours)**

Module – IV

Convolutional Networks: The Convolution Operation, Pooling, Variants of the Basic Convolution Function, Structured Outputs, Data Types, Efficient Convolution Algorithms, Strategies for unsupervised CNs.

Textbook1: Ch 9.1 to 9.9 **(8 Hours)**

Module – V

Recurrent and Recursive Neural Networks: Unfolding Computational Graphs, Recurrent Neural Network, Bidirectional RNNs, Deep Recurrent Networks, Recursive Neural Networks, The Long Short-Term Memory.

Textbook1: Ch 10.1 to 10.6, 10.10 **(8 Hours)**

Course outcomes:

The student will be able to:

CO1: Apply the concepts of neural networks for scenario-based applications.

CO2: Apply the feedforward neural network concepts for deep learning architecture.

CO3: Analyze the modeling algorithms for optimization of neural networks.

CO4: Analyze the algorithms on Convolutional Neural Network (CNN) for decision making.

Text books:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, 2016.
2. Simon Haykin, Neural networks and Learning Machines, Third Edition, Pearson, 2016.

References:

1. Bengio, Yoshua. "Learning deep architectures for AI." Foundations and trends in Machine Learning, 2009.
2. N.D.Lewis, "Deep Learning Made Easy with R: A Gentle Introduction for Data Science", January 2016.
3. Nikhil Buduma, "Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms", O'Reilly publications.

Continuous Course Assessment (CCA) suggested:

1. Presentation on Case Study/Real-time Applications for Deep Learning.
2. Implementation of Sample Programs on Deep Learning Concepts.

B.E. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS) applicable for 2022 Scheme

SEMESTER – VII

Quantum Computing (3:0:0:0) 3

(Effective from the academic year 2025-26)

Course Code	BCS703D	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	3:0:0:0	SEE Marks	50
Total Number of Contact Hours	40(Theory)	Exam Hours	03

Course objectives:

This course will enable the students to:

1. Get the foundational understanding of quantum mechanics and its application to computation.
2. Master the principles and representation of qubits, including Bloch sphere visualization, superposition, entanglement, and qubit measurement.
3. Develop the ability to construct and evaluate quantum circuits,
4. Understand and implement key quantum algorithms and protocols
5. Learn quantum error correction and fault tolerance techniques.

Preamble: Quantum computing course aims to equip students with a foundational understanding of quantum mechanics and its application to computation. This includes learning about qubits, quantum gates, algorithms, and error correction. The course will also likely explore the practical aspects of quantum computing, such as using software tools like Qiskit.

Module – I

Introduction to Quantum Computing: History & Overview, Review of linear algebra: Dirac notation, Hilbert spaces, Unitary, Hermitian, and Normal matrices, Inner product, Outer product, Tensor product, Postulates of Quantum Mechanics, Stern and Gerlach experiment, Qubit, Bloch Sphere

Textbook 1: chapter 1.12.1,2.6

Textbook2:Chapter 2.1,2.4, 3.1,3.4,3.19 **(8 Hours)**

Module – II

Quantum Circuits & Programming: Circuit model, fundamental quantum gates (Pauli, Hadamard, CNOT), Constructing and analyzing circuits, implementing measurement, introduction to Qiskit programming, quantum parallelism, no-cloning theorem.

Textbook 1: chapter 1.3,4.5

Textbook2: Chapter 8.6,9.2, 9.8 Reference:6 **(8 Hours)**

Module – III

Introductory Quantum Algorithms

Probabilistic Versus Quantum Algorithm, Phase Kick-Back, The Deutsch Algorithm, The Deutsch–Jozsa Algorithm, Simon's Algorithm, Entanglement Bell states, entanglement theory, quantum teleportation and superdense coding, Grover's algorithm.

