Robotic Process Automation Design and Development (UiPath)		Semester	7
Course Code	**BCG701**	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 hours Theory + 8-10 Lab slots	Total Marks	100
Credits	04	Exam Hours	03
Examination nature (SEE)	Theory/practical		

Course objectives:
This course will enable students to,

- Understand fundamental concepts of automation using UiPath StudioX.
- Learn and Understand UI Automation activities.
- Learn and Understand Mail Automation and Word Automation activities.
- Learn and Understand Excel Automation activities.
- Learn and Understand File Automation and Presentation Automation activities.

Teaching-Learning Process (General Instructions)
These are sample Strategies; that teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based-Learning (PBL), which fosters student's Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.

MODULE-1

Robotic Process Automation: Overview: Return on Investment (ROI), Automation Types, UiPath StudioX. **Common Concepts:** Notebook, Activity Inputs, Activity Outputs, Common Properties, Common Activities, Write Line: Message Box, Input Dialog. Modify Text., Text to Left/Right, Delay, if, Switch., Repeat Number Of Times. , Skip Current, Exit Loop, Get Username/Password, Get Orchestrator Asset, Save For Later, Wait for Download, Group.

Chapter 1 , Chapter 3

MODULE-2

UI Automation: Sample Overview, Activities Reference, Use Application/Browser, Go To URL, Navigate Browser, Highlight, Take Screenshot, Check App State, Click, Type Into, Select Item, Check/Uncheck, Get Text, Get Attribute, Extract Table Data, Hover, Keyboard Shortcuts, Get Active Window, Maximize Window, Minimize Window, Hide Window, Restore Window, Move Window. App/Web Recorder.

Chapter 4

MODULE-3

Mail Automation: Sample Overview: Desktop Outlook Setup, File System Structure, Activities Reference: Use Desktop Outlook App, Use Gmail, For Each Email, Mark Email As Read/Unread, Forward Email, Save Email Attachments, Save Email, Send Email, Send Calendar Invite, Move Email, Reply to Email, Archive Email, Delete Email. **Word Automation:** Sample Overview: Word Setup, File System Structure, Activities Reference: Use Word File..., Save Document As, Read Text, Set Bookmark Contant, Replace Text in Document, Append Text, Insert DataTable in Document, Replace Picture,Add Picture, Save Document as PDF.

Chapter 5 , Chapter 6

MODULE-4

Excel Automation: Sample Overview, Activities Reference, Use Excel File, Insert Sheet, Rename Sheet, Duplicate

Sheet, Delete Sheet, For Each Excel Sheet, Insert Column,Text To Columns..,Delete Column, Insert Rows, Delete Rows,Find First/Last Data Row, For Each Excel Row, Write Cell, Create Pivot Table, Format as Table, Change Pivot Data Source, Refresh Pivot Table, Append Range, Copy Range, Sort Range, Clear Sheet/Range/Table, Auto Fill, Fill Range, Write Range, Read Cell Formula.., Read Cell Value, Format Cells, Export to CSV.., Save Excel File, Save Excel File As...,Save Excel File As PDF, VLookup, Filter, Run Spreadsheet Macro. **Chapter 7**
MODULE-5
File Automation: Sample Overview. Activities Reference: Get Folder Info, Folder Exists, Create Folder, Delete Folder, Copy Folder., Move Folder, For Each File In Folder, Compress/Zip Files, Extract/Unzip Files, Get File Info, File Exists, Create File, Delete File, Copy File, Move File, Write Text File, Append Line.., Read Text File. **Presentation Automation:** Sample Overview, File System Structure., Activities Reference, Use PowerPoint Presentation, Copy Paste Slide, Delete Slide, Add New Slide, Replace Text in Presentation, Add Text to Slide., Add Data Table to Slide, Add Image/Video to Slide, Add File to Slide., Run Presentation Macro, Save PowerPoint File As., Save Presentation as PDF. **Chapter 9, Chapter 10**

PRACTICAL COMPONENT OF IPCC *(May cover all / major modules)*

Sl.NO	Experiments
1	Develop automation in UiPath StudioX to demonstrate the following activities: Write Line: Message Box, Input Dialog. Modify Text., Text to Left/Right and Delay
2	Develop automation in UiPath StudioX to demonstrate the following activities: if, Switch., Repeat Number Of Times. , Skip Current, Exit Loop
3	Develop UI automation in UiPath StudioX to demonstrate the following activities: Use Application/Browser, Go To URL, Navigate Browser, Highlight and Take Screenshot
4	Develop UI automation in UiPath StudioX to demonstrate the following activities: Check App State, Click, Type Into, Select Item, Check/Uncheck, Get Text, Get Attribute, Extract Table Data and Hover.
5	Develop UI automation in UiPath StudioX to demonstrate the following activities: Get Active Window, Maximize Window, Minimize Window, Hide Window, Restore Window and Move Window.
6	Develop Word automation in UiPath StudioX to demonstrate the following activities: Use Word File.., Save Document As, Read Text, Replace Text in Document, Append Text, Replace Picture, Add Picture, Save Document as PDF.
7	Develop Excel automation in UiPath StudioX to demonstrate the following activities: Use Excel File, Insert Sheet, Rename Sheet, Duplicate Sheet, Delete Sheet, For Each Excel Sheet, Insert Column, Text To Columns, Delete Column.
8	Develop Excel automation in UiPath StudioX to demonstrate the following activities: Insert Rows, Delete Rows, Find First/Last Data Row, For Each Excel Row, Write Cell, Create Pivot Table., Save Excel File As..., Save Excel File As PDF
9	Develop Excel automation in UiPath StudioX to demonstrate the following activities: Use Excel File, Insert Sheet, Rename Sheet, Duplicate Sheet, Delete Sheet, For Each Excel Sheet, Insert Column, Text To Columns., Delete Column.
10	Develop Excel automation in UiPath StudioX to demonstrate the following activities: Refresh Pivot Table, Append Range, Copy Range, Sort Range, Clear Sheet/Range/Table, Auto Fill, Fill Range, Write Range, Read Cell Formula.., Read Cell Value, Format Cells, Export to CSV
11	Develop File automation in UiPath StudioX to demonstrate the following activities: Get Folder Info, Folder Exists, , For Each File In Folder, Compress/Zip Files, Extract/Unzip Files, Get File Info, File Exists, Create File, Delete File, Copy File, Move File, Write Text File, Append Line.., Read Text File
12	Develop Presentation automation in UiPath StudioX to demonstrate the following activities: Use PowerPoint Presentation, Copy Paste Slide, Delete Slide, Add New Slide, Add Text to Slide., Add Data Table to Slide, Add Image/Video to Slide, Save PowerPoint File As.,Save Presentation as PDF.

Course outcomes (Course Skill Set):
At the end of the course, the student will be able to:

- Demonstrate Common RPA concepts using UiPath StudioX.
- Develop UI automation in UiPath StudioX.
- Implement Mail automation and Word automation in UiPath StudioX.
- Develop Excel automation in UiPath StudioX.
- Implement File automation and Presentation automation in UiPath StudioX.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

CIE for the theory component of the IPCC (maximum marks 50)

- IPCC means practical portion integrated with the theory of the course.
- CIE marks for the theory component are **25 marks** and that for the practical component is **25 marks**.
- 25 marks for the theory component are split into **15 marks** for two Internal Assessment Tests (Two Tests, each of 15 Marks with 01-hour duration, are to be conducted) and **10 marks** for other assessment methods mentioned in 22OB4.2. The first test at the end of 40-50% coverage of the syllabus and the second test after covering 85-90% of the syllabus.
- Scaled-down marks of the sum of two tests and other assessment methods will be CIE marks for the theory component of IPCC (that is for **25 marks)**.
- The student has to secure 40% of 25 marks to qualify in the CIE of the theory component of IPCC.

