GOVERNMENT OF ASSAM

STATE COUNCIL FOR TECHNICAL EDUCATION, ASSAM

Director of Technical Education, Assam.

DRAFT SYLLABUS OF 4th SEMESTER

Computer Engineering

Inclusion of National Education Policy, 2020:

As envisioned in the NEP 2020 the State Council for Technical Education, Assam will be responsible for mentoring its affiliated Polytechnics so that they develop capabilities and achieve minimum benchmarks in academic and curricular matters, teaching and assessment.
State Council for Technical Education (SCTE), Assam will notify the pool of MOOCs courses duly approved by its statutory bodies and map them in Academic Bank of Credits (ABC). Affiliated Polytechnics may opt for any of these courses as required. State Council for Technical Education (SCTE) will take suitable action for preparation of syllabus for these courses including learning objective and learning outcome.
The State Council for Technical Education (SCTE) may fix the minimum number of credits to be earned by the students from a parent institution.
All the affiliating institutions will complete Academic Bank of Credits registration through SCTE. Institute will monitor the opening of ABC account by all the students.

Salient features

Salient features that are to be considered for developing the curriculum aligned to NEP 2020 are as follows:

- Reduced number of credits.
- Introduction of Student Induction Program.
- Well defined learning objectives & outcomes for each course.
- Inclusion of courses on socially relevant topics.
- Built-in flexibility to the students in terms of professional elective and open Elective courses.
- Mandatory internship to equip the students with practical knowledge and provide them exposure to real time industrial environments.
- Virtual Labs.
- Mapping of Courses to its equivalent NPTEL/SWAYAM Course.
- Course on 'Entrepreneurship and Startups' to encourage entrepreneurial mindset.
- Introduction of Design Thinking and Universal Human Value course.

Basic Guidelines:

The existing Credit System is revised as Choice Based Credit System (CBCS) in line with NEP guidelines, to infuse innovation and flexibility. No hard separation between streams, between curricular and extra-curricular, between Vocational and Academic, Multidisciplinary and holistic education across the disciplines, Ethics and Human & Constitutional values, Life skills, use of technology as part of all curriculum.

- An academic year is divided into two semesters as per AICTE guidelines.
- A semester consists of approximately 90 working days. One working week will have approximately 40 hours of instructional time.
- There shall also be a Winter Internship Program for duration of 4 weeks/one month from 1st January to -31st January.
- Summer term courses may be offered on a fast-track mode to enable students to complete arrears/special courses.
- Teachers may avail semester end vacations after the end of each semester. Subject to completion all kinds of examination related work.
- The Polytechnics under the Directorate of Technical Education can decide on the kind of courses to be offered in the summer term, based on the requirements and also based on the availability of teaching faculty.
- Internship/apprenticeship can be carried out during the winter term, mandatorily for regular student.
- Students who wish to exit after 1st year or 2nd year of study, have to undergo mandatory

bridge courses as defined in the course structure.

Induction Program:

The Essence and Details of Induction program can be find from the 'Detailed Guide on Student Induction program', as available on AICTE Portal,
(Link: https://www.aicteindia.org/sites/default/files/Detailed%20Guide%20on%20Student%20Induction%20program.pdf),
The Induction program for students to be offered right at the start of the first year for three-week duration.

- Physical activity
- Creative Arts
- Universal Human Values
- Literary
- Proficiency Modules
- Lectures by Eminent People
- Visits to local Areas

Mandatory Visits/ Workshop/Expert Lectures:

a) It is mandatory to arrange one industrial visit every semester for the students of each branch.
b) It is mandatory to conduct a One-week workshop during the winter break after fifth semester on professional/ industry/ entrepreneurial orientation.
c) It is mandatory to organize at least one expert lecture per semester for each branch by inviting resource persons from domain specific industry.

GENERAL COURSE STRUCTURE & CREDIT DISTRIBUTION

Definition of Credit:

1 Hr. Lecture (L) per week	1 credit
1 Hr. Tutorial (T) per week	1 credit
1 Hr. Practical (P) per week	0.5 credit
2 Hours Practical (P) per week	1 credit

A. Range of Credits:

In the light of the fact that a typical Model Four-year Under Graduate degree program in Engineering has about 160 credits, the total number of credits proposed for the three-year Diploma program in Engineering & Technology is 120.

B. Structure of Diploma Engineering program:

The structure of Diploma Engineering program shall have essentially the following categories of courses with the breakup of credits as given:

Sr. No.	Category	Suggested Breakup of Credits
1.	Humanities & Social Sciences courses	8*
2.	Basic Science courses	19*
3.	Engineering Science courses	15*
4.	Program Core courses (Branch specific)	45*
5.	Program Elective courses (Branch specific)	12*
6.	Open Elective courses (from other technical and /or emerging subjects)	9*
7.	Project work, seminar and internship in industry or elsewhere	12*
8.	Audit Courses [Environmental Sciences, Induction training, Indian Constitution, Essence of Indian Traditional Knowledge etc.]	(non-credit)
	Total	**120***

**Minor variation is allowed as per need of the respective disciplines.*

C. Course code and definition:

Course code	Definitions
L	Lecture
T	Tutorial
P	Practical
HS	Humanities & Social Sciences Courses
BS	Basic Science Courses
ES	Engineering Science Courses
PC	Program Core Courses
PE	Program Elective Courses
OE	Open Elective Courses
AU	Audit Courses
SI	Summer Internship
PR	Project
SE	Seminar

D. Course level coding scheme:

Three-digit number (odd numbers are for the odd semester courses and even numbers are for even semester courses) used as suffix with the Course Code for identifying the level of the course e.g.

101, 102 … etc. for first year
201, 202 …. Etc. for second
year 301, 302 … for third year

E. Evaluation Scheme for 4th Semester of Diploma in Engineering courses under State Council for Technical Education (SCTE), Assam

For Theory Courses:

(The weightage of Internal assessment is 40% and for End Semester Exam is 60%) The student has to obtain at least 40% marks individually both in internal assessment and end semester exams to pass.

For Practical Courses:

(The weightage of Internal assessment is 60% and for End Semester Exam is 40%) The student has to obtain at least 40% marks individually both in internal assessment and end semester exams to pass.

For Summer Internship / Projects / Seminar etc.

Evaluation is based on work done, quality of report, performance in viva-voce, presentation etc.