Textbook 1: chapter 5.1,5.2, 6.1,6.2,6.3, 6.4,6.5

Textbook2: Chapter 7.1 Reference :7 **(8 Hours)**

Module – IV

Algorithms with Super polynomial speed-up: Quantum Phase Estimation and the Quantum Fourier Transform, Eigen estimation, Finding- orders, Order Finding Problem, The Eigenvalue Estimation Approach to Order Finding, Shor's Approach to Order Finding. Algorithm for the Finite Abelian Hidden Subgroup Problem.

Textbook 1: Chapter 7.1,7.2 7.3, 7.5 **(8 Hours)**

Module – V

Quantum Error Correction: The Error Model, Classical Error Correction, The Classical Three-Bit Code, Fault Tolerance, Quantum Error Correction, Three- and Nine-Qubit Quantum Codes, Fault-Tolerant Quantum Computation

Textbook 1: Ch. 10.1-10.6 **(8 Hours)**

Course Outcomes:

The Students will be able to

CO1: Apply quantum computing components to model computational tasks in a quantum framework.

CO2: Analyze quantum circuit models to understand their functionality and limitations.

CO3: Evaluate core quantum algorithms for correctness and performance.

CO4: Design quantum error correction schemes for resilient quantum circuits.

Text Books:

1. Phillip Kaye, Raymond Laflamme et. al., An introduction to Quantum Computing, Oxford University press, 2007.
2. David Mcmahon, Quantum Computing Explained, Wiley-Interscience A John Wiley & Sons, Inc., Publication 2008.

References:

1. Quantum Computation and Quantum Information, M. A. Nielsen & I.Chuang, Cambridge University Press (2000).
2. Gilbert Strang, Introduction to Linear Algebra, Fifth Edition, Wellesley-Cambridge Press, 2016.
3. Andrew Childs, Ronald de Wolf, and John Watrous on quantum theory and computing
4. Stephen Barnett, Quantum Information, Oxford University Press, 2009.
5. Chuang and M.Nielsen, Quantum Computation and Quantum Information, Cambridge University Press, 2012
6. https://quantum.cloud.ibm.com/docs/en/guides
7. https://quantum.cloud.ibm.com/learning/en/courses/fundamentals-of-quantum-algorithms/grover-algorithm/grover-algorithm-description

Continuous Course Assessment (CCA) suggested:

1. **Quirk Quantum Circuit Simulation.**
   - Students design and simulate basic quantum logic gates and circuits using algassert.com/quirk.
   - Outcome: Hands-on familiarity with quantum gate operations
2. **Qiskit Quantum Circuit Coding.**
   - Students use Qiskit freeware to code and execute quantum circuits, mapping them to IBM Quantum Experience supercomputers.
   - Outcome: Practical coding and execution skills in a real quantum computing environment.
3. **Review Paper on Applications.**
   - Students prepare a review paper highlighting real-world applications of quantum computing, referencing existing implementations.
   - Outc*ome:* Research and analytical skills in current quantum computing trends.
4. **Poster/Presentation on Trends.**
   - Students design a poster or give a classroom presentation on topics such as quantum cryptography, quantum AI, or quantum algorithms.
   - Outcome: Ability to communicate technical concepts visually and verbally.
5. **Mini Project.**
   - **Bre**akup: Problem Definition (1M), Theoretical Understanding (2M), Implementation & Simulation (3M), Report (2M), Presentation (2M).
   - Outcome: End-to-end project experience with documentation and presentation skills.

B.E. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS) applicable for 2022 Scheme

SEMESTER – VII

Introduction to Algorithms (3:0:0:0) 3

(Effective from the academic year 2025-26)

Course Code	BCS704A	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	3:0:0:0	SEE Marks	50
Total Number of Contact Hours	40 (Theory)	Exam Hours	03

Course Objectives:

This course will enable students to:

1. Explain the methods of analyzing the algorithms and to analyse performance of Algorithms.
2. State algorithm's efficiencies using asymptotic notations.
3. Solve problems using algorithm design methods such as the brute force method, greedy method, divide and conquer, decrease and conquer, transform and conquer, Dynamic programming, backtracking and branch and bound.
4. Choose the appropriate data structure and algorithm design method for a specified Application.
5. Introduce P and NP classes.