CIE for the practical component of the IPCC

- **15 marks** for the conduction of the experiment and preparation of laboratory record, and **10 marks** for the test to be conducted after the completion of all the laboratory sessions.
- On completion of every experiment/program in the laboratory, the students shall be evaluated including viva-voce and marks shall be awarded on the same day.
- The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report. Each experiment report can be evaluated for 10 marks. Marks of all experiments' write-ups are added and scaled down to **15 marks**.
- The laboratory test **(duration 02/03 hours)** after completion of all the experiments shall be conducted for 50 marks and scaled down to **10 marks.**
- Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for **25 marks**.
- The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

SEE for IPCC

Theory SEE will be conducted by University as per the scheduled timetable, with common question

papers for the course (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored by the student shall be proportionally scaled down to 50 Marks.

The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only. Questions mentioned in the SEE paper may include questions from the practical component.

Suggested Learning Resources:

Textbooks

1. Adeel Javed, Anum Sundrani, Nadia Malik, Sidney Madison Prescott, Robotic Process Automation using UiPath StudioX: A Citizen Developer's Guide to Hyperautomation, 1st Edition, Apress, April 2021

Reference Books:

1. Tom Taulli , The Robotic Process Automation Handbook : A Guide to Implementing RPA Systems, 2020, ISBN-13 (electronic): 978-1-4842-5729-6, Publisher : Apress
2. Alok Mani Tripathi, Learning Robotic Process Automation, Publisher: Packt Publishing Release Date: March 2018 ISBN: 9781788470940

Web links and Video Lectures (e-Resources):

- https://docs.uipath.com/studio/standalone/2022.10/user-guide/install-studio
- https://docs.uipath.com/studiox/standalone/2024.10/user-guide/automation-basics
- https://docs.uipath.com/studiox/standalone/2024.10/user-guide/common-activities
- https://docs.uipath.com/studiox/standalone/2024.10/user-guide/ui-automation
- https://www.youtube.com/watch?v=QE3dFUlTb-A
- https://docs.uipath.com/studiox/standalone/2024.10/user-guide/excel-automation
- https://www.youtube.com/watch?v=R-QszWgB29A
- https://docs.uipath.com/studiox/standalone/2024.10/user-guide/csv-automation
- https://www.youtube.com/watch?v=hBV4-FkR31I
- https://docs.uipath.com/studiox/standalone/2024.10/user-guide/mail-automation
- https://www.youtube.com/watch?v=4Tu4-cwuKqY
- https://docs.uipath.com/studiox/standalone/2024.10/user-guide/word-automation
- https://www.youtube.com/watch?v=NsRejhgvrsI
- https://docs.uipath.com/studiox/standalone/2024.10/user-guide/powerpoint-automation
- https://www.youtube.com/watch?v=Ht2I3IqNmLI

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

1. Installation of UiPath StudioX (Refer: https://docs.uipath.com/studio/standalone/2022.10/user-guide/install-studio)
2. Demonstrate the difference between UiPath Studio and UiPath StudioX
3. Course project (Automation of a process) – 10 Marks

PARALLEL COMPUTING		Semester	VII
Course Code	**BCS702**	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 hours Theory + 8-10 Lab slots	Total Marks	100
Credits	04	Exam Hours	03
Examination nature (SEE)	Theory/Practical		

Course objectives:

This course will enable to,

- Explore the need for parallel programming
- Explain how to parallelize on MIMD systems
- To demonstrate how to apply MPI library and parallelize the suitable programs
- To demonstrate how to apply OpenMP pragma and directives to parallelize the suitable programs
- To demonstrate how to design CUDA program

Teaching-Learning Process (General Instructions)

These are sample Strategies that teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only traditional lecture methods, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Programming assignment, which fosters student's Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.

MODULE-1

Introduction to parallel programming, Parallel hardware and parallel software – Classifications of parallel computers, SIMD systems, MIMD systems, Interconnection networks, Cache coherence, Shared-memory vs. distributed-memory, Coordinating the processes/threads, Shared-memory, Distributed-memory.

MODULE-2

GPU programming, Programming hybrid systems, MIMD systems, GPUs, Performance – Speedup and efficiency in MIMD systems, Amdahl's law, Scalability in MIMD systems, Taking timings of MIMD programs, GPU performance.

MODULE-3

Distributed memory programming with MPI – MPI functions, The trapezoidal rule in MPI, Dealing with I/O, Collective communication, MPI-derived datatypes, Performance evaluation of MPI programs, A parallel sorting algorithm.

MODULE-4

Shared-memory programming with OpenMP – openmp pragmas and directives, The trapezoidal rule, Scope of variables, The reduction clause, loop carried dependency, scheduling, producers and consumers, Caches, cache coherence and false sharing in openmp, tasking, tasking, thread safety.

MODULE-5

GPU programming with CUDA - GPUs and GPGPU, GPU architectures, Heterogeneous computing, Threads, blocks, and grids Nvidia compute capabilities and device architectures, Vector addition, Returning results from CUDA kernels, CUDA trapezoidal rule I, CUDA trapezoidal rule II: improving performance, CUDA trapezoidal rule III: blocks with more than one warp.

PRACTICAL COMPONENT OF IPCC

Sl.NO	Experiments
1	Write a OpenMP program to sort an array on n elements using both sequential and parallel mergesort(using Section). Record the difference in execution time.
2	Write an OpenMP program that divides the Iterations into chunks containing 2 iterations, respectively (OMP_SCHEDULE=static,2). Its input should be the number of iterations, and its output should be which iterations of a parallelized for loop are executed by which thread. For example, if there are two threads and four iterations, the output might be the following: a. Thread 0 : Iterations 0 −− 1 b. Thread 1 : Iterations 2 −− 3
3	Write a OpenMP program to calculate n Fibonacci numbers using tasks.
4	Write a OpenMP program to find the prime numbers from 1 to n employing parallel for directive. Record both serial and parallel execution times.
5	Write a MPI Program to demonstration of MPI_Send and MPI_Recv.
6	Write a MPI program to demonstration of deadlock using point to point communication and avoidance of deadlock by altering the call sequence
7	Write a MPI Program to demonstration of Broadcast operation.
8	Write a MPI Program demonstration of MPI_Scatter and MPI_Gather
9	Write a MPI Program to demonstration of MPI_Reduce and MPI_Allreduce (MPI_MAX, MPI_MIN, MPI_SUM, MPI_PROD)

Course outcomes (Course Skill Set):

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

- Explain the need for parallel programming
- Demonstrate parallelism in MIMD system.
- Apply MPI library to parallelize the code to solve the given problem.
- Apply OpenMP pragma and directives to parallelize the code to solve the given problem
- Design a CUDA program for the given problem.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

CIE for the theory component of the IPCC (maximum marks 50)

- IPCC means practical portion integrated with the theory of the course.
- CIE marks for the theory component are **25 marks** and that for the practical component is **25 marks**.
- 25 marks for the theory component are split into **15 marks** for two Internal Assessment Tests (Two Tests, each of 15 Marks with 01-hour duration, are to be conducted) and **10 marks** for other assessment methods mentioned in 22OB4.2. The first test at the end of 40-50% coverage of the syllabus and the second test after covering 85-90% of the syllabus.
- Scaled-down marks of the sum of two tests and other assessment methods will be CIE marks for the theory component of IPCC (that is for **25 marks)**.
- The student has to secure 40% of 25 marks to qualify in the CIE of the theory component of IPCC.

