

DEPARTMENT OF CSE(AI&ML)

SEMESTER 3



25UMAT301	PROBABILITY, RANDOM PROCESSES AND NUMERICAL TECHNIQUES	Category	L	T	P	R	Credit	Year of Introduction
		BSC	3	0	0	0	3	2025

Preamble:

This course introduces students to the modern theory of probability and random processes, covering important models of random variables and analysis of random processes. A brief course in numerical techniques familiarises students with some basic numerical techniques which are especially useful when analytical solutions are hard to find.

Prerequisite: Nil

Course Outcomes: After the completion of the course, the student will be able to:

CO	Course Outcomes	Knowledge Level
CO1	Explain the concept, properties and important models of discrete random variables and to apply in suitable random phenomena.	K3 Apply
CO2	Explain the concept, properties and important models of continuous random variables and to apply in suitable random phenomena.	K3 Apply
CO3	Demonstrate random processes using autocorrelation, power spectrum and Poisson process model as appropriate.	K3 Apply
CO4	Compute roots of equations, find the value definite integrals and perform interpolation on given numerical data using standard numerical techniques	K3 Apply

Mapping of course outcomes with program outcomes and program specific outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	2	2	-	-	-	-	-	-	1
CO2	3	2	2	2	-	-	-	-	-	-	1
CO3	3	2	2	2	-	-	-	-	-	-	1
CO4	3	2	2	2	-	-	-	-	-	-	1

Avg	3	2	2	2	-	-	-	-	-	-	1
-----	---	---	---	---	---	---	---	---	---	---	---

Mapping of course outcomes with Knowledge and Attitude Profile (WK)

	WK 1	WK 2	WK 3	WK 4	WK 5	WK6	WK 7	WK8	WK9
CO1	*	*	*	*	*			*	
CO2	*	*	*	*	*			*	
CO3	*	*	*	*	*			*	
CO4	*	*	*	*	*			*	

WKs mapped with the course

WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
*	*	*	*	*			*	

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	40	60	2.5 hrs

Continuous Internal Evaluation Pattern:

Attendance	: 5 marks
Continuous Assessment Test	: 20 marks
Continuous Assessment Assignment	: 15 marks

Module 1 (11 hrs)

Discrete random variables and their probability distributions, Expectation, mean and Variance, Binomial distribution, Poisson distribution, Poisson approximation to the binomial distribution, Discrete bivariate distributions, marginal distributions, Independent random variables, Expectation (multiple random variables).

(Text-1: Relevant topics from sections-3.1-3.4, 3.6, 5.1)

Module 2 (11 hrs)

Continuous random variables and their probability distributions, Expectation, mean and variance, Uniform, exponential and normal distributions, Continuous bivariate distributions, marginal distributions, Independent random variables, Expectation (multiple random variables), i. i. d random variables and Central limit theorem (without proof).

(Text-1: Relevant topics from sections-4.1-4.4, 3.6, 5.1)

Module 3 (10 hrs)

Random processes and classification, mean and autocorrelation, wide sense stationary (WSS) processes, autocorrelation and power spectral density of WSS processes and their properties, Poisson process-distribution of inter-arrival times, combination of independent Poisson processes(merging) and subdivision (splitting) of Poisson processes (results without proof).

(Text-2: Relevant topics from sections-8.1-8.5, 8.7, 10.5)

Module 4 (10hrs)

Solution of equations – Newton-Raphson method and Regula- Falsi method. Interpolation-finite differences, Newton's forward and backward difference method, Newton's divided difference method and Lagrange's method. Numerical integration-Trapezoidal rule and Simpson's 1/3rd rule (Proof or derivation of the formulae not required for any of the methods in this module)

(Text 3- Relevant topics from sections 19.1, 19.2, 19.3, 19.5)

Text Books:

1. Devore, J. L. (2012). *Probability and statistics for engineering and the sciences* (8th ed.). Cengage Learning.
2. Ibe, O. C. (2005). *Fundamentals of applied probability and random processes*. Elsevier.
3. Kreyszig, E. (2016). *Advanced engineering mathematics* (10th ed.). John Wiley & Sons.

Reference Books:

1. Hossein Pishro-Nik. (2014). *Introduction to Probability, Statistics and Random Processes*. Kappa Research. <http://www.probabilitycourse.com>
2. V. Sundarapandian. (2009). *Probability, Statistics and Queuing Theory*. PHI Learning.
3. John A. Gubner. (2006). *Probability and Random Processes for Electrical and Computer Engineers*. Cambridge University Press.
4. B. S. Grewal. (2010). *Higher Engineering Mathematics* (36th ed.). Khanna Publishers.

CourseCode	25UMAT301
Course Name	PROBABILITY, RANDOM PROCESSES AND NUMERICAL TECHNIQUES
Max.Marks:60	Duration:2.5Hours

Part A

(Answer all questions. Each question carries 3 marks)

Qn No.	Question	Marks	CO	BL
1	The probability that there will be 0, 1, 2, 3 defective light bulbs in a randomly selected batch of bulbs produced in a factory are 0.6, 0.3, 0.08, 0.02 respectively. Find the mean and variance of the number of defective bulbs in a randomly selected batch	3	1	L3
2	Determine the binomial distribution for which mean is 24 and variance is 6.	3	1	L2
3	The cumulative distribution function of a continuous random variable X is defined by: $F(x) = \dots$. Find (i) the value of k. (ii) $P(X > 4)$	3	2	L2
4	A continuous random variable X is uniformly distributed over $[-4, 6]$. Find the mean and variance of X.	3	2	L3
5	State any three properties of autocorrelation function.	3	3	L2
6	Find the power spectral density of the WSS power process with autocorrelation function $2 + 4 \dots$	3	3	L3
7	Find an approximate root of the equation $x^3 - 2x - 5 = 0$ using Regula Falsi method.	3	4	L2
8	Evaluate using trapezoidal rule in 6 sub intervals.	3	4	L2

Part B

(Answer any one full question from each Module. Each carry 9 marks)

Module I

Qn No.	Question									Marks	CO	BL	
9a	A Random variable X has the following probability distribution function									5	1	L3	
	x	0	1	2	3	4	5	6	7				8
	f(x)	a	3a	5a	7a	9a	11a	13a	15a				17a
	Determine (i) the value of 'a' (ii) $P(X < 3)$												

	(iii) $P(X \geq 3)$ (iv) $P(0 < X < 5)$			
b	Show that Poisson distribution is the limiting case of Binomial distribution as $n \rightarrow \infty$, $p \rightarrow 0$.	4	1	L2

OR

10a	The joint probability mass function of two random variables X and Y is given by where k is a constant. 1) Find the value of k. 2) Find the marginal distribution functions of X and Y. 3) Are X and Y independent?	4	1	L2
b	The monthly breakdown of a computer is a random variable having Poisson distribution with a mean equal to 1.8. Find the probability that this computer will function for a month (1) without a break down (2) with at least one break down.	5	1	L3

Module II

Qn No.	Question	Marks	CO	BL
11a	In an examination, 30% of the students got marks below 40 and 10% got marks above 75. Assuming the marks are normally distributed, find the mean and standard deviation of the distribution.	5	2	L3
b	Let X be a random variable with probability density function 1) Find the constant c. 2). Find E(X) and V(X).	4	2	L2
OR				
12a	The joint probability distribution function of two continuous random variable X and Y is. Check whether X and Y are independent.	4	2	L2
b	A distribution with unknown mean μ has variance equal to 1.5. Use Central Limit Theorem to determine how large a sample should be taken from the distribution in order that the probability will be at least 0.95 that the sample mean will be within 0.5 of the population mean.	5	2	L3

Module III

Qn No.	Question	Marks	CO	BL
13. a	A random process X(t) is defined by $X(t) = 3 \cos(4t + \theta)$, where θ is uniformly distributed in $[0, 2\pi]$. Find the mean, auto correlation and auto covariance.	5	3	L3

13. b	Find the power spectral density function of the WSS process whose autocorrelation function is given by)	4	3	L2
OR				
14. a	If X(t) is a WSS process with autocorrelation . Find the mean and variance.	5	3	L3