Note: The internal assessment is based on the student's performance in mid semester tests (two best out of three), quizzes, assignments, class performance, attendance, viva-voce in practical, lab record etc.

F. Mapping of Marks to Grades

Each course (Theory/Practical) is to be assigned 100 marks, irrespective of the number of credits, and the mapping of marks to grades may be done as per the following table:

Range of Marks	Assigned Grade	Grade Point
91-100	A^+	10
81-90	A	9
71-80	B^+	8
61-70	B	7
51-60	C^+	6
46-50	C	5
40-45	D	4
< 40	F (Fail due to less marks)	-
-	F^R (Fail due to shortage of attendance and therefore, to repeat the course)	-

A. Credit Distribution, Mark Distribution and Break up of Internal Assessment Marks as per following table:

A. Semester wise Credit point distribution

SL. NO	SEMESTER	CREDIT POINTS
1	THIRD	20
2	FOURTH	20
3	FIFTH	20
4	SIX	20
TOTAL CREDIT POINTS		80

1.1Subject Category wise Credit point Distribution

COURSE CATEGORY	SEM III	SEM IV	SEM V	SEM VI	TOTAL CREDIT POINT
Programme Core Courses	18	12	10	8	48
Programme Elective Courses	0	2	6	0	08
Open Elective Courses	2	2	2	2	08
Audit Courses	0	0	0	0	00
Semester Internship	0	2	0	2	04
PROJECT and Seminar	0	2	2	5	09
HSC	0	0	0	3	03
TOTAL CREDIT	20	20	20	20	80

Course Structure:

FOURTH SEMESTER						
SL.NO	COURSE CODE	COURSE TITLE	L	T	P	C
1	COPC-401	Operating System	2	0	0	2
2	COPC-402	Computer Architecture and Organization	2	0	0	2
3	COPC-403	Data Structure	3	0	0	3
4	COPC-404	Digital Electronics	2	1	0	3
5	COPC-405	Operating System Lab	0	0	2	1
6	COPC-406	Data Structure Lab	0	0	2	1
7	COPE-401	Multimedia Lab	0	1	3	2
8	COPE-402	Scripting Language (Python) Lab	0	1	3	2
9	OE-401	Visual Programming Lab	0	1	3	2
10	OE-402	Fundamentals of Data Analysis and Cloud	2	0	0	2
11	SI-01	Internship	0	0	4	2
12	PR-01	Minor Project	0	0	4	2
13	SE-01	Seminar/Guest Lecture/remedial	0	0	4	0

***NOTE: - 1. Choose any one subject code COPE-401 or COPE-402**

2. Choose any one subject code OE-401 or OE-402

Marks Distribution (4th Semester):

Sl. No.	Course Code	Course Title	Internal Assessment (Theory)		ESE		Internal Assessment (Practical)		Practical Test		Total Marks (Course)	Pass Marks
			Total Marks	Pass Marks	Total Marks	Pass Marks	Total Marks	Pass Marks	Total Marks	Pass Marks		
1	COPC-401	Operating System	40	16	60	24					100	40
2	COPC-402	Computer Architecture and Organization	40	16	60	24					100	40
3	COPC-403	Data Structure	40	16	60	24					100	40
4	COPC-404	Digital Electronics	40	16	60	24					100	40
5	COPC-405	Operating System Lab					60	24	40	16	100	40
6	COPC-406	Data Structure Lab					60	24	40	16	100	40
7	COPE-401	Multimedia Lab					60	24	40	16	100	40
8	COPE-402	Scripting Language (Python) Lab					60	24	40	16	100	40
9	OE-401	Visual Programming Lab					60	24	40	16	100	40
10	OE-402	Fundamentals of Data Analysis and Cloud Computing.(IS)	40	16	60	24					100	40
11	SI-01	Internship									100	40
12	PR-01	Minor Project									100	40
13	SE-01	Seminar/Guest Lecture/remedial										
										Total	1000	

***NOTE : -**

1. Choose any one subject code COPE-401 or COPE-402

2. Choose any one subject code OE-401 or OE-402

Break Up of Internal Assessment Marks
Internal Assessment for Theory
(TA+HA&CT)

Component	**Teacher's Assessment (TA)**			**Home Assignment & Class Test (HA&CT)**				**Total**
	Attendance	Discipline	Class Participation	Home Assignment	Presentation / Seminar	Quiz	Class Test	
Maximum Marks	5	1	2	4	6	2	20	**40**
Note: Three (3) class tests must be conducted for each course. For calculation of Internal Assessment, best two (2) class tests out of the three (3) conducted is to be considered.								

Internal Assessment for Practical (PA)

For 60 marks

Component	**Maximum Marks**
Teacher's Assessment (TA)	
Attendance	10
Discipline	2
Practical Assignment	
Practical Work and/or Laboratory Report	30
Presentation	10
Viva	8
Total	**60**

a) **Grading System proposed is Absolute Grading System.**
b) **Conversion factor from Cumulative Grade Point Average (CGPA) to Percentage (%) is 10.**

Range of Attendance with Marks (for Theory):

Sl No.	Range of Attendance	Marks
1	96% and above	5 Mark
2	91% to 95%	4 Mark
3	86% to 90%	3 Mark
4	81% to 85%	2 Mark
5	76% to 80%	1 Mark
6	75%	Only permitted to appear in the exam
7	Between 60% to 75%	NC: May appear in the exam if there are genuine reasons
8	Below 60%	DC: Cannot appear in the exam

Range of Attendance with Marks (for Practical):

Sl No.	Range of Attendance	Marks
1	96% and above	10 Mark
2	91% to 95%	9 Mark
3	86% to 90%	8 Mark
4	81% to 85%	7 Mark
5	76% to 80%	5 Mark
6	75%	Only permitted to appear in the exam
7	Between 60% to 75%	NC: May appear in the exam if there are genuine reasons
8	Below 60%	DC: Cannot appear in the exam

Valid Ground for Non-collegiate students:

1. Medical grounds.
2. Participation in a state/national level competition, including the journey period as approved by the Head of Institution.
3. Participation in a seminar of national/international level, workshop or conference, including the journey period as approved by the Head of Institution.
4. Participation in voluntary programmes conducted by the Institute/Directorate as per Government notification.
5. Any other grounds which the Examination Committee, in consultation with the Secretary, SCTE may approve.

Current Examination Fee structure:

- Non Collegiate Fees: Rs. 300/- (per subject)
- Examination Fee: Rs. 750/-
- Retest Fee: Rs. 300/- (per subject)

N.B: The rates may be changed subject to Government order.