Preamble:

The course Design and Analysis of Algorithms aims to provide a strong foundation in algorithmic problem solving and computational thinking. It introduces students to the concepts and techniques used in designing efficient algorithms and in analyzing their performance. This course prepares students to design algorithms using design techniques and also evaluate their correctness and efficiency, making them to solve complex computational problems in real-world scenarios.

Module – I

Introduction: What is an Algorithm, It's Properties. Algorithm Specification-using natural language, using Pseudo code convention, Fundamentals of Algorithmic Problem solving, Analysis Framework- Time efficiency and space efficiency, Worst-case, Best-case and Average case efficiency.

Performance Analysis: Estimating Space complexity and Time complexity of algorithms.

Asymptotic Notations: Big-Oh notation (O), Omega notation (Ω), Theta notation (Θ) with examples, Basic efficiency classes, Mathematical analysis of Non-Recursive and Recursive Algorithms with Examples problems. Solving Recurrences: Master Theorem, Substitution Method, Iteration Method. **Book-1: 2.1,2.2, 2.3, 2.4, 5(Introduction, Master Theorem).**

Textbook -2: 1.1, 1.2, 1.3. **(8 Hours)**

Module – II

Review of Basic Data Structures: Arrays, linked lists, stacks, queues, Trees, Graphs.

Brute force design technique: String matching algorithm with complexity.

Searching Algorithms: Linear Search, Binary Search.

Sorting Algorithms: Selection Sort, Insertion Sort, Bubble Sort, (Divide and Conquer) Merge Sort, Quick Sort.

Time Complexity Analysis of Sorting Algorithms: Understanding the efficiency of various sorting techniques.

Textbook -1: 1.4, 2.1-2.4, 3.1-3.2, 4.1, 4.4 (binary search), 5.1, 5.2.

Textbook -2: 3.5, 3.6. **(8 Hours)**

Module – III

Greedy Algorithms: Knapsack Problem and Dijkstra's algorithm for shortest paths.
Dynamic Programming: Fibonacci Series, 0/1 Knapsack and Traveling Salesperson Problem.
Backtracking: N-Queen Problem and Sub-Set problem.
Textbook -1: 2.5.
Textbook -2: 4.3, 4.9, 5.7, 5.9, 7.2, 7.3. **(8 Hours)**

Module – IV

Graph Traversal Algorithms: Breadth-First Search (BFS) and Depth-First Search (DFS), All pair Shortest Path - Floyd's Algorithm, Transitive closure – Warshall's Algorithm.
Decrease and Conquer: Insertion Sort, Topological Ordering.
Transform and Conquer Approach: Introduction, Heaps and Heap Sort.
Textbook -1: 3.5, 4.1, 4.2, 6.4, 8.4. **(8 Hours)**

Module – V

NP-Complete and NP-Hard problems: Basic concepts, non- deterministic algorithms, P, NP, NP-Complete, and NP-Hard classes. Approximation Algorithms, and an introduction to Randomized Algorithms.
Textbook -1: 11.3, 12.2, 12.3.
Textbook -2: 14.4. **(8 Hours)**

Course outcomes
At the end of the course the student will be able to:
CO1: Make use of asymptotic notations and mathematically represent the complexity of the algorithm.
CO2: Apply searching/sorting algorithms and brute force methods to solve the computational problems.
CO3: Analyze Greedy Method, dynamic programming techniques to solve the problems.
CO4: Illustrate backtracking, approximation algorithms and various classes P, NP, NP Complete and NP Hard problems.

Textbooks
1. Introduction to the Design and Analysis of Algorithms, Anany Levitin: 3rd Edition, 2017.Pearson.
2. Computer Algorithms/C++, Ellis Horowitz, Satraj Sahni and Rajasekaran, 2nd Edition, 2014, Universities Press.