14. b	Prove that sum of two independent Poisson processes is a Poisson process.	4	3	L2
Module IV				
Qn No.	Question	Marks	CO	BL
15. a	Find the positive solution of $2\sin x = x$ starting with $x_0 = 2$ using Newton Raphson's method up to 4 decimal places	4	4	L3
15. b	Using Lagrange's interpolation method to find the polynomial $f(x)$ which agree with the following data: $f(1) = 1$, $f(3) = 27$, $f(4) = 64$. Hence find $f(2)$.	5	4	L3
OR				
16. a	Compare by using (i) trapezoidal method (ii) Simpson's method with step size $h = 0.25$. Compare the results with exact value obtained by actual integration.	5	4	L3
16. b	Find $y(3)$ Using Newton's divided difference formula, Given $y(1) = -26$, $y(2) = 12$, $y(4) = 256$, $y(6) = 844$.	4	4	L3



25UCST301	DATA STRUCTURES	Category	L	T	P	R	Credit	Year of Introduction
		PCC	3	1	0	0	4	2025

Preamble: This course serves as a fundamental area of computer science that focuses on the structured organization, storage, and manipulation of data to develop efficient algorithmic solutions. It encompasses both linear and non-linear data structures, algorithm design strategies, and

complexity analysis, equipping learners to assess performance trade-offs and construct optimized solutions for complex computational challenges.

Prerequisite: PIC204 Programming in C.

Course Outcomes: After the completion of the course, the student will be able to:

CO	Course Outcomes	Knowledge Level
CO1	Apply algorithm complexity using asymptotic notations and implement linear data structures including arrays, stacks, and queues.	K3 Apply
CO2	Implement singly, doubly, and circular linked lists and apply suitable memory allocation strategies.	K3 Apply
CO3	Apply tree and graph data structures to perform traversals, searching, and real-world problem solving.	K3 Apply
CO4	Implement and compare sorting algorithms and hashing techniques with appropriate collision resolution strategies.	K3 Apply

Mapping of course outcomes with program outcomes and program specific outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	—	—	—	—	—	—	—	2	3	2
CO2	3	3	2	—	—	—	—	—	—	—	2	3	2
CO3	3	3	3	2	—	—	—	—	—	—	2	3	2
CO4	3	3	2	2	—	—	—	—	—	—	3	3	3
Avg	3	3	2.25	2	—	—	—	—	—	—	2.25	3	2.25

Mapping of course outcomes with Knowledge and Attitude Profile (WK)

	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	*	*	*	*	*	*	*	*	*
CO2	*	*	*	*	*	*	*	*	*
CO3	*	*	*	*	*	*	*	*	*
CO4	*	*	*	*	*	*	*	*	*

Wks Mapped with the course

WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
*	*	*	*	*			*	

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	40	60	2.5 hrs

Continuous Internal Evaluation Pattern:

Attendance	: 5 marks
Continuous Assessment Test	: 20 marks
Continuous Assessment Assignment	: 15 marks

SYLLABUS

Module 1 (14 hrs)

Definitions; Data Abstraction; Performance Analysis - Time & Space Complexity, Asymptotic Notations; Polynomial representation using Arrays, Sparse matrix (Tuple representation); Stacks and Queues - Stacks, Queues, Circular Queues, Double Ended Queues, Priority Queue; Evaluation of Expressions- Infix to Postfix, Evaluating Postfix Expressions Searching Techniques: - Linear Search, Binary Search.

Module 2 (14 hrs)

Singly Linked List - Operations on Linked List, Stacks and Queues using Linked List, Polynomial representation using Linked List; Doubly Linked List; Circular Linked List; Memory allocation - First-fit, Best-fit, and Worst-fit allocation schemes.

Module 3 (14 hrs)

Trees: - Representation Of Trees; Binary Trees - Types and Properties, Binary Tree Representation, Tree Operations, Tree Traversals; Expression Trees; Binary Search Trees - Binary Search Tree Operations; Binary Heaps- Binary Heap Operations. Graphs: - Definitions; Representation of Graphs; Depth First Search and Breadth First Search; Applications of Graphs. Case Study: Design and Implementation of Tree and Graph Data Structures in a Smart City Management System.

Module 4 (14 hrs)

Sorting Techniques: - Selection Sort, Insertion Sort, Quick Sort, Merge Sort, Bubble Sort, Heap Sort, Radix Sort. Hashing - Hashing functions: Mid square, Division, Folding, Digit Analysis; Collision Resolution: Linear probing, Quadratic Probing, Double hashing, Open hashing.

Text Books:

1. Tenenbaum, A. M., Langsam, Y., & Augenstein, M. J. (2021). Data structures and algorithms in C (2nd ed.). Prentice Hall.
2. Horowitz, E., Sahni, S., & Mehta, D. (2022). Fundamentals of data structures in C (6th ed.). University Press.
3. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). Introduction to algorithms (4th ed.). MIT Press.

Reference Books:

1. Thareja, R. (2021). Data structures using C (3rd ed.). Oxford University Press.
2. Weiss, M. A. (2019). Data structures and algorithm analysis in C (4th ed.). Pearson.
3. Tremblay, J. P., & Sorenson, P. G. (2017). Introduction to data structures with applications (2nd ed.). Tata McGraw Hill.

END SEMESTER EXAMINATION – MODEL QUESTION PAPER

Course Code	25UCST301
Course Name	DATA STRUCTURES
Max. Marks: 60	Duration: 2.5 Hours

Part A (Answer all questions. Each question carries 3 marks)				
Qn No.	Question	Marks	CO	BL
1.	Define time complexity and space complexity. Explain asymptotic notations O , Ω , and Θ with examples.	3	CO1	L2
2.	Illustrate an algorithm to implement a circular queue using arrays. Show how it avoids the wastage problem of a linear queue.	3	CO1	L3
3.	Illustrate an algorithm to insert a node at the beginning, middle, and end of a singly linked list.	3	CO2	L3
4.	Differentiate singly linked list and doubly linked list. Explain the memory allocation strategies: First-fit, Best-fit, and Worst-fit.	3	CO2	L2
5.	Explain the inorder, preorder, and postorder traversal of a binary tree with an example.	3	CO3	L2
6.	Describe BFS and DFS traversal techniques for a graph with a suitable example.	3	CO3	L2
7.	Compare Quick Sort and Merge Sort in terms of time complexity and space requirements.	3	CO4	L2
8.	Explain the mid-square and division hashing functions. How is collision resolved using linear probing?	3	CO4	L2

Part B (Answer any one full question from each Module. Each carry 9 marks)				
Module I				
Qn No.	Question	Marks	CO	BL
9.1	Explain the evaluation of arithmetic expressions using stacks. Convert the infix expression $A*(B+C)/D-E$ to postfix and evaluate it assuming $A=2$, $B=3$, $C=4$, $D=7$, $E=1$.	5	CO1	L3
9.2	Describe the sparse matrix tuple representation. Perform the addition of two sparse matrices.	4	CO1	L3
OR				
10.1	Explain the concept of Priority Queue and Double-Ended Queue (Deque) with their operations and applications.	5	CO1	L2
10.2	Illustrate an algorithm for binary search. Compare its performance with linear search using Big-O notation.	4	CO1	L3
Module II				
Qn No.	Question	Marks	CO	BL
11.1	Implement a singly linked list with operations: creation, insertion at various positions, deletion, and traversal. Trace	5	CO2	L3