Conditions for Dis-Collegiate (DC) student:

(i) Any candidate who is an internal student but has attended less than 60% of classes in any one or more subjects on theory and/ or practical shall not be eligible to appear in the respective semester examination and termed as Dis-collegiate student.

(ii) Dis-collegiate students will have to repeat the semester by taking admission in the next session.

Note: The above guidelines may be changed/modified as per AICTE and Government of Assam notification from time to time.

Detailed Fourth Semester Curriculum Contents (Computer Engineering Branch)

Course Title – Operating System

Semester : 4th

Course Code: COPC-401

Credits : 2 (L : 2 T : 0 P : 0)

Course Objectives :

- To develop understanding of fundamental concepts, architecture, and functional components of modern operating systems.
- To provide knowledge of processes, threads, CPU scheduling, synchronization, and deadlock handling in operating systems.
- To impart understanding of memory management techniques, paging, segmentation, and virtual memory organization.
- To enable students to understand file systems, disk management, and I/O mechanisms for efficient data storage and retrieval.

Course Outcomes :

After completing this course, students will be able to:

CO1. Students will be able to explain the purpose, functions, and types of operating systems and their evolution over time.

CO2. Students will be able to analyze and illustrate concepts of processes, threads, IPC, and apply CPU scheduling algorithms.

CO3. Students will be able to differentiate and apply various memory management techniques and page replacement strategies.

CO4. Students will be able to solve synchronization problems using semaphores/monitors and evaluate deadlock conditions and avoidance techniques.

CO5. Students will be able to describe file system structures, directory management, disk scheduling, and I/O mechanisms in operating systems.

Teaching Scheme:

	Teaching Scheme		
L	T	P	Total Hours per week
2	0	0	2

Examination Scheme:

	Internal Assessment (Theory)	ESE
Full Marks	40	60
Pass Marks	16	24

Detailed Course Content:

Unit	Topic/ Sub-topics	Intended Learning Outcomes	Hours
1	**Introduction to the operating system :** Overview of operating system, basic concepts, Functions, History, types of OS	1. Definition and goals of operating systems 2. Discuss various functions of operating system. 3. History and evolution of operating systems 4. Types of operating systems (Batch, Multiprogramming, Time-sharing, Distributed, Real-time)	2
2	**Process Management:** Process concepts, threads, Process states and transitions, operations on processes, IPC, Process Scheduling, Scheduling Algorithms	1. Define process, thread,Scheduling. 2. Discuss various features of processes including IPC.	6

		3. Describe the differences among short term, medium term, and long term scheduling. List out the scheduling criteria. 4. Explain various CPU scheduling algorithm viz. FCFS, SJF, Round Robin, Priority, and Multiple queue taking suitable example . 5. Differentiate between preemptive and non-preemptive scheduling.	
3	**Memory Management:** Garbage Collection and Compaction, Page Memory Management, Page replacement Algorithm, Swapping, Segmentation, Segmented page memory management, Demand page memory management, Virtual Memory	1. Define virtual memory, page fault, thrashing, and swapping. 2. Discuss various memory management techniques, including segmentation and paging. 3. Differentiate between (i) Internal and external fragmentation, (ii) Physical address and logical address, (iii) Paging and segmentation. 4. Explain the concept of demand paging, page replacement algorithms and allocation of frames.	5
4	**Synchronization and Deadlock:** Process synchronization, critical section problem, semaphores, classic problems of synchronization, deadlock characterization, deadlock prevention, detection and	1. Define Synchronization 2. critical section problem, 3. semaphore and monitor. 4. Solve classical problem of synchronization using semaphore and monitor. 5. Define deadlock and safe state.	4

	avoidance, recovery from deadlock banker's algorithm.	6. List necessary conditions for occurrences of deadlock. 7. Discuss various methods of handling deadlock. 8. Explain Banker's algorithm.	
5	**File System :** Concept of file and directories, File system Implementation, Security Issues in File, Protection Mechanism, Case Studies of UNIX file system	1. Define file. 2. List different file attributes. 3. Explain the basic operations that can be performed on files. 4. Discuss various file access methods. 5. Discuss directory structures and file system protection. 6. Explain various methods of allocating disk space. 7. Compare various approach of free-space management.	5
6	**I/O System:** Principles of I/O Hardware, Principles of I/O Software, Mass storage structure - overview of mass storage structure, disk structure, disk attachment, disk scheduling algorithms, swap space management,	1. Define disk, clocks, and port 2. Discuss various disk scheduling algorithm. 3. State the principles of I/O Hardware and software	5
7	**Internal assessment**		3

Distribution of Marks:

Unit	Topic	Type of Question			Total Marks
		Objective	Short	Descriptive	
1	Introduction to the operating system	1	2	4	7
2	Process Management	2	2	4	8
3	Memory Management	5	5	4	14
4	Synchronization and Deadlock	5	6	4	15
5	File System	3	2	4	9
6	I/O System	1	2	4	7
	Total	17	19	24	60

Suggested Learning Resources :

1. Operating System –Madnick and Donovan- MGH
2. Operating System Concepts –A. Silberschatz and P. Galvin –ADP
3. The UNIX Programming Environment – by Kernighan & Pike –PHI
4. UNIX – concepts & Application – by Sumitabha Das

Course Title : Computer Architecture and Organisation

Semester : 4th

Course Code: COPC-402

Credits : 2 (L : 2 T : 0 P : 0)

Course Objectives :

- To introduce fundamental concepts of computer architecture, functional units, logic circuits, and data representation techniques.
- To explain basic computer organisation, instruction formats, addressing modes, and control mechanisms.
- To develop an understanding of CPU operation, microprocessor architecture (Intel 8085), and assembly-level programming.
- To explain memory and I/O organisation, hierarchy, transfer modes, interrupts, and DMA mechanisms.

Course Outcomes :

After completing this course, students will be able to:

CO1: Describe the architecture, functional units, and bus structure of a computer.

CO2: Apply number systems and complements for numerical representation and arithmetic operations.

CO3: Explain instruction cycle, addressing modes, CPU organisation and differentiate between hardwired and microprogrammed control.

CO4: Demonstrate understanding of Intel 8085 microprocessor architecture and write basic assembly programs.

CO5: Describe memory hierarchy, cache/virtual memory, and various input-output transfer mechanisms, including interrupts and DMA.