Reference Books
1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI, 2009.
2. Design and Analysis of Algorithms, S. Sridhar, Oxford (Higher Education), 2014.

Alternate Assessment Tools (CCA) suggested:
1. MOOCs (Udemy / VTU / Springboard)

B.E. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS) applicable for 2022 Scheme

SEMESTER – VII

Introduction to Augmented Reality and Virtual Reality (3:0:0:0) 3

(Effective from the academic year 2025-26)

Course Code	BCS704B	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	3:0:0:0	SEE Marks	50
Total Number of Contact Hours	40 (Theory)	Exam Hours	03

Course Objectives:

This course will enable students to:

1. To gain a foundational understanding of Augmented Reality (AR) and Virtual Reality (VR).
2. To acquire essential technical skills for AR and VR development, including programming, 3D modelling, and using game development engines like Unity.
3. To understand how AR systems track objects and estimate their positions.
4. To gain knowledge of 3D computer graphics fundamentals, rendering techniques, and the tools and libraries available for 3D modelling and rendering in AR.
5. To learn design principles and interaction techniques specific to AR, including software architecture, design patterns, and user interface design.

Preamble:

This course introduces the fundamentals of Augmented Reality (AR) and Virtual Reality (VR), focusing on their growing significance in areas like education, healthcare, industry, and entertainment. It equips students with essential technical skills such as 3D modelling, real-time rendering, object tracking, and development using tools like Unity. Emphasizing both theory and practical application, the course covers system design, user interaction, and pose estimation, enabling students to build immersive AR/VR solutions for real-world challenges.

Module – 1

Introduction to Augmented Reality: History of AR, AR Scenarios, the future of AR, Applications of AR.

Virtually Everything for Everyone: What is Virtual reality, Differences between virtual reality and Augmented reality, Applications versus games, Types of VR experiences, types of HMD, How virtual reality works?, Technical Skills that are important to VR.

Textbook 1: Chapter 1– 1.1 to 1.6 and Textbook 2: Chapter 1- 1.1 to 1.7 **(8 Hours)**

Module – 2

Understanding Unity, Content and scale: Technical requirements, Installing Unity, Getting started with unity, Creating a simple diorama.

Setting up your project for VR: Introducing the Unity XR platform, Choosing your target VR Platform and toolkits, Enabling virtual reality for your platform, Building and run your project, Building for Oculus Quest.

Textbook 2: Chapter 2 – 2.1 to 2.4 , Chapter 3 – 3.2 to 3.9 **(8 Hours)**

Module – 3

Pose Estimation and Tracking: Pose Tracking in AR, Classifications of Tracking, Stationary Tracking System, Mobile Sensor-Based Tracking, Optical Tracking, Hybrid Tracking, and Marker- Based Tracking and AR, Diminished Reality, Marker-less Tracking and AR.

Textbook 1: : Chapter 3- 3.1 to 3.9 **(8 Hours)**

Module – 4

3D Graphics in AR: Basics of 3D Computer Graphics, 3D Rendering, 3D Model

Importers/Loaders, 3D modeling software's, Available Graphics libraries.
Textbook 1: Chapter 5 – 5.1 to 5.5 **(8 Hours)**

Module – 5

Designing AR Systems: Design principles for AR, Designing interactions for AR, Software Architecture and Design Patterns for AR, Designing AR interfaces, Examples of AR Interfaces.
Textbook 1: Chapter 6- 6.1 to 6.5 **(8 Hours)**

Course Outcomes:

The students will be able to:

CO1: Explain AR/VR concepts, applications, HMDs, and technical skills.

CO2: Apply Unity and XR tools to configure, build, and run VR projects with Oculus Quest.

CO3: Analyze pose estimation, object tracking, 3D graphics, and interaction design techniques for AR/VR applications.

CO4**:** Design AR/VR systems by integrating models, rendering, interaction mechanisms, and deploy on target platforms.