	the operations with a sample list.			
11.2	Explain how polynomials are represented using linked lists. Add two polynomials: $P1 = 3x^3 + 2x + 1$ and $P2 = x^2 + 4x + 5$.	4	CO2	L3
OR				
12.1	Explain doubly linked list. Write algorithms for insertion and deletion in a doubly linked list. Demonstrate traversal in both directions.	5	CO2	L3
12.2	Implement a circular linked list and perform insertion and deletion. Explain how it differs from a singly linked list.	4	CO2	L3
Module III				
Qn No.	Question	Marks	CO	BL
13.1	Explain binary search tree (BST) operations: insertion, deletion, and search. Apply the operations on the sequence: 50, 30, 70, 20, 40, 60, 80.	5	CO3	L3
13.2	Explain binary heap and its operations (insertion and deletion). Demonstrate the max-heap construction for the array [4, 10, 3, 5, 1].	4	CO3	L3
OR				
14.1	Describe the representation of graphs using adjacency matrix and adjacency list. Apply DFS and BFS on the given graph starting from vertex A.	5	CO3	L3
14.2	Explain expression trees. Construct an expression tree for the expression $(a + b) * (c - d) / e$ and perform the traversals.	4	CO3	L3
Module IV				
Qn No.	Question	Marks	CO	BL
15.1	Write an algorithm to perform quick sort with an example. Trace the sort on the array [64, 34, 25, 12, 22, 11, 90] and analyze its best, average, and worst-case complexities.	5	CO4	L3
15.2	Explain hashing with open addressing. Demonstrate double hashing by inserting keys {23, 45, 67, 89, 11} into a hash table of size 10.	4	CO4	L3
OR				
16.1	Explain Merge Sort with a trace on the array [38, 27, 43, 3, 9, 82, 10]. Compare its time and space complexity with Heap Sort.	5	CO4	L3
16.2	Describe the Radix Sort algorithm. Sort the array [170, 45, 75, 90, 802, 24, 2, 66] using radix sort and show each pass.	4	CO4	L3

25UCST302	OPERATING SYSTEMS	Category	L	T	P	R	Credit	Year of Introduction
		PCC	3	1	0	0	4	2025

Preamble: This course provides a comprehensive understanding of the overall functioning of a computer system, the trade-offs between performance and functionality, and how responsibilities are divided between hardware and software. It introduces key concepts such as memory management, device management, process management and file management within an operating system. It enables learners to grasp the fundamental principles of operating system design, empowering them to analyze issues, enhance system features, resolve operating system problems, and manage computer resources effectively.

Prerequisite: PIC204 Programming in C

Course Outcomes: After the completion of the course, the student will be able to:

CO	Course Outcomes	Knowledge Level
CO1	Illustrate the concepts of process management and process scheduling in Operating systems	K3 Apply
CO2	Illustrate process synchronization mechanisms and deadlock prevention, avoidance and detection mechanisms	K3 Apply
CO3	Apply various memory management techniques in Operating Systems to manage system resources efficiently.	K3 Apply
CO4	Apply various file management techniques in Operating Systems.	K3 Apply

Mapping of course outcomes with program outcomes and program specific outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	2	—	—	—	—	—	—	1	3	1
CO2	3	3	2	2	—	—	—	—	—	—	1	3	1
CO3	3	3	2	2	—	—	—	—	—	—	1	3	1
CO4	3	3	2	2	—	—	—	—	—	—	1	3	1
AVG	3	3	2	2	—	—	—	—	—	—	1	3	1

Mapping of course outcomes with Knowledge and Attitude Profile (WK)

	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	*	*	*	*	*			*	
CO2	*	*	*	*	*			*	
CO3	*	*	*	*	*			*	
CO4	*	*	*	*	*			*	

WKs Mapped with the course

WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
-----	-----	-----	-----	-----	-----	-----	-----	-----

*	*	*	*	*			*	
---	---	---	---	---	--	--	---	--

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	40	60	2.5 hrs

Continuous Internal Evaluation Pattern:

Attendance	: 5 marks
Continuous Assessment Test	: 20 marks
Continuous Assessment Assignment	: 15 marks

SYLLABUS

Module 1 (14 hrs)

Operating system overview – Functions and Services - Operating systems structures - Systems calls - Kernel approaches Building and booting an OS - Operating-System Debugging. Process Overview - Process states - Process control block - Context-switching - Processes operations - Process scheduling - CPU scheduling: Non-preemptive, preemptive - Multi-queue scheduling - Multi-level feedback queue scheduling.

Module 2 (15 hrs)

Inter Process Communication - Shared memory, message passing, examples of IPC systems. Race condition Critical section problem, Hardware support for synchronization, Mutex locks. Semaphores - Classical synchronization problems. Concept of a thread, Multithreading benefits, Multi-threading Models. Deadlocks - Resource allocation graphs Deadlock: prevention, avoidance, detection and recovery.

Module 3 (14 hrs)

Concept of address spaces, Swapping, Contiguous memory allocation, Address Translation and TLB. Segmentation – Address translation in segmentation, TLBs, TLB hits and misses, Handling TLB misses, TLB structure, Reducing the page table size. Virtual Memory - Paging, Demand paging, Copy-on-write, Page replacement algorithms, Thrashing, Allocating Kernel Memory.

Module 4 (13 hrs)

File concept - Attributes, Operations, types, structure – Access methods, Protection. File-system implementation, Directory implementation. Allocation methods. Hard disk: Geometry, Magnetic disks, Solid State Devices, Disk scheduling, Disk formatting. Modern System architecture, Programmed I/O, Interrupts, DMA, Device interaction methods, The Device Driver.

Text Books:

1. Silberschatz, A., Galvin, P. B., & Gagne, G. (2015). *Operating system concepts* (9th ed.). Wiley India.
2. Tanenbaum, A. S. (2015). *Modern operating systems* (4th ed.). Prentice Hall.
3. Arpaci-Dusseau, A. C., & Arpaci-Dusseau, R. H. (2018). *Operating systems: Three easy pieces* (1st ed.). CreateSpace.

Reference Books:

1. Stallings, W. (2015). Operating systems (6th ed., Global ed.). Pearson.
2. Nutt, G., Chaki, N., & Neogy, S. (n.d.). Operating systems (3rd ed.). Pearson Education.
3. Dhamdhere, D. M. (2011). Operating systems (2nd ed.). Tata McGraw Hill.

END SEMESTER EXAMINATION – MODEL QUESTION PAPER

Course Code	25UCST302
Course Name	OPERATING SYSTEMS
Max. Marks: 60	Duration: 2.5 Hours

Part A (Answer all questions. Each question carries 3 marks)

Qn No.	Question	Marks	CO	BL
1.	Explain process states and the transitions between them with a state transition diagram.	3	CO1	L2
2.	Compare preemptive and non-preemptive CPU scheduling. Apply Round Robin scheduling (time quantum = 2) to processes P1(burst=4), P2(burst=3), P3(burst=5) arriving at t=0.	3	CO1	L3
3.	Explain the Critical Section problem and the three conditions that must be satisfied by any solution.	3	CO2	L2
4.	Describe the Banker's Algorithm for deadlock avoidance with a suitable example.	3	CO2	L2
5.	Explain paging and segmentation. Compare their memory utilization and external fragmentation.	3	CO3	L2
6.	Apply the LRU page replacement algorithm to the reference string 7,0,1,2,0,3,0,4,2,3 with 3 page frames and compute the number of page faults.	3	CO3	L3
7.	Explain the various file allocation methods: contiguous, linked, and indexed allocation.	3	CO4	L2
8.	Describe the SCAN and C-SCAN disk scheduling algorithms. Apply SCAN on the request queue [98, 183, 37, 122, 14, 124, 65, 67] starting from head position 53.	3	CO4	L3

Part B

(Answer any one full question from each Module. Each carry 9 marks)

Module I

Qn No.	Question	Marks	CO	BL
9.1	Explain the different CPU scheduling algorithms: FCFS, SJF, Priority Scheduling, and Round Robin. Compare their performance using the following process table: P1(AT=0, BT=6), P2(AT=2, BT=4), P3(AT=4, BT=2), P4(AT=6,	5	CO1	L3