CIE for the practical component of the IPCC

- **15 marks** for the conduction of the experiment and preparation of laboratory record, and **10 marks** for the test to be conducted after the completion of all the laboratory sessions.
- On completion of every experiment/program in the laboratory, the students shall be evaluated including viva-voce and marks shall be awarded on the same day.
- The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report. Each experiment report can be evaluated for 10 marks. Marks of all experiments' write-ups are added and scaled down to **15 marks**.
- The laboratory test **(duration 02/03 hours)** after completion of all the experiments shall be conducted for 50 marks and scaled down to **10 marks.**
- Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for **25 marks**.
- The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

SEE for IPCC

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored by the student shall be proportionally scaled down to 50 Marks

The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only. Questions mentioned in the SEE paper may include questions from the practical component.

Suggested Learning Resources:

Textbook:

1. Peter S Pacheco, Matthew Malensek – An Introduction to Parallel Programming, second

edition, Morgan Kauffman.
2. Michael J Quinn – Parallel Programming in C with MPI and OpenMp, McGrawHill.

Reference Books:

1. Calvin Lin, Lawrence Snyder – Principles of Parallel Programming, Pearson
2. Barbara Chapman – Using OpenMP: Portable Shared Memory Parallel Programming, Scientific and Engineering Computation
3. William Gropp, Ewing Lusk – Using MPI:Portable Parallel Programing, Third edition, Scientific and Engineering Computation

Web links and Video Lectures (e-Resources):

1. Introduction to parallel programming: https://nptel.ac.in/courses/106102163

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Programming Assignment at higher bloom level (10 Marks)

	CRYPTOGRAPHY & NETWORK SECURITY		Semester	7
	Course Code	**BCS703**	CIE Marks	50
	Teaching Hours/Week (L:T:P: S)	4:0:0:0	SEE Marks	50
	Total Hours of Pedagogy	50	Total Marks	100
	Credits	04	Exam Hours	3
	Examination type (SEE)	Theory		

Course objectives:

1. Understand the basics of Cryptography concepts, Security and its principle
2. To analyse different Cryptographic Algorithms
3. To illustrate public and private key cryptography
4. To understand the key distribution scenario and certification
5. To understand approaches and techniques to build protection mechanism in order to secure computer networks

Teaching-Learning Process

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) needs not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding
9. Use any of these methods: Chalk and board, Active Learning, Case Studies

Module-1 10 hours

A model for Network Security, Classical encryption techniques: Symmetric cipher model, Substitution ciphers-Caesar Cipher, Monoalphabetic Cipher, Playfair Cipher, Hill Cipher, Polyalphabetic Ciphers, One time pad, Steganography.

Block Ciphers and Data Encryption Standards: Traditional Block Cipher structures, data Encryption Standard (DES), A DES Example, The strength of DES, Block cipher design principles.

Chapter 1: 1.8 Chapter 3: 3.1, 3.2, 3.5 Chapter 4: 4.1, 4.2, 4.3, 4.4, 4.5

Module-2 10 hours

	Pseudorandom number Generators: Linear Congruential Generators, Blum Blum Shub Generator. Public key cryptography and RSA: Principles of public key cryptosystems-Public key cryptosystems, Applications for public key cryptosystems, Requirements for public key cryptography, Public key Cryptanalysis, The RSA algorithm: Description of the Algorithm, Computational aspects, The Security of RSA. Diffie-Hellman key exchange: The Algorithm, Key exchange Protocols, Man-in-the-middle Attack, Elliptic Curve Cryptography: Analog of Diffie-Hellman key Exchange, Elliptic Curve Encryption/Decryption, Security of Elliptic Curve Cryptography. Chapter 8: 8.2 Chapter 9: 9.1, 9.2 Chapter 10: 10.1, 10.4
	Module-3 10 hours
	Applications of Cryptographic Hash functions, Two simple Hash functions, Key management and distribution: Symmetric key distribution using symmetric encryption, Symmetric key distribution using asymmetric encryption, Distribution of public keys, X.509 Certificates, Public Key Infrastructures Chapter 11: 11.1, 11.2 Chapter 14: 14.1, 14.2, 14.3, 14.4, 14.5
	Module-4 10 hours
	User Authentication: Remote user authentication principles**,** Kerberos**,** Remote user authentication using asymmetric encryption. Web security consideration, Transport layer security. Email Threats and comprehensive email security, S/MIME, Pretty Good Privacy. Chapter 15: 15.1, 15.3, 15.4 Chapter 17: 17.1, 17.2 Chapter 19: 19.3, 19.4, 19.5
	Module-5 10 hours
	Domainkeys Identified Mail. IP Security: IP Security overview, IP Security Policy, Encapsulating Security Payload, Combining security associations, Internet key exchange. Chapter 19: 19.9 Chapter 20: 20.1, 20.2, 20.3, 20.4, 20.5

Course outcome

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

CO1: Explain the basic concepts of Cryptography and Security aspects

CO2: Apply different Cryptographic Algorithms for different applications

CO**3:** Analyze different methods for authentication and access control.

CO4: Describe key management, key distribution and Certificates.

CO5: Explain about Electronic mail and IP Security.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

Books

Text Books:

William stallings, "Cryptography and Network Security", Pearson Publication, Seventh Edition.

References:

1. Keith M Martin, "Everyday Cryptography", Oxford University Press
2. V.K Pachghare,"Cryptography and Network Security", PHI, 2nd Edition

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Group assignment (TWO) to implement Cryptographic Algorithms (15 + 10 marks)

	Deep Learning		Semester	7
	Course Code	**BCS714A**	CIE Marks	50
	Teaching Hours/Week (L: T:P: S)	3:0:0:0	SEE Marks	50
	Total Hours of Pedagogy	40	Total Marks	100
	Credits	03	Exam Hours	03
	Examination type (SEE)	Theory		

Course objectives:

- Understand the basic concepts of deep learning.
- Know the basic working model of Convolutional Neural Networks and RNN in decision making.
- Illustrate the strength and weaknesses of many popular deep learning approaches.
- Introduce major deep learning algorithms, the problem settings, and their applications to solve real world problems

Teaching-Learning Process (General Instructions)

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation/Demonstration to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem/Practical Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills, and practical skill such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Use animations/videos to help the students to understand the concepts.
7. Demonstrate the concepts using PYTHON and its libraries wherever possible

Module-1

Introducing Deep Learning: **Biological and Machine Vision:** Biological Vision, Machine Vision: The Neocognitron, LeNet-5, The Traditional Machine Learning Approach, ImageNet and the ILSVRC, AlexNet, TensorFlow Playground. **Human and Machine Language:** Deep Learning for Natural Language Processing: Deep Learning Networks Learn Representations Automatically, Natural Language Processing, A Brief History of Deep Learning for NLP, Computational Representations of Language: One-Hot Representations of Words, Word Vectors, Word-Vector Arithmetic, word2viz, Localist Versus Distributed Representations, Elements of Natural Human Language.