Teaching Scheme:

	Teaching Scheme		
L	T	P	Total Hours per week
2	0	0	2

Examination Scheme:

	Internal Assessment (Theory)	ESE
Full Marks	40	60
Pass Marks	16	24

Detailed Course Content :

Unit	Topics/ Sub-topics	Intended Learning Outcome	Hours
1	**Introduction to Computer Architecture :** Stored program concept — Von Neumann architecture (definition only) , Functional units, Bus structure, brief introduction to logic gates, flip-flop, encoder/decoder, multiplexer, half adder/full adder	1. Define a computer and identify its parts. 2. Define and explain computer bus architecture . 3. Define various logic gates and combinational circuits .	3
2	**Number Representation** : Signed numbers, Fixed point and floating point representation, Complements, various Arithmetic operation (with signed magnitude &	4. Define various number system like decimal, binary, octal & hexadecimal number systems. 5. Conversion from one number system to another number system	4

	floating point numbers), Booth's algorithm	6. Define 1's & 2's complements. 7. Subtract using 1's & 2's complements. 8. Describe working of Booth's algorithm.	
3	**Basic Computer Organisation and Design :** Instruction codes, Computer registers, computer instructions, Timing and Control, Instruction cycle, Instruction Pipeline , Memory-Reference Instructions, Input-output and interrupt, Complete computer description, Design of Basic computer, design of Accumulator Unit.	9. Define computer codes and instructions . 10. Define various I/O and concept of interrupt . 11. Design a computer and Accumulator unit .	3
4	**Central Processing Unit :** Component of ALU(in block diagram only), General Register organisation, Stack organisation, Instruction format, Different types of Instructions, addressing modes; Hardware control unit- its different functions, Microprogrammed control unit – control memory, micro program example, design of control unit. Introduction to Microprocessor (Intel 8085)- Pin Description and Internal Architecture, Instruction set (Addressing modes, Data transfer, arithmetic, logical, branch, stack and machine control groups of instruction set), Introduction to assembly language programming.	12. Define the various components of ALU. 13. Explain the difference between register and stack organisation of CPU design . 14. Define various instruction formats and addressing modes. 15. Explain the difference between Hardwired and Micro programmed control unit . 16. Explain the working of Intel 8085 microprocessor. 17. Write Assembly language program using 8085 instruction set.	10
5	**Memory Organisation :** Concept of bits, bytes and words, storage of numbers and characters, Memory hierarchy, Main memory, Auxiliary memory, Associative	18. Define bit, byte and word. 19. Explain the memory hierarchy . 20. Define the various types of memory used in a computer system.	4

	memory, Cache memory, Virtual memory	21. Explain cache memory and virtual memory .	
6	**Input-Output Organisation :** Various I/O devices, Input-output interface, Asynchronous Data Transfer, Mode of Transfer, Priority interrupt, DMA, Input-output processor	22. Define various modes of data transfer. 23. Explain the concept of Interrupt and DMA.	3
7	**Internal Assessment**		3

Distribution of Marks:

Unit	Topic	Type of Question			Total Marks
		Objective	Short	Descriptive	
1	Introduction to Computer Architecture	1	2	4	7
2	Number Representation	2	2	4	8
3	Basic Computer Organisation and Design	5	5	4	14
4	Central Processing Unit	5	6	4	15
5	Memory Organisation	3	2	4	9
6	Input-Output Organisation	1	2	4	7
	Total	17	19	24	60

Suggested Learning Resources :

1. Computer System Architecture – by Mano ,PHI
2. Computer Organisation & Architecture – by Stallings, PHI
3. Computer Organisation – by Carl Hamacher, ZvonkoVranesic and SafwatZaky, MGH
4. Microprocessor Architecture , Programming and Application with the 8085- Ramesh S Gaonkar

Course Title: Data Structure

Semester: 4th

Course Code: COPC-403

Credits : 3 (L : 3 T : 0 P : 0)

Course Objectives :

- To develop appropriate data structures in developing software to solve problems.
- To learn principles of algorithms.
- To familiarize with control and data structures of any programming language.

Course Outcomes :

After completing this course, students will be able to:
:

CO1: Explain the basic concepts of data structures and algorithms.

CO2: Analyse the various sorting and searching algorithms.

CO3: Implement linear data structure such as stacks, queues, linked lists.

CO4: Implement basic operations on binary trees, graphs and their traversal techniques and files.

Teaching Scheme:

	Teaching Scheme		
L	T	P	Total Hours per week
3	0	0	3

Examination Scheme:

	Internal Assessment (Theory)	ESE
Full Marks	40	60
Pass Marks	16	24

Detailed Course Content:

Unit	Topic/ Sub-topic	Intended Learning Outcomes	Hours
1	**Introduction to Data Structures** 1.1 Concept and Need of Data Structure, Definition, Abstract Data Type 1.2 Types of Data Structures: (i) Linear Data Structures (ii) Non-Linear Data Structures 1.3 Operations on Data Structures: (i) Traversing (ii) Insertion (iii) Deletion 1.4 Algorithms, Complexity- Time and space	1. Define data structure. 2. Define basic terminologies related to data structure. 3. Define complexity of an algorithm.	**4**
2	**Searching and Sorting** 2.1 Searching: Searching for an item in a data set using the following methods: (i) Linear Search (ii) Binary Search 2.2 Sorting: Sorting of data set in an order using the following methods: (i) Bubble Sort (ii) Selection Sort (iii) Insertion Sort (iv) Quick Sort (v) Bucket Sort (vi)Merge Sort	1. Define and distinguish between sorting and searching. 2. Distinguish between linear and binary search 3. Explain various sorting algorithms and find out their running time complexity (best case, average case and worst case).	**6**
3	**Linked List** 3.1 Difference between Static and Dynamic Memory Allocation.	1. Differentiate between Static and Dynamic Memory Allocation.	**6**