	BT=8).			
9.2	Describe the process control block (PCB) and context switching mechanism in detail.	4	CO1	L2
OR				
10.1	Illustrate multi-level feedback queue scheduling with a diagram. Discuss its advantages over multi-level queue scheduling.	5	CO1	L3
10.2	Describe the various types of OS structures: monolithic, layered, microkernel, and modular. Compare their advantages and disadvantages.	4	CO1	L2
Module II				
Qn No.	Question	Marks	CO	BL
11.1	Demonstrate semaphores and their use in solving classical synchronization problems: Producer-Consumer, Reader-Writer, and Dining Philosopher problems.	5	CO2	L3
11.2	Interprete the four necessary conditions for deadlock (Coffman conditions).	4	CO2	L2
OR				
12.1	Illustrate multithreading benefits and models: Many-to-One, One-to-One, and Many-to-Many. Discuss the differences between user threads and kernel threads.	5	CO2	L3
12.2	Describe the hardware support for synchronization: TestAndSet and Swap instructions. Explain how mutex locks are implemented.	4	CO2	L2
Module III				
Qn No.	Question	Marks	CO	BL
13.1	Describe virtual memory management with demand paging. Illustrate the working of page replacement algorithms: FIFO, Optimal, and LRU with the reference string 3,2,1,0,3,2,4,3,2,1,0,4 and 4 frames.	5	CO3	L3
13.2	Explain TLB (Translation Lookaside Buffer) and its role in address translation. Describe TLB hits and misses and methods to handle TLB misses.	4	CO3	L2
OR				
14.1	Illustrate segmentation with paging (combined). Describe address translation in segmentation and how it differs from pure paging.	5	CO3	L3
14.2	Explain thrashing and its causes. Describe the working set model and how it is used to prevent thrashing.	4	CO3	L2
Module IV				
Qn No.	Question	Marks	CO	BL

15.1	Illustrate the inode-based file system implementation	5	CO4	L3
15.2	Describe DMA (Direct Memory Access) and its role in I/O management. Explain the device driver architecture and interrupt handling.	4	CO4	L2
OR				
16.1	Apply FCFS, SCAN, and C-SCAN disk scheduling algorithms to the disk request queue [82, 170, 43, 140, 24, 16, 190] with head starting at 50. Calculate total head movement for each.	5	CO4	L3
16.2	Compare solid-state devices (SSD) with magnetic disk drives in terms of access time, seek time, and latency. Describe disk formatting and its types.	4	CO4	L2



25UAMT303	OBJECT ORIENTED PROGRAMMING IN PYTHON	Category	L	T	P	R	Credit	Year of Introduction
		PBL	3	0	0	1	4	2025

Preamble: This course introduces the principles of Object-Oriented Programming using Python. The course focuses on designing reusable software components using classes and objects and implementing encapsulation, abstraction, inheritance, and polymorphism. It enables students to develop modular and scalable applications using Python.

Prerequisite: Nil

Course Outcomes: After the completion of the course, the student will be able to:

CO	Course Outcomes	Knowledge Level
CO1	Explain Python programming fundamentals including control structures and data structures.	K2 Understand
CO2	Design and implement classes and objects applying encapsulation and abstraction.	K3 Apply
CO3	Apply inheritance and polymorphism to develop reusable applications	K3 Apply
CO4	Develop modular object-oriented applications using file handling, exception handling and packages.	K6 Create

Mapping of course outcomes with program outcomes and program specific outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	—	—	1	—	—	—	—	—	—	3	1
CO2	3	3	3	1	2	—	—	—	1	—	1	3	2
CO3	3	3	3	1	2	—	—	—	1	—	—	3	3
CO4	3	3	3	2	3	—	—	—	2	1	1	3	3
Avg	3	2.6	3	1.3	2	—	—	—	1.3	1	1	3	2.3

Mapping of course outcomes with Knowledge and Attitude Profile (WK)

	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	*	*	*	*		*		*	
CO2	*	*	*	*	*	*		*	
CO3	*	*	*	*	*	*			
CO4	*	*	*	*	*	*		*	

Wks Mapped with the course

WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
*	*	*	*	*	*		*	

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	60	40	2.5 hrs

Continuous Internal Evaluation Pattern:

Attendance: 5

Internal Test 1: 15

Internal Test 2: 15

Assignment/Tutorial/Industrial Visit: 10

Mini Project: 15

Total: 60

SYLLABUS

Module 1 (13 hrs)

Introduction to Python - Features of Python, Applications of Python, Installation process, Python Interpreter, IDEs. Variables and Data Types - Numeric types (int, float, complex), Boolean type, Type conversion and casting. Operators - Arithmetic, Relational, Logical, Assignment, Bitwise, Membership, Identity operators. Input and Output Statements - input(), print(), Formatting output, f-strings. Conditional Statements - if, if-else, nested if, elif ladder. Looping Statements - for loop, while loop, break, continue, pass. Strings - String indexing, slicing, string methods. Data Structures - Lists, Tuples, Sets, Dictionaries, Built-in functions. Functions - Defining functions, parameters, return values, default arguments, keyword arguments, recursion, lambda functions.

Module 2 (14 hrs)

Introduction to OOP - Procedural vs Object Oriented Programming, Characteristics of OOP. Classes and Objects - Class definition, Object creation, self keyword. Constructors and Destructor - __init__() method, __del__() method, Instance variables. Types of Methods - Instance methods, Class methods, Static methods. Encapsulation - Data hiding, Public, Private and Protected members. Abstraction - Abstract classes, abc module, Abstract methods. Practice Applications - Class-based applications (Student Management System / Bank Management System).

Module 3 (14 hrs)

Inheritance - Single, Multiple, Multilevel, Hierarchical inheritance. Method Overriding - Rules of overriding, super() function. Method Resolution Order (MRO) - Concept of MRO, C3 Linearization. Polymorphism - Duck typing, Operator overloading. Magic Methods - __str__(), __repr__(), __add__(), __len__(). Exception Handling - try, except, else, finally, Raising exceptions. User Defined Exceptions - Creating custom exception classes.

Module 4 (15 hrs)

File Handling - File modes, Reading and Writing text files. Working with Files - Binary files, with statement. CSV and JSON Handling - Reading and writing CSV files, JSON parsing and serialization. Modules - Creating modules, import statement, from-import. Packages - Creating packages, __init__.py, Package structure. Iterators - Iterables, __iter__(), __next__(). Generators - yield keyword, Generator functions. Mini Project - Design and implementation of an OOP-based real-world and sustainable application.

Completion of an industrial visit or industrial training is desirable for this course.

Text Books:

1. Reema Thareja, Python Programming: Using Problem Solving Approach, Oxford University Press, 3rd Edition (2025).
2. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, O'Reilly Media, 3rd Edition (2024).
3. Paul Barry, Head First Python: A Brain-Friendly Guide, O'Reilly Media, 3rd Edition (2023).

Reference Books:

1. Mark Lutz, *Learning Python*, O'Reilly Media, 6th Edition (2025).
2. Herbert Schildt, Python: *The Complete Reference*, McGraw Hill Education, 3/e (2018).
3. Wesley J. Chun, *Core Python Applications Programming*, Pearson Education, 3/e (2015).