Text book 2 : Chapter 1, 2

Module-2

Regularization for Deep Learning: Parameter Norm Penalties, Norm Penalties as Constrained Optimization, Regularization and Under-Constrained Problems, Dataset Augmentation, Noise Robustness, Semi- Supervised Learning, Multi-Task Learning, Early Stopping, Parameter Tying and Parameter Sharing, Sparse Representations, **Optimization for Training Deep Models:** How Learning Differs from Pure Optimization, Basic Algorithms. Parameter Initialization Strategies, Algorithms with Adaptive Learning Rates.

Text book 1 : Chapter 7 (7.1 to 7.10), Chapter 8 (8.1,8.3,8.4,8.5)

Module-3

	Convolution neural networks: The Convolution Operation, Motivation, Pooling, Convolution and Pooling as an Infinitely Strong Prior, Variants of the Basic Convolution Function, Structured Outputs, Data Types, Efficient Convolution Algorithms, Convolutional Networks and the History of Deep Learning. **Text book 1 : Chapter 9 (9.1 to 9.8, 9.11)**
	Module-4
	Sequence Modelling: Recurrent and Recursive Nets: Unfolding Computational Graphs, Recurrent Neural Networks, Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks. Long short-term memory. **Text book 1 : Chapter 10 (10.1 to 10.6, 10.10)**
	Module-5
	Interactive Applications of Deep Learning: Natural Language Processing: Preprocessing Natural Language Data: Tokenization, Converting All Characters to Lowercase, Removing Stop Words and Punctuation, Stemming, Handling *n*-grams, Preprocessing the Full Corpus, Creating Word Embeddings with word2vec: The Essential Theory Behind word2vec, Evaluating Word Vectors, Running word2vec, Plotting Word Vectors, The Area under the ROC Curve: The Confusion Matrix, Calculating the ROC AUC Metric, Natural Language Classification with Familiar Networks: Loading the IMDb Film Reviews, Examining the IMDb Data, Standardizing the Length of the Reviews, Dense Network, Convolutional Networks, Networks Designed for Sequential Data: Recurrent Neural Networks, Long Short-Term Memory Units, Bidirectional LSTMs, Stacked Recurrent Models, Seq2seq and Attention, Transfer Learning in NLP. **Text book 2 : Chapter-8**

Course outcomes (Course Skill Set):

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

1. Interpret the concepts of neural networks learning processes.
2. Illustrate deep learning methods using regularization and Optimization process
3. Design deep learning models using convolutional operations.
4. Analyze sequential data to build recurrent and recursive models.
5. Demonstrate the different interactive applications of deep learning.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:

Books

1. Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep Learning, MIT Press, 2016. https://www.deeplearningbook.org/lecture_slides.html
2. John Krohn, Grant Beyleveld, Aglae Bassens, Deep Learning Illustrated, A Visual, Interactive Guide to Artificial Intelligence, Pearson, 2022.

Web links and Video Lectures (e-Resources):

https://www.youtube.com/watch?v=VyWAvY2CF9c
https://www.youtube.com/watch?v=7sB052Pz0sQ
https://www.youtube.com/watch?v=Mubj_fqiAv8
https://www.coursera.org/learn/neural-networks-deep-learning

- **https://onlinecourses.nptel.ac.in/noc20_cs62/preview**

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Programming Assignments, such as implementation of CNN and Recurrent neural network models - 10 Marks
- Group assignment (Group of two) on recent developments in Deep learning – Refer IEEE/ACM/Elsevier etc publications - 15 Marks

Social Network Analysis		Semester	7
Course Code	**BAD714D**	CIE Marks	50
Teaching Hours/Week (L: T:P: S)	3:0:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Examination type (SEE)	Theory		

Course objectives:

- To introduce the fundamentals of Social Network Analysis and its significance in understanding societal connections and behaviors.
- To analyze various models of network growth and understand the properties of real-world networks.
- To explore link analysis algorithms and their applications in understanding relationships within a network.
- To study community detection methods and their relevance in identifying meaningful clusters within networks.
- To understand link prediction techniques and their application in forecasting future connections within a network.

Teaching-Learning Process (General Instructions)

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) needs not to be only traditional lecture method, can make use of digital tools to visually demonstrate key ideas that could be adopted to attain the outcomes.
2. Use think-pair-share strategies where students collaborate in pairs or groups to discuss concepts and solve small problems before sharing their understanding with the class.
3. Use real-world examples such as social media platforms or professional networks (e.g., LinkedIn) to demonstrate the concepts of Social Network Analysis.
4. Conduct practical sessions using software like Python with network libraries (e.g., NetworkX) to model and visualize network growth.
5. Use step-by-step explanations to demonstrate algorithms like PageRank and SimRank, followed by coding sessions for implementation.
6. Use network visualization tools (e.g., Gephi, Cytoscape) to help students identify and analyze communities in networks.
7. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information.
8. Demonstrate ways to solve the same problem and encourage the students to come up with their own creative solutions.

Module-1

Networks and Society - What is Social Network Analysis, why do We Study Social Networks, Applications of Social Network Analysis, Preliminaries, Three Levels of Social Network Analysis.

Network Measures - Network Basics, Node Centrality, Assortativity, Transitivity and Reciprocity, Similarity, Degeneracy.

T1 – Chapter 1 (1.1. – 1.5), Chapter 2 (2.1 – 2.6)

Module-2

Network Growth Models - Properties of Real-World Networks, Random Network Model, Ring Lattice Network Model, Watts–Strogatz Model, Preferential Attachment Model, Price's Model, Local-world Network Growth Model, Network Model with Accelerating Growth, Aging in Preferential Attachment.

T1 – Chapter 3 (3.1 – 3.9)

Module-3

Link Analysis - Applications of Link Analysis, Signed Networks, Strong and Weak Ties, Link Analysis Algorithms, PageRank, Personalised PageRank, DivRank, SimRank, PathSIM.
T1 – Chapter 4 (4.1 – 4.8)

Module-4

Community Structure in Networks - Applications of Community Detection, Types of Communities, Community Detection Methods, Disjoint Community Detection, Overlapping Community Detection, Local Community Detection, Community Detection vs Community Search, Evaluation of Community Detection Methods.
T1 – Chapter 5 (5.1 – 5.8)

Module-5

Link Prediction - Applications of Link Prediction, Temporal Changes in a Network, Problem Definition Evaluating Link Prediction Methods, Heuristic Models, Probabilistic Models, Supervised Random Walk, Information-theoretic Model, Latest Trends in Link Prediction.
T1 – Chapter 6 (6.1 – 6.9)

Course outcome (Course Skill Set)

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

1. Illustrate the core concepts of Social Network Analysis and its levels of study.
2. Demonstrate the different network growth models for real-world networks
3. Apply algorithms of PageRank and SimRank to analyze and interpret link relationships.
4. Apply community detection methods and evaluating their effectiveness in real-world scenarios.
5. Analyze heuristic, probabilistic, and supervised models to predict network link formations and changes.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:

Books

1. Tanmoy Chakraborty, "Social Network Analysis", Wiley India Pvt. Ltd., 2021

Reference Books

1. Albert-Laszlo Barabasi, "Network Science", Cambridge University Press, 2016
2. Stanley Wasserman, Katherine Faust, "Social Network Analysis: Methods and Applications", Cambridge University Press, 1994

Web links and Video Lectures (e-Resources):

- https://onlinecourses.nptel.ac.in/noc22_cs117/preview
- https://social-network-analysis.in/
- https://www.coursera.org/learn/social-network-analysis

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

Activity 1: Network Visualization and Analysis (10 Marks)

Understand network basics, measures, and visualization techniques using a real-world dataset.