	3.2 Introduction to Linked List 3.3 Type of Lists: Linear List, Circular List, Doubly Linked List. 3.4 Operations on a Singly Linked List: Creating a Linked List, Inserting a new node in a Linked List, Deleting a node from a Linked List, Traversing a Singly Linked List.	2. Create a suitable structure using a Linked List to represent a Node. 3. Create Algorithm to add or remove a specified item from a Linear Linked List	
4	**Stack** 4.1 Introduction to Stack: Definition, Stack as an ADT, Operations on Stack- (Push, Pop) 4.2 Stack Implementation using Array and representation using Linked List. 4.3 Applications of Stack: Reversing a List, Polish Notations, Conversion of Infix to Postfix Expression, Evaluation of Postfix Expression etc. 4.4 Recursion: Definition and Applications.	1. Represent Stack using Array and Linked List. 2. Create Algorithm to carry out the PUSH and POP operations in a Stack. 3. Use Stack to transform the given expression from Infix to Postfix. 4. Evaluate Postfix Expression. 5. Explain Recursion and Application of Recursion	**5**
5	**Queue** 5.1 Introduction to Queue: Queue as an ADT, Queue representation in memory using Array and representation using a Linked List. 5.2 Types of Queues: Linear Queue, Circular Queue, Concept of Priority Queue, Double-Ended Queue. 5.3 Queue Operations: INSERT, DELETE,	1. Represent Queue using Array and Linked List. 2. Explain the characteristics of different types of Queue. 3. Create Algorithm to carry out the INSERT and DELETE Operations on a Queue.	**4**

6	**Tree** 6.1 Introduction to Trees Terminologies: 6.2 Tree Types: General Tree, Binary Tree, Binary Search Tree (BST). 6.3 Binary Tree Traversal: In-Order Traversal, Preorder Traversal, Post-Order Traversal. 6.4 Heap, Heap Sort	1. Define tree, binary tree, Binary Search Tree and various terms related to tree. 2. Explain various operations on trees and corresponding algorithms related to tree traversal. 3. Define heap, explain heap sort algorithm	**6**
7	**Graphs and Their Application:** 8.1 Graph Theory Terminology 8.2 Sequential Representation of Graphs, Adjacency matrix, Path Matrix 8.3 Linked Representation of a graph 8.4 Traversing a Graph, DFS, BFS 8.5 Spanning tree: Prim's and Kruskal's algorithm	1. Define graph and graph theory related terminology. 2. Explain representation of graphs using adjacency matrix and linked list. 3. Discuss graph traversal algorithms 4. Define spanning tree and algorithms of Spanning Tree.	**6**
8	**Introduction to File Organization :** 8.1 Sequential, Index-sequential and Direct file Organization.	5. Explain various file organisation.	**2**
9	**Internal Assessment (Presentation Class test)**		**6**

Distribution of Marks:

Unit	Topic	Type of Question			Total Marks
		Objective	Short	Descriptive	
1	Introduction to Data Structures	1	2	3	6
2	Searching and Sorting	2	3	3	8

3	Linked List	2	4	3	9
4	Stack	2	2	3	7
5	Queue	2	2	3	7
6	Tree	2	3	3	8
7	Graphs and their Application	2	3	3	8
8	Introduction to File Organization	2	2	3	7
	Total	15	21	24	60

Suggested Learning Resources:

1. Data Structure : Seymoler Lipschutz (Schaum Series)
2. Fundamentals of Computer Algorithms- by Horowitz, E & Sahani, S- Galgotia
3. Data Structure Theory Applications: Trembly & Sorenson, TMH

Course Title - Digital Electronics

Semester- 4th

Course Code : COPC-404

Credits : 3 (L : 2 T : 1 P : 0)

Course Objectives :

- To develop appropriate number systems for digital electronics.
- To learn principles of boolean algebra and combinational digital system.
- To familiarize with digital logic circuits.

Course Outcomes :

After completing this course, students will be able to:

CO1: Comprehend and apply various number systems for solving digital electronics problems.
CO2: Explain and utilize the principles of Boolean algebra, logic gates and logic families in digital circuit design.
CO3: Analyze and design combinational and sequential circuits, including flip flops, counters, registers and data converters(D/A and A/D).
CO4: Demonstrate understanding of different memory devices and display circuits used in digital systems.

Teaching Scheme:

Teaching Scheme			
L	T	P	Total Hours per week
2	1	0	3

Examination Scheme:

	Internal Assessment (Theory)	ESE
Full Marks	40	60
Pass Marks	16	24

Detailed Course Content:

Unit	Topic/ Sub topic	Intended Learning Outcomes	Hours
1	**Understanding of number system** : Binary, Octal, Decimal, Hexadecimal number system and their conversion. Different coding system i.e. binary, BCD, Excess 3, Grey, ASCII and Parity Code	1. To understand the various number system. 2. To know the conversion of one number system to another.	5
2	**Boolean Algebra:** Definition of Boolean constant, variables and function, the rules of Boolean algebra, De Morgan's theorem, Min term, Max term, POS, SOP, Karnaugh Maps and their use for simplification of Boolean expressions, Don't care conditions.	1. To get familiarize with the principles & rules of Boolean algebra. 2. To understand the different procedure for simplification of Boolean expressions.	5
3	**Logic Gates and Logic Families :** **Concepts of logic gates:** Symbol, definition, truth table and logic expression of OR, AND, NOT, NOR, NAND, X-OR, X-NOR gates, Universal Gates, Realization of logic circuit from their logic expression. **Realization of logic families:** RTL, DTL, TTL, PMOS, NMOS and CMOS family. Designing of Universal Gates using NMOS, PMOS and CMOS.	1. To conceptualize logic gates. 2. To distinguish different logic gates. 3. To learn the realization of logic families.	6
4	**Combinational Digital System : Arithmetic circuits**	1. To analyze & construct combinational digital system.	5

	Half Adder, Full Adder, Half Subtractor, Full Subtractor, Encoder, Decoder, Multiplexer, Demultiplexer Digital Comparator, Parity Checker/Generator BCD to Seven Segment Decoder		
5	**Flip Flops :** Latch, R-S, J-K, T, D flip flops, Clocked flip flop, Master slave J-K flip flop.	1. To understand the digital logic circuit of different flip flops. 2. To distinguish and evaluate the different flip flops.	4
6	**Registers & Counters :** Shift Register, Serial in Serial out (SISO), Serial in Parallel out (SIPO), Parallel in Serial out (PISO), Parallel in Parallel out (PIPO) **Counters:** Synchronous, Asynchronous Counter, Ring Counter, Up/Down Counter	1. To get define the register & counter. 2. To understand the digital logic circuit of register & counter. 3. To distinguish different registers & counters.	5
7	**Data Converters & Memory Devices:** **Data Converters** – Digital to Analog converters, Analog to Digital Converters **Memory Devices** Classification of Memories – RAM Organization, Address Lines and Memory Size, Static RAM, Bipolar RAM, cell Dynamic RAM, D RAM, DDR RAM **Read Only memory –** ROM organization, Expanding memory, PROM, EPROM, EEPROM, Flash Memory	1. To understand the working of D/A and A/D converter. 2. To describe the function, characteristics & structure of different memory systems.	5
8	**Display Device:** Segment format, Working and operating principle of LED, LCD display, Dot Matrix Format	1. To understand the working of display devices.	4
9	**Internal Assessment (Presentation Class test)**		**6**