Textbooks:

1. Chethankumar G Shetty, "Augmented Reality: Theory, Design and Development", McGrawHill Publications 2020.
2. Jonathan Linowes-"Unity 2020 Virtual Reality Project" Third Edition Packet> paperback – 30 Jully 2020.

References:

1. Jonathan Linowes, Krystian Babilinski, "Augmented Reality for Developers: Build practical augmented reality applications with unity, ARCore, ARKit and Vuforia" , Paperback – Import, Packt Publishing Limited 9 October2017.
2. Schmalstieg/Hollerer, "Augmented Reality: Principle & Practice", Paperback-12, Person Education India, October 2016.
3. Chitra Lele, "Artificial Intelligence Meets Augmented Reality: Redefining Regular Reality" , Paperback- 1, BPB Publications, January 2019.

Continuous Course Assessment (CCA) suggested:

1. Design and development of AR/VR model.
2. MOOC course (Udemy / VTU / Springboard)

B.E. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS) applicable for 2022 Scheme

SEMESTER – VII

Introduction to Cyber Security (3:0:0:0) 3

(Effective from the academic year 2025-26)

Course Code	BCS704C	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	3:0:0:0	SEE Marks	50
Total Number of Contact Hours	40 (Theory)	Exam Hours	03

Course Objectives:

This course will enable students to:

1. To familiarize cybercrime terminologies and perspectives
2. To understand Cyber Offenses and Botnets
3. To gain knowledge on tools and methods used in cybercrimes
4. To understand phishing and computer forensics

Preamble:

In this course, It outlines the purpose, scope, and guiding principles of a cybersecurity policy or strategy. It establishes the context for why the policy is needed, what it aims to protect, and who it applies to.

Module – 1

Introduction to Cybercrime: Cybercrime: Definition and Origins of the Word, Introduction to Cybercrime and Information Security, Who are Cybercriminals, Cyber-attacks and **Threats**: Classifications of Cybercrimes, An Indian Perspective, Global Perspectives.

Text book : Chapter 1 (1.1 to 1.5, 1.7, 1.9) **(8 Hours)**

Module – 2

Cyber Offenses: How Criminals Plan Them: Introduction, How criminals plan the attacks, Social Engineering, Cyber Stalking, Cybersafe & cybercrimes. Botnets: The fuel for cybercrime, Attack Vector.

Text Book : Chapter 2 -2.1 to 2.7 **(8 Hours)**

Module – 3

Tools and Methods used in Cybercrime: Introduction, Proxy Servers, Anonymizers, Phishing, Password Cracking, Key Loggers and Spyware, Virus and Worms, Trozen Horses and Backdoors, Steganography, DoS and DDOS Attacks, Attacks on Wireless networks.

Text Book : Chapter 4 - 4.1 to 4.9,4.12 **(8 Hours)**

Module – 4

Phishing and Identity Theft: Introduction, methods of phishing, phishing, phishing techniques, spear phishing, types of phishing scams, phishing toolkits and spy phishing, counter measures, Identity Theft.

Text Book: Chapter 5- 5.1 to 5.3 **(8 Hours)**

Module – 5

Understanding Computer Forensics and Cyber laws: Introduction, Historical Background of Cyber forensics, Digital Forensics Science, Need for Computer Forensics, Cyber Forensics and Digital Evidence, Digital Forensic Life cycle, Chain of Custody Concepts, network forensics. Hacking and Indian Laws.

Text Book: Chapter 7 -7.1. to 7.5, 7.7 to 7.9 , Chapter 1-1.8 **(8 Hours)**

Course Outcomes:

The students will be able to:

CO1: Explain the fundamentals of cybercrime, threats, and global perspectives.

CO2: Analyze the software vulnerabilities and security solutions to reduce the risks.

C03: Design and modify security architectures considering organizational requirements.
C04: Apply forensic tools and methods to investigate cybercrimes, and interpret legal frameworks.