Instructions:

1. Choose a small real-world dataset (e.g., social media connections, collaboration networks, or communication networks).
2. Use a network analysis tool such as Gephi, NetworkX, or Cytoscape to visualize the dataset.
3. Analyze the following:
   - Node centrality measures (degree, closeness, and betweenness).
   - Network transitivity and reciprocity.
   - Similarity or assortativity in the network.
4. Submit a report that includes the network visualization and a summary of key findings.

Assessment Criteria:

- Clarity of visualization (3 marks)
- Accuracy in calculating and interpreting network measures (5 marks)
- Quality of the report (2 marks)

Activity 2: Community Detection and Link Prediction Project (15 Marks)

Apply community detection techniques and predict future connections within a network.

Instructions:

1. Select a medium-sized dataset (e.g., email communications, citation networks, or transport networks).
2. Perform the following tasks:
   - Identify and visualize communities using two different community detection methods (e.g., Disjoint and Overlapping Community Detection).
   - Evaluate the detected communities using appropriate evaluation metrics (e.g., modularity).
   - Use a link prediction algorithm (e.g., supervised random walk or probabilistic models) to forecast future connections within the network.
3. Prepare a detailed report with visuals and findings.

Assessment Criteria:

- Accuracy and comparison of community detection methods (7 marks)
- Implementation and results of link prediction (5 marks)
- Overall presentation and report quality (3 marks)

EMBEDDED SYSTEMS		Semester	7
Course Code	**BIS714C**	CIE Marks	50
Teaching Hours/Week (L: T:P: S)	3:0:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Examination type (SEE)	Theory		

Course objectives:

- Identify the components, purpose and applications of the Embedded Systems
- Learn the RTOS and IDE for Embedded System Design
- Understand the fundamentals of ARM-based systems and basic architecture of CISC and RISC
- Familiarize with ARM programming modules along with registers, CPSR and Flags

Teaching-Learning Process (General Instructions)

These are sample strategies; which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) does not mean only the traditional lecture method, but different types of teaching methods may be adopted to achieve the outcomes.
2. Utilize video/animation films to illustrate the functioning of various concepts.
3. Promote collaborative learning (Group Learning) in the class.
4. Pose at least three HOT (Higher Order Thinking) questions in the class to stimulate critical thinking.
5. Incorporate Problem-Based Learning (PBL) to foster students' analytical skills and develop their ability to evaluate, generalize, and analyze information rather than merely recalling it.
6. Introduce topics through multiple representations.
7. Demonstrate various ways to solve the same problem and encourage students to devise their own creative solutions.
8. Discuss the real-world applications of every concept to enhance students' comprehension.
9. Use any of these methods: Chalk and board, Active Learning, Case Studies.

Module-1

Introduction to Embedded Systems: What is an Embedded System? Embedded Systems Vs General Computing Systems, History of Embedded Systems, Classification of Embedded systems, Major Application Areas of Embedded Systems. Purpose of Embedded Systems.

The Typical Embedded System: Microprocessor vs. Microcontroller, RISC vs. CISC Processors, Harvard vs. Von-Neumann Processor Architecture, Big-Endian vs. Little-Endian Processors, Memory-ROM and RAM types, Sensors & Actuators, The I/O Subsystem – I/O Devices, Light Emitting Diode (LED), 7-Segment LED Display, Optocoupler, Relay, Piezo Buzzer, Push button switch, Communication Interfaces, On-board Communication Interfaces, External Communication Interfaces.

Textbook 1: Ch. 1.1-1.6, Ch. 2.1-2.4

Module-2

Embedded System Design Concepts: Characteristics and Quality Attributes of Embedded Systems, Operational and Non-Operational Quality Attributes. Embedded Systems-Application and Domain Specific, Hardware Software Co-Design and Program Modelling.

Embedded Firmware Design and Development: Embedded Firmware Design Approaches, Embedded Firmware Development Languages, Programming in Embedded C (Excluding C language).

Textbook 1: Ch. 3.1-3.2, Ch. 4.1-4.2 (4.2.1 and 4.2.2 only), Ch. 7.1-7.2, Ch. 9.1-9.3 (9.3.1 and 9.3.2 only)

Module-3

RTOS and IDE for Embedded System Design: Operating System Basics, Types of Operating Systems, Tasks, Process and Threads (Only POSIX Threads with an example program), Thread Preemption, Preemptive Task Scheduling Techniques, Task Communication, Task Synchronization Issues – Racing and Deadlock. How to Choose an RTOS, Integration and Testing of Embedded Hardware and Firmware, The Embedded System Development Environment.

Textbook 1: Ch. 10.1-10.3, 10.5.2, 10.7, 10.8.1.1, 10.8.1.2, 10.10, Ch. 12.1-12.2, Ch. 13.1

Module-4

ARM Embedded Systems: The RISC Design Philosophy, The ARM Design Philosophy, Embedded System Hardware, Embedded System Software.

ARM Processor Fundamentals: Registers, Current Program Status Register, Pipeline, Exceptions, Interrupts, and the Vector Table, Core Extensions

Textbook 2: Ch. 1.1-1.4, Ch. 2.1-2.5

Module-5

Introduction to the ARM Instruction Set: Data Processing Instructions, Branch Instructions, Load-Store Instructions, Software Interrupt Instruction, Program Status Register Instructions, Loading Constants.

Textbook 2: Ch. 3.1-3.6

Course outcome (Course Skill Set)

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

1. Explain the characteristics and attributes of an Embedded System.
2. Illustrate the hardware software co-design and firmware design approaches of Embedded Systems.
3. Demonstrate the need of real time operating system for Embedded System applications.
4. Explain the ARM Architectural features and Instructions.
5. Develop programs using ARM instruction set for an ARM Microcontroller.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:

Text Book:

1. Shibu K V, "Introduction to Embedded Systems", Second Edition, Tata McGraw Hill Education.
2. Andrew N Sloss, Dominic Symes and Chris Wright, "ARM System Developers Guide – Designing and Optimizing System Software", Elsevier, Morgann Kaufman Publishers.

Reference Books:

1. Raj Kamal, "Embedded Systems: Architecture and Programming", Tata McGraw Hill, 2008.
2. Raghunandan.G.H, "Microcontroller (ARM) and Embedded System", Cengage learning Publication, 2019.
3. "Insider's Guide to the ARM7 based microcontrollers", Hitex Ltd.,1st edition, 2005.

Web links and Video Lectures (e-Resources):

- https://alison.com/tag/embedded-systems
- https://www.youtube.com/watch?v=uFhDGagZzjs
- https://www.youtube.com/watch?v=G1c_WMD_5pU
- https://archive.nptel.ac.in/courses/106/105/106105193/

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Demonstrate the Installation and working of Keil Software - Student group of TWO (10 Marks).
- Using Keil software, observe the various Registers, Dump, CPSR etc. and write Assembly Language Programs (15 Marks).