END SEMESTER EXAMINATION – MODEL QUESTION PAPER

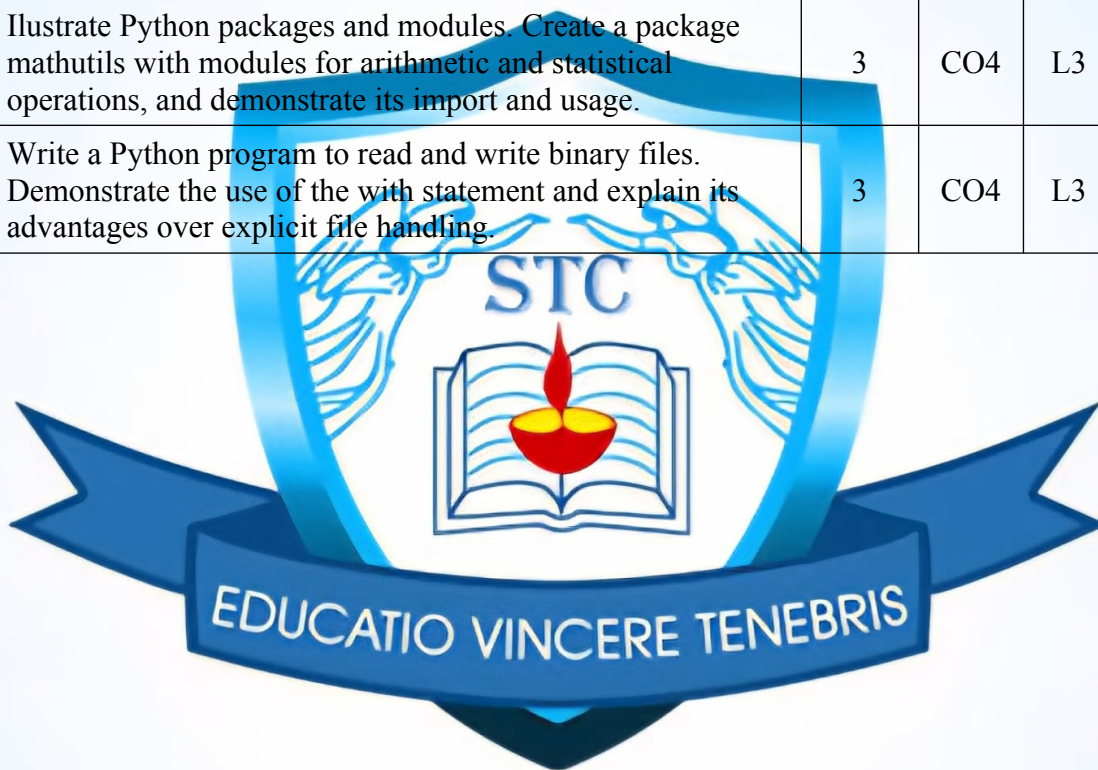
Course Code	25UAMT303
Course Name	OBJECT ORIENTED PROGRAMMING IN PYTHON
Max. Marks: 40	Duration: 2.5 Hours

Part A (Answer all questions. Each question carries 2 marks)				
Qn No.	Question	Marks	CO	BL
1.	Explain the different data types in Python. Write a program demonstrating type conversion between int, float, and string.	2	CO1	L3
2.	Write a Python program using recursion to compute the nth Fibonacci number. Trace the execution for n=5.	2	CO1	L3
3.	Explain encapsulation in Python. Illustrate a class BankAccount with private attributes and methods to deposit, withdraw, and display balance.	2	CO2	L3
4.	Differentiate instance methods, class methods, and static methods with suitable examples.	2	CO2	L2
5.	Explain Method Resolution Order (MRO) in Python. Trace the MRO for a class D inheriting from B and C, where both inherit from A (diamond problem).	2	CO3	L2
6.	Explain operator overloading in Python. Write a class Complex to overload the + operator for complex number addition.	2	CO3	L3
7.	Write a Python program to read a CSV file containing student records and display students who scored more than 75%.	2	CO4	L3
8.	Explain Python generators. Write a generator function to produce Fibonacci numbers up to n terms.	2	CO4	L3

Part B (Answer any one full question from each Module. Each carry 6 marks)	
Module I	

Qn No.	Question	Marks	CO	BL
9.1	Explain the built-in data structures of Python: List, Tuple, Set, and Dictionary. Compare their features and demonstrate at least three operations on each.	3	CO1	L2
9.2	Write a Python program using lambda functions and list comprehensions to filter even numbers from a list and compute their squares.	3	CO1	L3
OR				
10.1	Explain looping constructs in Python. Write a program to display a multiplication table for numbers 1 to 10 using nested loops.	3	CO1	L3
10.2	Write a Python function to sort a list of dictionaries by a specified key using default and keyword arguments.	3	CO1	L3
Module II				
Qn No.	Question	Marks	CO	BL
11.1	Illustrate a class hierarchy for a Student Management System using encapsulation and abstraction. Include attributes for name, roll number, marks, and methods for computing grade and displaying results.	3	CO2	L3
11.2	Explain abstract classes and abstract methods in Python using the abc module. Demonstrate with a class Shape having abstract methods area() and perimeter(), implemented by Rectangle and Circle.	3	CO2	L3
OR				
12.1	Explain constructors and destructors in Python. Write a program demonstrating __init__() and __del__() methods with a class Resource that manages a file resource.	3	CO2	L3
12.2	Demonstrate all types of methods (instance, class, and static) with a class Employee that tracks the employee count and computes annual salary.	3	CO2	L3
Module III				
Qn No.	Question	Marks	CO	BL
13.1	Explain all types of inheritance in Python with examples. Demonstrate multiple inheritance and resolve the diamond problem using MRO and super().	3	CO3	L3
13.2	Write a Python program demonstrating user-defined exceptions. Create a custom exception InvalidAgeError and use it in a function that validates age for voting eligibility.	3	CO3	L3
OR				
14.1	Explain duck typing and operator overloading in Python. Write a class Vector to overload +, -, and * (scalar	3	CO3	L3

	multiplication) operators.			
14.2	Explain magic methods in Python. Demonstrate the use of <code>__str__()</code> , <code>__repr__()</code> , <code>__add__()</code> , and <code>__len__()</code> with a suitable class.	3	CO3	L3
Module IV				
Qn No.	Question	Marks	CO	BL
15.1	Illustrate a mini project for a Library Management System using OOP concepts in Python. The system should support adding books, issuing and returning books, and maintaining member records using file handling with JSON.	3	CO4	L3
15.2	Explain iterators and generators in Python. Write a generator that lazily yields prime numbers up to a given limit.	3	CO4	L3
OR				
16.1	Illustrate Python packages and modules. Create a package <code>mathutils</code> with modules for arithmetic and statistical operations, and demonstrate its import and usage.	3	CO4	L3
16.2	Write a Python program to read and write binary files. Demonstrate the use of the <code>with</code> statement and explain its advantages over explicit file handling.	3	CO4	L3



25UCST304	SWITCHING THEORY AND LOGIC DESIGN	Category	L	T	P	R	Credit	Year of Introduction
		ESC	3	1	0	0	4	2025

Preamble: The course introduces the fundamental concepts of digital systems, including number systems, binary codes, Boolean algebra, and logic gates. It develops the ability to analyze and design combinational and sequential circuits using systematic methods such as minimization techniques, flip-flops, counters, and state machines. The course also provides an introductory understanding of programmable logic and Verilog HDL for describing basic digital circuits. Upon completion, students will be able to design and analyze simple digital systems and understand their application in computer organization and digital electronics.

Prerequisite: Nil

Course Outcomes: After the completion of the course, the student will be able to:

CO	Course Outcomes	Knowledge Level
CO1	Explain the fundamentals of digital systems, number systems, binary codes, Boolean algebra, and logic gates.	K2 Understand
CO2	Apply Boolean algebra and minimization techniques to design and implement combinational logic circuits.	K3 Apply
CO3	Apply the principles of flip-flops, finite state machines, and counters to analyze and design sequential logic circuits.	K3 Apply
CO4	Develop digital circuits such as shift registers, counters, and PLA-based designs, and describe them using basic Verilog HDL modeling concepts.	K3 Apply

Mapping of course outcomes with program outcomes and programme specific outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	1	1	1	—	—	—	—	—	—	1	1	—
CO2	3	3	3	2	—	—	—	—	—	—	1	1	—
CO3	3	3	3	2	—	—	—	—	—	—	1	1	—
CO4	3	3	3	2	2	—	—	—	—	—	2	1	—
Avg	3	2.5	2.5	1.75	2	—	—	—	—	—	1.25	1	—

Mapping of course outcomes with Knowledge and Attitude Profile (WK)

	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	*	*	*	*	*			*	
CO2	*	*	*	*	*			*	
CO3	*	*	*	*	*			*	
CO4	*	*	*	*	*	*		*	

Wks Mapped with the course

WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
*	*	*	*	*	*		*	

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	40	60	2.5 hrs

Continuous Internal Evaluation Pattern:

Attendance	: 5 marks
Continuous Assessment Test	: 20 marks
Continuous Assessment Assignment	: 15 marks

SYLLABUS

Module 1 (12 hrs)

Number Systems: Digital systems and binary numbers: number base conversions, octal/hexadecimal, complements, signed binary numbers. Binary Codes: BCD, Gray, Excess-3, and ASCII. Boolean Algebra: Basic definitions, postulates, and theorems; Boolean functions; standard and canonical forms (SOP and POS). Logic Gates: Standard gates (AND, OR, NOT, XOR, XNOR) and Universal gates (NAND, NOR); implementation of Boolean functions.