Distribution of Marks:

Unit	Topic	Type of Question			Total Marks
		Objective	Short	Descriptive	
1	Number System	2	2	4	8
2	Boolean Algebra	2	3	---	5
3	Logic Gates and Logic	1	4	4	9
4	Combinational Digital	2	2	4	8
5	Flip Flops	2	2	4	8
6	Registers & Counters	2	2	4	8
7	Data Converters & Memory	2	---	4	6
8	Display Device	2	2	4	8
Total		15	17	28	60

Suggested Learning Resources:

1. Modern Digital Electronics –RP Jain
2. Digital Principles & Application –Malvino & Leach
3. Digital Fundamentals – Thomas L. Floyed
4. Digital Electronics- Douglas V. Hall
5. Digital Electronics- A. Anand Kumar.

Course Title – Operating System Lab

Semester : 4th

Course Code : COPC-405

Credits : 1 (L : 0 T : 0 P : 2)

Course Objective :

- To use Linux/UNIX commands for process, file, and directory management.
- To write C / C++ / Java / Python programs related to OS concepts.
- To implement scheduling, memory, and synchronization algorithms in code.
- To perform shell scripting for automation of OS tasks.
- To use debugging tools, compiler commands, and command-line utilities.

Course Outcomes :

After completing this course, students will be able to:

CO1. Use the UNIX/Linux environment to execute essential OS commands, perform file and directory operations, write shell scripts, and manage processes using system utilities.

CO2. Develop, implement, and evaluate key operating system functionalities such as CPU scheduling, process creation/termination, inter-process communication, and file system operations.

CO3. Apply and analyse memory management techniques, including paging and page replacement algorithms, to optimise system performance.

Pre-requisites of the Course:

- Knowledge of programming

Teaching Scheme:

Teaching Scheme			
L	T	P	Total Hours per week
0	0	2	2

Examination Scheme:

	Internal Assessment (Practical)	Practical Test
Full Marks	60	40
Pass Marks	24	16

Laboratory Exercises (Operating System Practical Job List)

Expeiments	Topic/Subtopic	Hours
1.	Study and demonstration of basic functions of an operating system (booting, user interface, resource management).	2
2.	Execution of basic commands • DOS/Windows file and directory commands. • DOS/Windows system and disk commands. • UNIX/Linux file and directory commands. • UNIX/Linux file permission commands (chmod, chown, umask). • UNIX/Linux filters, pipes, and redirection commands. • UNIX/Linux process-related commands (ps, top, kill, nice, renice).	2
3.	Shell script • to perform arithmetic, conditional statements, and loops. • for file backup and search operations.	2
4.	Program to create multiple processes using fork() system call.	2
5.	Demonstration of process states using Linux tools.	2
6.	Simulation of CPU scheduling algorithms: • FCFS • SJF • Priority Scheduling • Round Robin	2
7.	Implementation of Classical Problem of Synchronisation • Solving Producer–Consumer problem using semaphores. • Simulating Reader–Writer problem using synchronization techniques. • Simulation of Dining Philosopher's Problem	2

8.	Demonstration of Peterson's solution for mutual exclusion.	2
9.	Program to implement Banker's Algorithm for Deadlock Avoidance	2
10.	Simulation of memory allocation techniques: • First Fit • Best Fit • Worst Fit.	2
11.	Implementation of page replacement algorithms: FIFO, LRU, and Optimal.	2
12.	Simulation of demand paging. • Simulation of disk scheduling algorithms: o FCFS o SSTF o SCAN o C-SCAN.	2
13.	Simulation of file allocation methods: • Sequential • Linked • Indexed	2
14.	Study of distributed system concepts through a case study (e.g., Hadoop or Google File System).	2
15.	Class test	**2**

Reference Books :

1. UNIX Shell Programming – Yashvant P Kanetkar - BPB Publications

2. Operating System Concepts –A. Silberschatz and P. Galvin –ADP

3. UNIX – Concepts & Application – by Sumitabha Das

Course Title: Data Structure Lab

Semester: 4th

Course Code: COPC-406

Credits : 1 (L : 0 T : 0 P : 2)

Course Objective:

- To implement Sorting and Searching.
- To implement Linear Data Structure.
- To implement Non-Linear Data Structure.

Course Outcome:

After completion of the course, students will be able to:

CO1: Apply Algorithm for executing programs of sorting and searching.

CO2: Execute programs on Linear Data Structure.

CO3: Execute programs on Non -Linear Data Structure.

Pre-requisites of the Course:

- Knowledge of programming

Teaching Scheme:

Teaching Scheme			
L	T	P	Total Hours per week
0	0	2	2

Examination Scheme:

	Internal Assessment (Practical)	Practical Test
Full Marks	60	40
Pass Marks	24	16

Detailed Course Content:

Unit	Topic/ Sub-topic	Hours
1.	**Searching and Sorting** 1. Perform Linear Search and Binary Search on an array 2. Implement Bubble sort, Insertion sort, Selection Sort, Merge sort, Quick sort using array as data structure.	**6**
2.	**Linked List** Create a linked list with nodes and perform the following 1. Display the linked List 2. Insert a new node at specified position. 3. Delete a node at specified position.	**4**
3.	**Stack** Create a stack and perform Pop, Push, Traverse operations on the stack using array.	**4**
4.	**Queue** Create a Linear Queue using array and implement different operations such as Insert, Delete and Display the queue elements.	**4**
5.	**Tree** Create a Binary Tree and perform the following Tree traversals 1. Preorder 2. Post order 3. In order	**4**
6.	**Graphs** Representation of Graph Graph Traversal -BFS and DFS	**4**
7.	**Class Test**	**4**

Suggested Learning Resources:

1. Data Structure : Seymoler Lipschutz (Schaum Series)
2. Fundamentals of Computer Algorithms- by Horowitz, E & Sahani, S- Galgotia
3. Data Structure Theory Applications: Trembly & Sorenson, TMH

Course Title : Multimedia Lab

Semester : 4th

Course Code : COPE- 401

Credits : 2 (L : 0 T : 1 P : 3)

Course Objectives:

- To learn the concept of sound editing and animation.
- To learn the concept of image editing.
- To learn the concept of video editing.