	Animation Principles and Design		Semester	7
	Course Code	**BCG714D**	CIE Marks	50
	Teaching Hours/Week (L: T:P: S)	3:0:0:0	SEE Marks	50
	Total Hours of Pedagogy	40	Total Marks	100
	Credits	03	Exam Hours	03
	Examination type (SEE)	Theory		

Course objectives:

- Introduce basic concepts of animation.
- Learn interpolation-based animation principles.
- Understand physical and behavioural animation techniques.

Teaching-Learning Process (General Instructions)

These are sample Strategies; that teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) needs not to be only traditional lecture method, can make use of digital tools to visually demonstrate key ideas that could be adopted to attain the outcomes.
2. Use think-pair-share strategies where students collaborate in pairs or groups to discuss concepts and solve small problems before sharing their understanding with the class.
3. Use case studies that apply machine learning in fields like finance, healthcare, and marketing to reinforce practical applications.
4. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information.
5. Utilize tools to visually demonstrate the impact of different concepts and methods of animation.
6. Demonstrate ways to solve the same problem and encourage the students to develop their own creative solutions.

Module-1

Introduction: Motion perception,The heritage of animation, Animation production, Computer animation production.
Technical Background: Spaces and transformations, Orientation representation.
Interpolating Values: Interpolation, Controlling the motion of a point along a curve, Interpolation of orientations, Working with paths.

Chapter-1 (1.1.-1.4), Chapter-2 (2.1, 2.2), Chapter-3 (3.1-3.4)

Module-2

Interpolation-Based Animation: Key-frame systems, Animation languages, Deforming objects, Three-dimensional shape interpolation, Morphing (two-dimensional).
Kinematic Linkages: Hierarchical modeling, Forward kinematics, Inverse kinematics.

Chapter-4 (4.1-4.5), Chapter-5 (5.1-5.3)

Module-3

Motion Capture: Motion capture technologies, Processing the images, Camera calibration, Three-dimensional position reconstruction, Fitting to the skeleton, Output from motion capture systems, Manipulating motion capture data.
Physically Based Animation: Basic physics—a review, Spring animation examples, Particle systems, Rigid body simulation, Cloth, Enforcing soft and hard constraints.

Chapter-6 (6.1-6.7), Chapter-7 (7.1-7.6)

Module-4

	Fluids - Liquids and Gases: Specific fluid models, Computational fluid dynamics. **Modeling and Animating Human Figures:** Overview of virtual human representation, Reaching and grasping, Walking, Coverings. **Facial Animation:** The human face, Facial models, Animating the face, Lip-sync animation. Chapter-8 (8.1,8.2), Chapter-9 (9.1-9.4), Chapter-10 (10.1-10.4)
	Module-5
	Behavioral Animation: Primitive behaviors, Knowledge of the environment, Modeling intelligent behavior, Crowd. **Special Models for Animation:** Implicit surfaces, Plants. Chapter-11 (11.1-11.4), Chapter-12 (12.1-12.3)

Course outcome (Course Skill Set)

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

1. Explain basic concepts of animation with examples.
2. Illustrate interpolation based animation techniques.
3. Demonstrate physical based animation with motion capture.
4. Apply algorithms to animate human figures and face.
5. Apply behavioural and special models for Animation.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:

Textbook

1. Rick Parent, Computer Animation: Algorithms and Techniques, Third Edition, 2012 Elsevier.

Reference Books:

2. Richard Williams, The Animator's Survival Kit, Expanded Edition: A Manual of Methods, Principles and Formulas for Classical, Computer, Games, Stop Motion and Internet Animators, Faber and Faber (Expanded Edition), 2009.
3. John M. Blain, The Complete Guide to Blender Graphics: Computer Modeling and Animation, 6th Edition, A K Peters/CRC Press, 2020.
4. Stuart Mealing, The Art and Science of Computer Animation, 2nd Edition , Intellect (UK), 1998.

Web links and Video Lectures (e-Resources):

- https://onlinecourses.swayam2.ac.in/cec20_cs08/
- https://nptel.ac.in/courses/106102065
- https://www.adobe.com/uk/creativecloud/animation/discover/computer-animation.html
- https://www.geeksforgeeks.org/computer-animation/

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Course project (group of two) using open source tools like Blender, Synfig, Animaker, pencil2d or any other relevant tool (25 marks). The projects must cover the following major concepts:
    1. Motion Capture
    2. Spring animation, Particle systems
    3. Rigid body simulation, Cloth
    4. Reaching and grasping, Walking, Coverings.
    5. Animating the face, Lip-sync animation
    6. Crowd, surfaces and plants

Introduction to DBMS		Semester	7
Course Code	**BCS755A**	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	3
Examination nature (SEE)	Theory		

Course objectives:

- To Provide a strong foundation in database concepts, technology, and practice.
- To Practice SQL programming through a variety of database problems.
- To Understand the relational database design principles.
- To Demonstrate the use of concurrency in database.
- To Design and build database applications for real world problems.

Teaching-Learning Process (General Instructions)

These are sample Strategies; that teachers can use to accelerate the attainment of the various course outcomes.

- Lecturer method (L) needs not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
- Use of Video/Animation to explain functioning of various concepts.
- Encourage collaborative (Group Learning) Learning in the class.
- Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
- Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.
- Use any of these methods: Chalk and board, Active Learning, Case Studies.

MODULE-1

Introduction to Databases: Introduction, Characteristics of database approach, Advantages of using the DBMS approach, History of database applications.

Overview of Database Languages and Architectures: Data Models, Schemas, and Instances. Three schema architecture and data independence, database languages, and interfaces, The Database System environment.

Textbook 1:Ch 1.1 to 1.8, 2.1 to 2.6

MODULE-2

Conceptual Data Modeling using Entities and Relationships: Entity types, Entity sets and structural constraints, Weak entity types, ER diagrams, Specialization and Generalization.

Mapping Conceptual Design into a Logical Design: Relational Database Design using ER-to-Relational mapping

Textbook 1: Ch 3.1 to 3.10, 9.1 & 9.2

MODULE-3

Relational Model: Relational Model Concepts, Relational Model Constraints and relational database schemas, Update operations, transactions, and dealing with constraint violations.

Relational Algebra: Unary and Binary relational operations, additional relational operations (aggregate, grouping, etc.) Examples of Queries in relational algebra.

Textbook 1: Ch 5.1 to 5.3, Ch 8.1 to 8.5

MODULE-4

SQL: SQL data definition and data types, Schema change statements in SQL, specifying constraints in SQL, retrieval queries in SQL, INSERT, DELETE, and UPDATE statements in SQL, Additional features of SQL

Normalization: Database Design Theory – Introduction to Normalization using Functional and Multivalued Dependencies: Informal design guidelines for relation schema, Functional Dependencies, Normal Forms based on Primary Keys, Second and Third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form.

Textbook 1: Ch 6.1 to 6.5,14.1 to 14.7

MODULE-5

SQL: Advanced Queries: More complex SQL retrieval queries, Specifying constraints as assertions and action triggers, Views in SQL.

Concurrency Control in Databases: Two-phase locking techniques for Concurrency control, Concurrency control based on Timestamp ordering, Multiversion Concurrency control techniques, Validation Concurrency control techniques, Granularity of Data items and Multiple Granularity Locking.

Textbook 1: Ch 7.1 to 7.3, 21.1 to 21.5

Course outcomes (Course Skill Set):
At the end of the course, the student will be able to:

- Demonstrate the basic elements of a database management system.
- Design entity relationship and convert entity relationship diagrams into RDBMS.
- Use Structured Query Language (SQL) for database manipulation.
- Apply normalization to increase the efficiency of database design.
- Illustrate the concepts of concurrency control techniques.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.