Textbooks

1. Sunit Belapure and Nina Godbole, "Cyber Security: Understanding Cyber Crimes, Computer Forensics And Legal Perspectives", Wiley India Pvt Ltd, ISBN: 978-81- 265-21791, 2011, First Edition (Reprinted 2018)

References

1. Donaldson, S., Siegel, S., Williams, C.K., and Aslam, A. (2015). —Enterprise Cyber security -How to Build a Successful Cyber defense Program against Advanced Threats||, Apress, 1st Edition.
2. Network Security Bible, Eric Cole, Ronald Krutz, James W. Conley, 2nd Edition, Wiley India Pvt. Ltd, 2011.

Continuous Course Assessment (CCA) suggested:

1. Illustration of standard case study of cybercrime and Vulnerabilities.

B.E. COMPUTER SCIENCE AND ENGINEERING

Choice Based Credit System (CBCS) applicable for 2022 Scheme

SEMESTER – VII

Parallel Computing Laboratory (0:0:1:0) 1

(Effective from the academic year 2025-26)

Course Code	BCSL706	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	0:0:1:0	SEE Marks	50
Total Number of Contact Hours	26 (Practical)	Exam Hours	03

Course Objectives:

This course will enable students to:

1. Design and implement high performance versions of standard single threaded algorithms.
2. Demonstrate the architectural features in the GPU and MIC hardware accelerators.
3. Design programs to extract maximum performance in a multicore, shared memory execution environment processor.
4. Develop programs using OPENMP, MPI and CUDA.

Program List

PART A

1.	Given a nxn matrix A and a vector x of length n, their product y=A·x. Write a program to implement the multiplication using OpenMP PARALLEL directive.
2.	Consider a Scenario where a person visits a supermarket for shopping. He purchases various items in different sections such as clothing, gaming, grocery, stationary. Write an OpenMP program to process his bill parallelly in each section and display the final amount to be paid. (Sum of elements parallelly).
3.	Consider a Person named X on the earth; to find his accurate position on the globe we require the value of Pi. Write a program to compute the value of pi function by Numerical Integration using OpenMP PARALLEL section.
4	Using OpenMP, design and develop a multi-threaded program to generate and print Fibonacci Series. One thread must generate the numbers up to the specified limit and another thread must print them. Ensure proper synchronization
5.	A university awards gold medals to students who have achieved the highest CGPA. Write a program using OpenMP to find the student with the highest CGPA.
6.	Multiply two square matrices (1000, 2000 or 3000 dimensions). Compare the performance of a sequential and parallel algorithm using OpenMP.
7.	Assume you have n robots which pick mangoes in a farm. Write a program to calculate the total number of mangoes picked by n robots parallelly using MPI.
8.	Design a MPI program that uses blocking send/receive routines and non-blocking send/receive routines.
9.	Design a program that implements application of MPI Collective Communications.

Course Outcomes:

CO1: Apply OpenMP parallel programming constructs to develop efficient shared-memory applications.
CO2: Demonstrate the use of MPI to develop message-passing parallel programs for distributed memory architectures.
CO3: Design and implement GPU solutions for compute-intensive problems using CUDA.

Text Books

1. Introduction to parallel computing Ananth Grama, Anshul Gupta, Vipin Kumar, George Karypis Pearson education publishers second edition, 2003.
2. Programming Massively Parallel Processors on Approach David B Kirk, Wenmei W. Hwu Elsevier and nvidia publishers First edition 2010.
3. Introduction to High Performance Computing for Scientists and Engineers Georg Hager, Gerhard Wellein Taylor and Francis Group, LLC, CRC Press 2011.

Reference Books

1. Parallel Programming for Multicore and cluster systems Thomas Rauber and Gudula Runger Springer International Edition, 2009.
2. Parallel Programming in C with MPI and Open MP Michael J. Quin McGraw Hill 1st Edition, 2003.