Course Outcome:

On completion of this course students will be able to:

CO1: Create, mix and design sound effects using sound editing software.

CO2: Design effective graphics using photo editing software

CO3: Develop videos by importing clips and edit using video editing software.

Pre-requisites:

- Familiarity with fundamental concepts of graphics, audio, and video formats used in daily computing.
- Ability to use computer software applications for editing and content creation.

Teaching Scheme:

Teaching Scheme			
L	T	P	Total hours per week
0	1	3	4

Examination Scheme:

	Internal Assessment (Practical)	Practical Test
Full Marks	60	40
Pass Marks	24	16

Detailed Course Content:

Unit	Topic/ Sub-topic	Hours
1.	**Sound Editing** Importing, Recording, Editing and Repairing Audio. Mixing and Applying Effects on Audio,	**8**
2.	**Image Editing** Working with Images using different tools, Layers, Colour Palettes, Animation.	**8**
3.	**Video Editing** Capturing Clips, Editing, Mixing, Synchronization with Audio.	**10**
4.	**Class Test**	**4**

Suggested Learning Resources:

I. Tay Vaughan, *Multimedia: Making It Work*, 9th Edition, McGraw-Hill Education.

II. Ze-Nian Li & Mark S. Drew, *Fundamentals of Multimedia*, Pearson Education.

III. Ranjan Parekh, *Principles of Multimedia*, 2nd Edition, Tata McGraw-Hill.

Course Title – Scripting Language(Python) Lab

Semester: 4th

Course Code: COPE-402

Credits : 2 (L : 0 T : 1 P : 3)

Course Objectives

- **To familiarize students with fundamental concepts and syntax of Python programming**, enabling them to write, debug, and execute basic programs effectively.
- **To equip students with the skills to apply Python programming techniques** for solving simple computational and real-world problems.
- **To provide a strong foundational understanding of Python** for practical applications in diverse domains such as data analysis, web development, database management, and machine learning.

Course Outcome:

After completion of the course students will be able to:

CO1: Write simple Python programs using basic syntax, data types, and control structures such as loops and conditionals

CO2: Use Python to solve common computational problems and automate simple tasks, including reading from and writing to files.

CO3: Analyse and visualize data using basic Python data structures and libraries

CO4: Design and Develop python programs in the field of Web Development, Database Management System and Machine Learning

Pre-requisites:

1. Concept of programming paradigms.
2. OOPs concept.

Teaching Scheme:

Teaching Scheme			
L	**T**	**P**	**Total Hours per week**
0	**1**	**3**	**4**

Examination Scheme:

	Internal Assesment (Practical)	Practical Test
Full Marks	60	40
Pass Marks	24	16

Detailed Course Content:

Unit No.	Topic/Subtopic	Hours
1	**Introduction to python environment** • Overview of Python • Setting up the Python environment • Writing and running your first Python script • Python IDEs: IDLE, PyCharm, Jupyter Notebook • Basic syntax: comments, variables, and data types	4
2	**Control Structures** • Conditional statements: if, else, elif • Looping: for, while • Control flow: break, continue, pass	3
3	**Functions and Modules** • Defining and calling functions • Function arguments and return values • Introduction to modules: ○ Organizing projects into modules, Importing own module as well as external modules • Understanding packages	4
4	**File Handling, Error and Exception Handling** • Reading from and writing to files • Working with file paths • Understanding exceptions • Try, except, else, and finally blocks • Handling exceptions in file operations	3
5	**Object Oriented Programming in Python**	4

	• class and object creation • constructor and destructor • inheritance • polymorphism	
6	**Working with Libraries** • NumPy for numerical computations • Pandas for data manipulation • Matplotlib for data visualization	5
7	**Database Programming with Python:** • Introduction to Python's database API • Connecting to various relational databases • Executing SQL Queries: • Fetching result	4
8	**Internal Assessment**	3

Suggested Learning Resources:

1. Taming Python by Programming, Jeeva Jose, Khanna Publishing House
2. Starting Out with Python, Tony Gaddis, Pearson
3. Core Python Programming, Wesley J. Chun, Prentice Hall
4. Python Programming: Using Problem Solving Approach, Reema Thareja, Oxford University
5. Introduction to Computation and Programming Using Python. John V. Guttag, MIT Press.
6. Beginning Python using Python 2.6 and Python 3, James Payne, Wrox publishing
7. Programming Python, Fourth Edition by Mark Lutz, O'Relly

Course Title – Visual Programming Lab

Semester- 4th

Course Code : OE-401

Credits : 2 (L : 0 T : 1 P : 3)

Course Objectives:

- To understand the .Net Framework.
- To Develop Proficiency in Visual programming concepts and tools.
- To Apply Problem-Solving Skills in Application Development.
- To Implement and Customize Graphical User Interfaces (GUIs)

Course Outcome:

After completion of this course students will be able to:

CO1: Develop applications using VB.NET

CO2: Assemble multiple forms, modules, and menus to create VB.NET project.

CO3: Develop database driven application using ADO.NET.

Pre-requisites:

3. Concept of programming paradigms.
4. OOPs concept.

Teaching Scheme:

Teaching Scheme			
L	T	P	Total hours per week
0	1	3	4

Examination Scheme:

	Internal Assessment (Practical)	Practical Test
Full Marks	60	40
Pass Marks	24	16

Detailed Course Content:

Unit No.	Topics/Sub-Topics	Hours
1	**Getting Started with VB.NET** – Introduction to VB.NET and Visual Basic Forms – Creating a New Project and Saving Work – Using Toolbox and Adding Basic Controls (Textbox, Buttons) – Variables and String Variables – Writing and Running First VB.NET Code	3
2	**Conditional Logic, Loops and console application** – If Statement and Select Case – Conditional Operators – Introduction to Loops: For Loop and Do Loop – Basic Programs using Loops –Reading input from the console and displaying output to the console – Writing console programs using decision-making and loops	4
3	**Menus, Controls and User Interface** – Adding Menus and Sub-Menus to Forms – Writing Code for Menu Items (Cut, Copy, Paste, Undo) – Show/Hide Controls – PictureBox and Adding Images – Checkbox and Option Button (Radio Button) Handling	3
4	**Debugging and Error Handling** – Types of Errors: Design-Time, Run-Time, Logic – Try–Catch Structure – Breakpoints and Debugging Tools	3