Module 2 (14 hrs)

Gate-Level Minimization: SOP/POS simplification using Karnaugh maps (up to 4 variables); don't care conditions; Quine-McCluskey (Tabulation) method for algorithmic simplification. Combinational Circuits: Design of Half/Full Adders and Subtractors; Parallel Adder; BCD Adder; Magnitude Comparator, Encoders, Decoders, Multiplexers, and Demultiplexers; Code converters.

Module 3 (15 hrs)

Flip-flops: SR, JK, T, and D flip-flops; level and edge triggering; truth tables, excitation tables, and characteristic equations; conversion between flip-flop types. Finite State Machines (FSM): Concept of State, Mealy and Moore Machines, Design of Sequence Detectors, State Diagrams and Tables. Counter design: Concept of counters; asynchronous (ripple) counters—binary up, binary down and BCD counters; synchronous counters—binary up, binary down, binary up/down counters, and BCD counters.

Module 4 (15 hrs)

Shift registers: Serial In Serial Out (SISO), Serial In Parallel Out (SIPO), Parallel In Parallel Out (PIPO), Parallel In Serial Out (PISO), bidirectional with parallel load. Specialized Counters: Ring counter, Johnson counter – timing sequences and state diagrams. Programmable logic: Programmable Logic Array (PLA), implementation of basic circuits using PLA. Verilog HDL: Introduction to HDL and basic concepts of Verilog; module structure, data types, operators, and modeling styles (gate-level, dataflow, behavioral). Design of simple combinational and sequential circuits using Verilog syntax; basic idea of test benches and simulation (theoretical only).

Text Books:

1. Mano, M. M. (2016). *Digital logic and computer design*. Pearson Education.

2. Floyd, T. L. (2017). *Digital fundamentals* (11th ed.). Pearson Education.
3. Palnitkar, S. (2003). *Verilog HDL: A guide to digital design and synthesis* (2nd ed.). Pearson.

Reference Books:

1. Mano, M. M., & Ciletti, M. D. (2018). *Digital design with an introduction to the Verilog HDL* (6th ed.). Pearson Education.
2. Givone, D. D. (2017). *Digital principles and design*. Tata McGraw Hill.
3. Wakerly, J. F. (2018). *Digital design: Principles and practices* (5th ed.). Pearson.



END SEMESTER EXAMINATION – MODEL QUESTION PAPER

Course Code	25UCST304
Course Name	SWITCHING THEORY AND LOGIC DESIGN
Max. Marks: 60	Duration: 2.5 Hours

Part A (Answer all questions. Each question carries 3 marks)				
Qn No.	Question	Marks	CO	BL
1.	Convert (245) ₁₀ to binary, octal, and hexadecimal. Also find the 1's and 2's complement of (10110101) ₂ .	3	CO1	L2
2.	Explain Gray code. Convert (13) ₁₀ to Gray code. Also convert (1011) _{Gray} to its decimal equivalent.	3	CO1	L2
3.	Simplify the Boolean function $F(A,B,C,D) = \sum m(0,1,2,5,8,9,10)$ using a Karnaugh map.	3	CO2	L3
4.	Design a full adder using basic logic gates. Also implement it using NAND gates only.	3	CO2	L3
5.	Explain the SR flip-flop with its truth table, characteristic equation, and excitation table. Convert it to a D flip-flop.	3	CO3	L2
6.	Design a 3-bit synchronous up counter. Draw the state diagram and state table.	3	CO3	L3
7.	Explain SISO and SIPO shift registers with timing diagrams. Give one application of each.	3	CO4	L2
8.	Write a Verilog behavioral model for a 2:1 multiplexer and describe its testbench.	3	CO4	L3

Part B (Answer any one full question from each Module. Each carry 9 marks)				
Module I				
Qn No.	Question	Marks	CO	BL
9.1	Explain Boolean algebra postulates and theorems. Prove De Morgan's theorems. Simplify: $F = ABC + A'BC + AB'C + AB'C'$ using Boolean algebra.	5	CO1	L3
9.2	Explain BCD, Excess-3, and ASCII codes. Convert (2025) ₁₀ to BCD and Excess-3. Also show BCD addition: 7 + 8.	4	CO1	L3
OR				
10.1	Implement the Boolean function $F(A,B,C) = \sum m(1,3,5,6,7)$ using: (i) AND-OR gates (SOP), (ii) OR-AND gates (POS), (iii) NAND gates only, and (iv) NOR gates only.	5	CO1	L3

10.2	Explain the working of NAND and NOR gates as universal gates. Implement a Full Subtractor using NAND gates only.	4	CO1	L3
Module II				
Qn No.	Question	Marks	CO	BL
11.1	Simplify $F(A,B,C,D) = \Sigma m(1,3,5,7,9,11,13,15)$ using the Quine-McCluskey (tabulation) method. Verify using a K-map.	5	CO2	L3
11.2	Design a 4-bit BCD adder. Show the correction logic needed when the sum exceeds 9.	4	CO2	L3
OR				
12.1	Design a 4-bit magnitude comparator that compares two 4-bit numbers A and B. Derive the logic expressions and draw the circuit.	5	CO2	L3
12.2	Implement a 4:1 multiplexer using basic gates. Use a 4:1 multiplexer to implement the Boolean function $F(A,B,C) = \Sigma m(0,2,5,7)$.	4	CO2	L3
Module III				
Qn No.	Question	Marks	CO	BL
13.1	Design a Mealy state machine for a sequence detector that detects the pattern '1011'. Draw the state diagram, state table, and obtain the next-state and output logic equations.	5	CO3	L3
13.2	Design a 4-bit synchronous BCD counter using JK flip-flops. Derive the flip-flop input equations and draw the logic circuit.	4	CO3	L3
OR				
14.1	Explain the conversion between flip-flop types. Convert a JK flip-flop to (i) D flip-flop and (ii) T flip-flop. Derive the conversion logic.	5	CO3	L3
14.2	Design a 3-bit asynchronous ripple counter. Explain its limitations compared to synchronous counters.	4	CO3	L3
Module IV				
Qn No.	Question	Marks	CO	BL
15.1	Explain the Johnson counter and Ring counter. Draw their timing diagrams and state sequences for a 4-bit configuration. Compare their modulus counts.	5	CO4	L2
15.2	Implement the Boolean functions $F1 = AB + AC$ and $F2 = A'B + BC$ using a PLA. Show the PLA truth table and draw the PLA structure.	4	CO4	L3
OR				
16.1	Explain Verilog HDL modeling styles: gate-level, dataflow,	5	CO4	L3

	and behavioral. Write Verilog models for a 4-bit binary adder using all three styles.			
16.2	Explain PIPO and PISO shift registers with diagrams and timing waveforms. Write a Verilog behavioral model for a 4-bit PISO shift register.	4	CO4	L3



25UHUT001	ECONOMICS FOR ENGINEERING PRACTICES	Category	L	T	P	R	Credit	Year of Introduction
		HMC	2	0	0	0	2	2025

Preamble: This syllabus covers the fundamentals of Microeconomics, Macroeconomics, and Engineering Economics. It bridges economic theory with practical engineering and management, teaching students to apply analytical tools to scarcity, market dynamics, and financial decision-making.