The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered

Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.

For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

- Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (duration 03 hours).
- The question paper will have ten questions. Each question is set for 20 marks.
- There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), should have a mix of topics under that module.
- The students have to answer 5 full questions, selecting one full question from each module.
- Marks scored shall be proportionally reduced to 50 marks.

Suggested Learning Resources:

Text Books:

1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.

Reference Books:

1. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

Course Project (25 marks)

- For any problem selected
    - Develop the application having at least five tables & domain areas shall include health care, agriculture & so on.

	Introduction to Algorithms		Semester	7
	Course Code	BCS755B	CIE Marks	50
	Teaching Hours/Week (L: T:P: S)	3:0:0:0	SEE Marks	50
	Total Hours of Pedagogy	40	Total Marks	100
	Credits	03	Exam Hours	03
	Examination type (SEE)	Theory		

Course objectives:

- To learn the methods for analyzing algorithms and evaluating their performance.
- To demonstrate the efficiency of algorithms using asymptotic notations.
- To solve problems using various algorithm design methods, including brute force, greedy, divide and conquer, decrease and conquer, transform and conquer, dynamic programming, backtracking, and branch and bound.
- To learn the concepts of P and NP complexity classes.

Teaching-Learning Process (General Instructions)

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) does not mean only the traditional lecture method, but different types of teaching methods may be adopted to achieve the outcomes.
2. Utilize video/animation films to illustrate the functioning of various concepts.
3. Promote collaborative learning (Group Learning) in the class.
4. Pose at least three HOT (Higher Order Thinking) questions in the class to stimulate critical thinking.
5. Incorporate Problem-Based Learning (PBL) to foster students' analytical skills and develop their ability to evaluate, generalize, and analyze information rather than merely recalling it.
6. Introduce topics through multiple representations.
7. Demonstrate various ways to solve the same problem and encourage students to devise their own creative solutions.
8. Discuss the real-world applications of every concept to enhance students' comprehension.

Module-1

INTRODUCTION: What is an Algorithm?, Fundamentals of Algorithmic Problem Solving, Important problem Types, Fundamental Data Structures, Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, ,Analysis Framework, Asymptotic Notations and Basic Efficiency Classes,

Chapter 1 (Sections 1.1 to 1.4), Chapter 2 (2.1, 2.2)

Module-2

FUNDAMENTALS OF THE ANALYSIS OF ALGORITHM EFFICIENCY: Mathematical Analysis of Non-recursive Algorithms, Mathematical Analysis of Recursive Algorithms.

BRUTE FORCE APPROACHES: Selection Sort and Bubble Sort, Sequential Search and Brute Force String Matching.

Chapter 2(Sections 2.3,2.4), Chapter 3(Section 3.1,3.2)

	Module-3
	Exhaustive Search (Travelling Salesman problem and Knapsack Problem). **Depth First search and Breadth First search.** **DECREASE-AND-CONQUER:** Insertion Sort, Topological Sorting. **DIVIDE AND CONQUER:** Merge Sort, Binary Tree Traversals. **Chapter 3(3.4,3.5), Chapter 4 (Sections 4.1,4.2), Chapter 5 (Section 5.1,5.3)**
	Module-4
	TRANSFORM-AND-CONQUER: Balanced Search Trees (AVL Trees), Heaps and Heapsort. **SPACE-TIME TRADEOFFS:** Sorting by Counting: Comparison counting sort, Input Enhancement in String Matching: Horspool's Algorithm, Hashing. **Chapter 6 (Sections 6.3,6.4), Chapter 7 (Sections 7.1,7.2, 7.3)**
	Module-5
	DYNAMIC PROGRAMMING: Three basic examples, The Knapsack Problem and Memory Functions. **THE GREEDY METHOD:** Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees and Codes. **Chapter 8 (Sections 8.1,8.2), Chapter 9 (Sections 9.2,9.3,9.4)**

Course outcome (Course Skill Set)

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

1. Explain the algorithm design steps and computational problem types.
2. Apply the asymptotic notational method to analyze the performance of the algorithms in terms of time complexity.
3. Demonstrate divide & conquer approaches and decrease & conquer approaches to solve computational problems.
4. Make use of the transform & conquer design approach to solve the given real-world or complex computational problems.
5. Apply greedy and dynamic programming methods to solve graph & string-based computational problems.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:

Textbooks

1. Introduction to the Design and Analysis of Algorithms, By Anany Levitin, 3rd Edition (Indian), 2017, Pearson.

Reference books

1. Computer Algorithms/C++, Ellis Horowitz, SatrajSahni and Rajasekaran, 2nd Edition, 2014, Universities Press.
2. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI.
3. Design and Analysis of Algorithms, S. Sridhar, Oxford (Higher Education)

Web links and Video Lectures (e-Resources):

- Design and Analysis of Algorithms: https://nptel.ac.in/courses/106/101/106101060/

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

1. Problem Solving - Competitive programming (Hacker Rank/ Hacker Earth / Leetcode) – 10 Marks
2. Problem solving (Numerical examples) related to different algorithms – 15 Marks

	SOFTWARE ENGINEERING		Semester	7
	Course Code	**BCS755C**	CIE Marks	50
	Teaching Hours/Week (L:T:P: S)	3:0:0:0	SEE Marks	50
	Total Hours of Pedagogy	50	Total Marks	100
	Credits	04	Exam Hours	3
	Examination type (SEE)	Theory		

Course objectives:

To understand foundational principles and the evolving nature of software engineering.
- To learn various software process models and their practical applications.
- To acquire skills in gathering, modeling, and validating software requirements.
- To apply Agile methodologies and understand core software engineering practices.
- To build a foundation for software design, testing, and quality assurance.

Teaching-Learning Process

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) needs not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding
9. Use any of these methods: Chalk and board, Active Learning, Case Studies

Module-1

Software and Software Engineering: The nature of Software, The unique nature of WebApps, Software Engineering, The software Process, Software Engineering Practice, Software Myths.

Process Models: A generic process model, Process assessment and improvement, Prescriptive process models: Waterfall model, Incremental process models, Evolutionary process models, Concurrent models, Specialized process models. Unified Process , Personal and Team process models

Textbook 1: Chapter 1: 1.1 to 1.6, Chapter 2: 2.1 to 2.5

Module-2

	Understanding Requirements: Requirements Engineering, Establishing the ground work, Eliciting Requirements, Developing use cases, Building the requirements model, Negotiating Requirements, Validating Requirements. **Requirements Modeling Scenarios, Information and Analysis classes**: Requirement Analysis, Scenario based modeling, UML models that supplement the Use Case, Data modeling Concepts, Class-Based Modeling. Requirement Modeling Strategies : Flow oriented Modeling , Behavioral Modeling. **Textbook 1: Chapter 5: 5.1 to 5.7, Chapter 6: 6.1 to 6.5, Chapter 7: 7.1 to 7.3**
	Module-3
	Agile Development: What is Agility?, Agility and the cost of change. What is an agile Process?, Extreme Programming (XP), Other Agile Process Models, A tool set for Agile process . **Principles that guide practice**: Software Engineering Knowledge, Core principles, Principles that guide each framework activity. **Textbook 1: Chapter 3: 3.1 to 3.6, Chapter 4: 4.1 to 4.3**
	Module-4
	Software Design: Design within the context of software engineering, Design process and quality, Design concepts: abstraction, modularity, architecture, patterns. **Architectural Design**: Architectural styles and patterns, reference architectures, component-level design, designing class-based components, conducting component-level design, design for reuse. **Textbook 1:Chapter 8: 8.1–8.6, Chapter 9: 9.1–9.5**
	Module-5
	Software Testing: Introduction to software testing, Strategic approach, Test strategies for conventional and object-oriented software, Validation testing, System testing, White-box and Black-box testing, Basis Path Testing, Control structure testing. **Software Quality:** Concepts of quality, Software quality assurance, Reviews, Software reliability and metrics. **Textbook 1: Chapter 14: Sections 14.1 to 14.5,Chapter 15: Sections 15.1 to 15.5, Chapter 19: Sections 19.1 to 19.5**