5	**Arrays and String Manipulation** – Creating and Assigning Values to Arrays – String Basics and Common String Methods (Trim, InStr, Substring, Equals, Replace, Insert, Split, Join)	3
6	**Functions, Subs and Events** – Introduction to Subs and Functions – ByVal and ByRef – Creating Functions and Subs – Common Events: Click, MouseDown, KeyDown, Form Load	2
7	**Object-Oriented Programming** – Classes and Objects – Creating Methods and Properties – Basic Inheritance	2
8	**VB.NET and Databases** – Using Database Wizard – DataSets and DataAdapters – Displaying Data and Navigation – Basic Add, Update, Delete Operations	3
9	**Working with Multiple Forms** – Anchor and Dock Controls – Adding Toolbars – Creating Multiple Forms – Modal and Non-Modal Forms – Passing Values Between Forms	3
10	**Internal Assessment**	4

Suggested Learning Resources:

1.Beginner's Guide to Visual Basic .NET Programming: A Practical Approach to VB.NET by Serhan Yamacli

2. Programming in Visual Basic. Net by Julia Case Bradley

3. Visual Basic. Net by C Muthu

Course Title: Fundamentals of Data Analysis and Cloud Computing (IS)

Semester : 4th

Course Code: OE-402

Credits : 2 (L : 2 T : 0 P : 0)

Course Objective :

- To provide students with foundational knowledge of data analysis principles **and** cloud computing concepts**.**
- To introduce how data is collected, processed, and interpreted to generate meaningful insights, and how modern cloud technologies offer scalable, secure, and efficient platforms for storing and analyzing data.

Course Outcome:

After completion of the course, students will be able to:

CO1: Analyze, visualize, and interpret datasets to identify underlying patterns and trends.

CO2: Create interactive GUI application using Python Programming concept.

CO3: Describe the devices with sensors, processing ability, software and other technology that connect and exchange data with other systems

CO4: Understand core cloud concepts, characteristics, service models, and deployment architectures.

.

Pre-requisites of the Course:

- Basic knowledge of computers, internet usage, simple mathematics, and spreadsheets is required for this course.

Examination Scheme:

	Internal Assessment	ESE
Full Marks	40	60
Pass Marks	16	24

Detailed Course Content:

Course Code	Course Name	Infosys Course	Link	Hours
OE-402	Fundamentals of Data Analysis and Cloud Computing.(IS)	Introduction to Data Science	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_12666306402263577000_shared/overview	3h 30 mins
		Programming in Python	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013842098630598656122 50_shared/overview#:~:text=feedback-,Programming%20in%20Python%20Beginner,-Programming%20in%20Python	3h 53mins
		Developing with MongoDB	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013094411833860096184 5_shared/overview	4h 3mins
		Introduction to Cloud computing	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_29245015089922640000_shared/overview	5h 44mins
		Internet of Things 101	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_21553622882521997000_shared/overview#:~:text=feedback-,Internet%20of%20Things%20101,-This%20course%20introduces	8h 23mins
		Introduction to Data Warehouse Testing	https://infyspringboard.onwingspan.com/web/en/app/toc/lex_11254008285505206000_shared/overview	6h 13min

The assessment will be based on the following:

Internal Assessment:

i. Report Submission
ii. Presentation /Viva

ESE Assessment :

i) Based on course completion certificate from Infosys Spring Board.

Course Title - Internship

Semester- 4th

Course Code : SI-01

Credits : 2 (L : 0 T : 0 P : 4)

Course Objective:

1. To strengthen the association of students with industry/organization.
2. To create awareness amongst the students, the recent trends of computer engineering in industries/Organization.
3. To percept the role and responsibility in an industry/oreganization.

Course Outcome:

After completion of the internship students will be able to:

CO1: To learn the application of knowledge in real world problems.
CO2: To get exposure to team-work and leadership quality.
CO3: To deal with industry-professionals and ethical issues in the work environment.

Examination Scheme:

	Assessment
Full Marks	100
Pass Marks	40

Course Contents:

- It is mandatory that student has to complete an Industrial Internship for a duration of two weeks during the End semester break of Semester III .
- The internship should be carried out in any industry/organization with relevant applications of Computer Engineering, under the guidance of a mentor from the same industry

- The student will maintain a log of work done on daily basis and important ideas or practices that he / she has learnt during the internship.
- The student has to submit a report based on the internship immediately after the completion of the internship.

The assessment will be based on the following:

ii) The work executed/completed at industry/organization by student.
iii) Presentation on the experience gained.
iv) The submission of the report.
v) Completion certificate from the industry/organization.

Course Title – Minor Project

Semester- 4th

Course Code : PR-01

Credits : 2 (L : 0 T : 0 P : 4)

Course Objective:

Course Objectives:

- To enhance programming, analysis, design, testing, and problem-solving abilities.
- To create tangible work (like an app, website, or system) to showcase skills to employers.
- To prepare for technical interviews by demonstrating enthusiasm and practical competency.

Course Outcome:

After completion of the Mini Project students will be able to:

CO1: Apply technical knowledge and skills to develop a working solution for a real- world scenario.
CO2: Develop skills and critical thinking to prepare for major project.
CO3: Develop team building, co-ordination and presentation of project works to the evaluators.

Examination Scheme:

	Assessment
Full Marks	100
Pass Marks	40

Course Contents:

A minor project is an assignment that the student needs to complete during the semester to strengthen the understanding of fundamentals through effective application of the subjects learnt.

Guidelines for Minor Project

i) Students are expected to undertake one minor project during the semester based on the courses learnt in the current semester along with all the subjects learnt in earlier semesters.
ii) The students may take up the project individually or in group. However, if project is done in group, each student must be given a responsibility for distinct modules.
iii) Selected project/module must have relevant scope as per the marks assigned and must be carried out in the institute.
iv) Internal guide should monitor and evaluate the progress of the project. Students are expected to show working demo of the project during final evaluation.
v) Students are expected to submit a mini project report as a part of final submission

The assessment will be based on the following:

The minor project will be assessed internally by the project committee of the department on the following criteria

i) The project execution /completion by the student/group.
ii) Presentation on the minor project.
iii) The submission of the report.

****