Prerequisite: Nil

Course Outcomes: After the completion of the course, the student will be able to:

CO	Course Outcomes	Knowledge Level
CO1	Understand the fundamentals of various economic issues using laws and learn the concepts of demand, supply, elasticity and production function.	K2 Understand
CO2	Develop decision making capability by applying concepts relating to costs and revenue, and acquire knowledge regarding the functioning of firms in different market situations.	K3 Apply
CO3	Outline the macroeconomic principles of monetary and fiscal systems and national income.	K2 Understand
CO4	Make use of the possibilities of value analysis and engineering, and take investment decisions through capital budgeting techniques.	K3 Apply

Mapping of course outcomes with program outcomes and program specific outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	0	0	-	2	2	-	-	-	-		
CO2	3	3	2	0	1	2	-	-	1	1	3		
CO3	2	2	0	0	-	3	3	1	-	-	2		
CO4	3	3	3	2	2	-	2	1	1	1	3		
Avg	2.75	2.5	2.5	2	1.5	2.3	2.3	1	1	1	2.6		

Mapping of course outcomes with Knowledge and Attitude Profile (WK)

	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	*	*	*	*	*		*		*
CO2	*	*	*	*	*	*	*	*	
CO3	*	*	*	*	*		*	*	*

CO4	*	*	*	*	*	*		*	*
-----	---	---	---	---	---	---	--	---	---

Wks Mapped with the course

WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
*	*	*	*	*	*	*	*	*

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	40	60	2.5 hrs

Continuous Internal Evaluation Pattern:

Attendance	: 5 marks
Continuous Assessment Test	: 20 marks
Continuous Assessment Assignment	: 15 marks

SYLLABUS

Module 1 (7 hrs)

Nature and significance of economics, Basic economic problems – Production Possibility Curve – Utility – Law of diminishing marginal utility – Demand: Factors determining demand – Law of Demand – Demand curve- Price elasticity of demand- measurement of price elasticity and its applications – Supply: factors determining supply - Law of supply – Supply curve- Equilibrium price determination- Changes in demand and supply and its effects on equilibrium price and quantity. Production. Production function, Factors of Production- Cobb-Douglas Production Function

Module 2 (7 hrs)

Cost: Cost concepts – Private cost and social cost – Sunk cost – Opportunity cost -Explicit and implicit cost –Short run cost curves –Long run average cost curve -Revenue concepts – Break-even point. Taxation-Direct and indirect taxes-principles of Taxation, GST.Market: Market: different market structures and market equilibrium – Kinked demand model- E-commerce, Stock market- SENSEX, NIFTY, Mutual Funds & ETFs (Concepts Only)

Module 3 (7 hrs)

National income: Concepts (GDP, GNP and NNP-Basic problems)– Final goods and Intermediate goods - Methods of Estimation –output method – expenditure method-- Difficulties in the measurement of national income. International trade, terms of Trade, Gain from International Trade, Free Trade vs. Protection. Dumping, and Balance of Payment. Inflation: Causes and Effects – Measures to Control Inflation - Monetary and Fiscal policies – Repo and reverse repo rate. Role of Science, Engineering and Technology In economic Development

Module 4 (7 hrs)

Value Analysis and value Engineering: Cost Value, Exchange Value, Use Value, Esteem Value - Aims, Advantages and Application areas of Value Engineering - Value Engineering

Procedure.Capital Budgeting: Time value of money - Net Present Value Method - Benefit Cost Ratio – Internal Rate of Return – Payback – Accounting Rate of Return.

Text Books:

1. Geetika, Piyali Ghosh and Chodhury, *Managerial Economics*, Tata McGraw Hill, 2015
2. H. G. Thuesen, W. J. Fabrycky, *Engineering Economy*, PHI, 1966.
3. R. Paneerselvam: *Engineering Economics*, PHI, 2012.
4. I M Pandey, *Financial Management*, Vikas Publishing House ,2015

Reference Books:

- 1 Leland Blank P.E, Anthony Tarquin P. E., *Engineering Economy* – Mc Graw Hill, 7th edition.
2. Khan M. Y, *Indian Financial System*, Tata McGraw Hill, 2011.
3. Donald G. Newman, Jerome P. Lavelle, *Engineering Economics and analysis*, Engg. Press, Texas, 2002.
4. Chan S. Park, *Contemporary Engineering Economics*, Prentice Hall of India Ltd, 2001.
5. Prasanna Chandra, *Financial Management: Theory and Practice*, Mc Graw Hill, 2007.

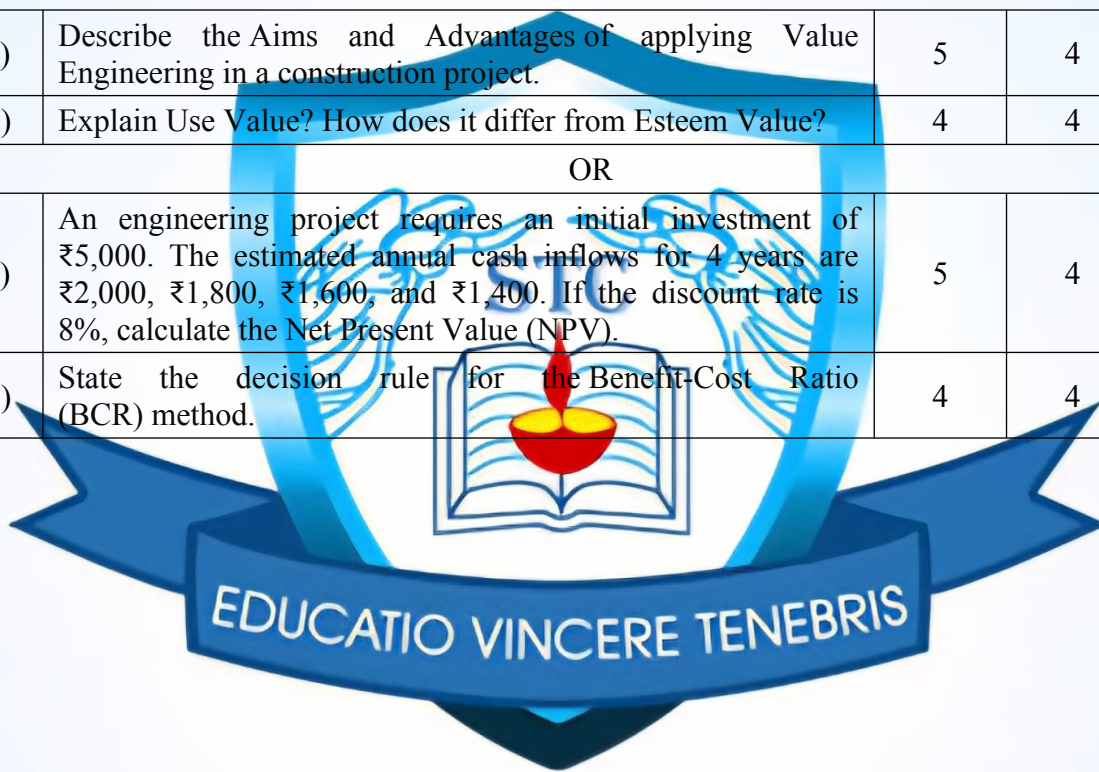


**END SEMESTER EXAMINATION - MODEL QUESTION PAPER
(STC 2025 SCHEME)**

Course Code	25UHUT001
Course Name	ECONOMICS FOR ENGINEERING PRACTICES
Max. Marks: 60	Duration: 2.5 Hours