Course outcome

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

1. Explain the software nature, engineering practices, myths, and software process models.
2. Apply requirements engineering, elicitation, modeling, and validation in software development.
3. Demonstrate agile principles, practices, and tools for software development agility.
4. Apply design concepts, process, and architecture for quality software development.
5. Explain software testing strategies and quality assurance for reliable software.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:

Textbook

Roger S. Pressman: Software Engineering – A Practitioner's Approach, 7th Edition, Tata McGraw Hill, 2010.

Web links and Video Lectures (e-Resources):

https://www.geeksforgeeks.org/software-engineering/software-engineering/

Activity-Based Learning (Suggested Activities in Class)/Practical-Based learning

- Course project (Group of two students): Simulation that covers all the phases of SDLC - 25 marks

	Computer Graphics		Semester	7
	Course Code	BCG755D	CIE Marks	50
	Teaching Hours/Week (L: T:P: S)	3:0:0:0	SEE Marks	50
	Total Hours of Pedagogy	40	Total Marks	100
	Credits	03	Exam Hours	03
	Examination type (SEE)	Theory		

Course objectives:

- To understand hardware, software, and OpenGL Graphics Primitives.
- To Illustrate an interactive computer graphic using the OpenGL.
- To implement the algorithms for 2D graphics Primitives and attributes.
- To demonstrate Geometric transformations, viewing on both 2D and 3D objects.
- To Infer the representation of curves, and surfaces.

Teaching-Learning Process (General Instructions)

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) does not mean only the traditional lecture method, but different types of teaching methods may be adopted to achieve the outcomes.
2. Utilize video/animation films to illustrate the functioning of various concepts.
3. Promote collaborative learning (Group Learning) in the class.
4. Pose at least three HOT (Higher Order Thinking) questions in the class to stimulate critical thinking.
5. Incorporate Problem-Based Learning (PBL) to foster students' analytical skills and develop their ability to evaluate, generalize, and analyze information rather than merely recalling it.
6. Introduce topics through multiple representations.
7. Demonstrate various ways to solve the same problem and encourage students to devise their own creative solutions.
8. Discuss the real-world applications of every concept to enhance students' comprehension.

Module-1

Computer Graphics and OpenGL:

Computer Graphics: Basics of computer graphics, Application of Computer Graphics, Video Display Devices: Random Scan and Raster Scan displays, graphics software.

OpenGL: Introduction to OpenGL, coordinate reference frames, specifying two-dimensional world coordinate reference frames in OpenGL, OpenGL point functions, OpenGL line functions, point attributes, line attributes, curve attributes, OpenGL point attribute functions, OpenGL line attribute functions, Line drawing algorithms(DDA, Bresenham's), circle generation algorithms (Bresenham's).

Chapter 1, Chapter 2 (2.1, 2.2), Chapter 3, Chapter 4 (4.1-4.4), Chapter (5.4-5.8), Chapter 6 (6.1)

Module-2

Fill area Primitives, 2D Geometric Transformations and 2D viewing:

	Fill area Primitives: Polygon fill-areas, OpenGL polygon fill area functions, fill area attributes, general scan line polygon fill algorithm, OpenGL fill-area attribute functions. 2D Geometric Transformations: Basic 2D Geometric Transformations, matrix representations and homogeneous coordinates. Inverse transformations, 2DComposite transformations, other 2D transformations, raster methods for geometric transformations, OpenGL raster transformations, OpenGL geometric transformations function, 2D viewing: 2D viewing pipeline, OpenGL 2D viewing functions. **Chapter 4 (4.6-4.8), Chapter 5 (5.9, 5.10), Chapter 6 (6.10), Chapter 7 (7.1-7.7, 7.9), Chapter 8 (8.1,8.4)**
	Module-3
	Clipping, 3D Geometric Transformations: Clipping: clipping window, normalization and viewport transformations, clipping algorithms,2D point clipping, 2D line clipping algorithms: cohen-sutherland line clipping only, polygon fill area clipping: Sutherland-Hodgeman polygon clipping algorithm only. 3D Geometric Transformations: 3D translation, rotation, scaling, composite 3D transformations, OpenGL geometric transformations functions. **Chapter 8 (8.2, 8.3, 8.5-8.8), Chapter 9 (9.1-9.4, 9.8)**
	Module-4
	3D Viewing and Visible Surface Detection: 3D Viewing: 3D viewing concepts, 3D viewing 10 pipeline, 3D viewing coordinate parameters, Transformation from world to viewing coordinates, Projection transformation, orthogonal projections, perspective projections, The viewport transformation and 3D screen coordinates. OpenGL 3D viewing functions. Visible Surface Detection Methods: Classification of visible surface Detection algorithms, depth buffer method only and OpenGL visibility detection functions. **Chapter 10 (10.1-10.10), Chapter 16 (16.1, 16.3, 16.14)**
	Module-5
	Curves and surfaces: Curved surfaces, quadric surfaces, super-quadratics, OpenGL Quadric-Surface and Cubic-Surface Functions, Bezier Spline Curves, Bezier surfaces, B-Spline curves, B-Spline surfaces, OpenGL curve functions. Corresponding OpenGL functions. **Chapter 13 (13.3-13.6,), Chapter 14 (14.8-14.11, 14.16)**
Course outcome (Course Skill Set) At the end of the course, the student will be able to: 1. Demonstrate computer graphics primitives with OpenGL. 2. Illustrate 2-D geometric transformations and viewing. 3. Implement clipping algorithms and, 3D Geometric Transformations. 4. Demonstrate 3 D viewing and visible surface detection methods. 5. Implement curves and surfaces with OpenGL functions.	

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.
- The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.
- For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (**duration 03 hours).**

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:

Textbooks

1. Computer Graphics with OpenGL, Donald Hearn, Pauline Baker and Warren Carithers, 4th Edition, 2011, Pearson.

Reference books

1. Interactive Computer Graphics- A Top Down approach with OpenGL, Edward Ange, Pearson Education, 5th edition, 2009.

Web links and Video Lectures (e-Resources):

- Computer Graphics: https://onlinecourses.nptel.ac.in/noc20_cs90/preview

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

Conduct experiments individually and submit the record of conduction – 25 marks

1. Implement Brenham's line drawing algorithm for all types of slope.
2. Clip a lines using Cohen-Sutherland algorithm
3. Clip a polygon using Sutherland-Hodgeman algorithm
4. Develop a program to animate a flag using Bezier Curve algorithm
5. Develop a program to fill the polygon using scan line algorithm