Part A (Answer all questions. Each question carries 3 marks)				
Qn No.	Question	Marks	CO	BL
1	Explain is the Opportunity Cost principle	3	1	L2
2	Briefly explain the Cobb-Douglas Production Function	3	1	L2
3	Distinguish between Sunk Cost and Incremental Cost.	3	2	L2
4	Interpret the law of diminishing marginal utility using a real-world consumer scenario.	3	2	L3
5	Define Reverse Repo Rate and its impact on money supply.	3	3	L2
6	Define Exchange Value in Value Engineering?	3	3	L2
7	Explain the concept of Payback Period in project evaluation.	3	4	L2
8	Describe the financial meaning of the Internal Rate of Return (IRR) in capital budgeting decisions.	3	4	L3
Part B (Answer any one full question from each Module. Each carries 9 marks)				
Module 1				
Qn No.	Question	Marks	CO	
7(a)	Illustrate the Law of Diminishing Marginal Utility with a schedule and diagram.	5	1	L2
(b)	Explain the effect of an increase in Consumer Income on the equilibrium price and quantity of a normal good.	4	1	L2
OR				
8(a)	When the price of a machine part is ₹500, the supply is 100 units. When the price increases to ₹600, the supply rises to 150 units. Calculate the Price Elasticity of Supply and interpret the result.	5	1	L2
(b)	Discuss the three stages of the Law of Variable Proportions.	4	1	L2
MODULE-2				
9(a)	Compare the features of Perfect Competition and Monopoly markets.	5	2	L2
(b)	A firm has a Fixed Cost of ₹20,000 and a Variable Cost of ₹50 per unit. If the selling price is ₹100 per unit, calculate the Break-even Point in units.	4	2	L3
OR				
10(a)	Explain the Kinked Demand Curve model and why it leads to price rigidity in an Oligopoly.	5	2	L2

(b)	Differentiate between Private Cost and Social Cost using the example of industrial pollution.	4	2	L3
MODULE-3				
11(a)	Discuss the Difficulties in the Measurement of National Income in a developing economy.	5	3	L2
(b)	From the data given below estimate GNP _{mp} , GNP _{fc} and NNP _{mp} and National income. GDP _{mp} 5000(in 100 billion), NFIA = -50, indirect tax =70, subsidy =30, depreciation=30.	4	3	L2
OR				
12(a)	Explain Demand-Pull Inflation? Suggest two Fiscal Policy measures to control it.	5	3	L2
(b)	Explain the difference between Final Goods and Intermediate Goods with examples.	4	3	L2
MODULE-4				
13(a)	Describe the Aims and Advantages of applying Value Engineering in a construction project.	5	4	L3
(b)	Explain Use Value? How does it differ from Esteem Value?	4	4	L2
OR				
14(a)	An engineering project requires an initial investment of ₹5,000. The estimated annual cash inflows for 4 years are ₹2,000, ₹1,800, ₹1,600, and ₹1,400. If the discount rate is 8%, calculate the Net Present Value (NPV).	5	4	L3
(b)	State the decision rule for the Benefit-Cost Ratio (BCR) method.	4	4	L2



25UHUT002	SUSTAINABILITY	Category	L	T	P	R	Credit	Year of Introduction
------------------	-----------------------	----------	---	---	---	---	--------	----------------------

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	40	60	2.5 hrs

Continuous Internal Evaluation Pattern:

Attendance	: 5 marks
Continuous Assessment Test	: 20 marks
Continuous Assessment Assignment	: 15 marks

SYLLABUS**Module 1 (7 hrs)**

Fundamentals of ethics -Moral Relativism vs. Objectivity. Consequentialism. Deontology Metaethics. Personal vs. professional ethics, Civic Virtue, Respect for others, Profession and Professionalism, Plagiarism, case studies, Technology and digital revolution, Cybertrust and cybersecurity, High technologies: connecting people and places- Managing conflict, Confidentiality, Codes of Ethics. Basic concepts in Gender Studies, gender, gender spectrum: gender stereotypes, Gender disparity and discrimination in education, Ethical values and practices in connection with gender Gender policy and women/transgender empowerment initiatives.

Module 2 (7 hrs)

Introduction to Environmental Ethics: key philosophical theories (anthropocentrism, biocentrism, ecocentrism). Deep Ecology. Intrinsic value. Sustainable Engineering Principles: Definition and scope, triple bottom line, life cycle analysis and sustainability metrics. Ecosystems and Biodiversity: An overview of various ecosystems in Kerala/India, and its significance. Landscape and Urban Ecology: Principles of landscape ecology, Urbanization and its environmental impact, Moral Extensionism

Module 3 (7 hrs)

Hydrology and Water Management: water cycle, Environmental flow, disruptions and disasters. Zero Waste Concepts and Practices: Definition of zero waste and its principles, Strategies for waste reduction, Case studies of successful zero waste initiatives. Circular Economy and Degrowth: Differences between linear and circular economies, AI-Driven Energy Optimization, Digital Twins, Bio-Engineering & Biomimicry, Carbon Capture and Storage (CCS), Mobility and Sustainable Transportation: Basic tenets of a Sustainable Transportation design, Integrated mobility systems, E-Mobility.

Module 4 (7 hrs)

Renewable Energy and Sustainable Technologies: (solar, wind, hydro, biomass), Challenges and opportunities in renewable energy adoption. Climate Change and Engineering Solutions: Kerala/India and the Climate crisis, Engineering solutions to mitigate, adapt and build resilience to climate change. Environmental Policies and Regulations: Ethical considerations in environmental policy-making. Case Studies and Future Directions: Analysis of real-world case studies, Emerging trends and future directions in environmental ethics and sustainability, Discussion on the role of engineers in promoting a sustainable future. The Sustainable Development Goal 17.

Text Books:

1. Whitbeck, C. (2011). Ethics in engineering practice and research (2nd ed.). Cambridge University Press.
2. Oakley, J. (2006). Virtue ethics and professional roles. Cambridge University Press.
3. Bakshi, B. R. (2019). Sustainable engineering: Principles and practice. Cambridge University Press.

4. Martin, M. W., & Schinzinger, R. (2014). Ethics in engineering (4th ed.). Tata McGraw-Hill Education.
5. Ahlawat, A. (2019). Sustainable development goals: Directive principles for sustainable India by 2030. Notion Press.

Reference Books:

1. Parmar, R. K. (2025). Green technology and sustainable development (1st ed.). ISBN-13 publication.
2. Kosanam, N. (2023). Beyond carbon: Responsible ways to restore ecological balance. ISBN-13 publication.



END SEMESTER EXAMINATION – MODEL QUESTION PAPER

Course Code	25UHUT002
Course Name	SUSTAINABILITY AND ETHICS IN ENGINEERING
Max. Marks: 60	Duration: 2.5 Hours

Part A (Answer all questions. Each question carries 3 marks)				
Qn No.	Question	Marks	CO	BL
1.	Differentiate between Moral Relativism and Objectivity in the context of professional decision-making.	3	1	L2
2.	Outline the key differences between personal ethics and	3	1	L2

	professional ethics for a practicing engineer.			
3.	Explain the concept of 'Intrinsic Value' within environmental philosophy frameworks.	3	2	L2
4.	Summarise the three core pillars that constitute the 'Triple Bottom Line' of sustainability	3	2	L2
5.	Describe the engineering definition and operational importance of maintaining 'Environmental Flow' in river systems.	3	3	L2
6.	Differentiate between a Linear Economy and a Circular Economy using a conceptual pipeline breakdown.	3	3	L2
7.	Describe the challenges faced by coastal infrastructure in Kerala due to the ongoing climate crisis.	3	4	L2
8.	Outline the overarching objective of Sustainable Development Goal 17 regarding global partnerships.	3	4	L2
Part B (Answer any one full question from each Module. Each carry 9 marks)				
Module I				
Qn No.	Question	Marks	CO	BL
9	Describe the key principles of Consequentialism and Deontology. Explain their relevance in assessing a workplace confidentiality conflict in engineering practice.	9	1	L2
OR				
10	Explain the role of gender policies and empowerment initiatives in addressing structural discrimination within educational institutions.	9	1	L2
Module II				
Qn No.	Question	Marks	CO	BL
11	Describe the key environmental philosophical theories of Anthropocentrism, Biocentrism, and Ecocentrism and their influence on modern industrial development policies.	9	2	L2
OR				
12	Explain the key phases of Life Cycle Analysis (LCA) and the sustainability indicators used to evaluate the environmental performance of engineering products.	9	2	L2
Module III				
Qn No.	Question	Marks	CO	BL
13.a	A town experiences frequent water shortages during summer. Apply the concepts of the water cycle and environmental flow to suggest suitable water management measures.	5	3	L3
13.b	Describe the concept of Carbon Capture and Storage (CCS) as a climate mitigation strategy. .	4	3	L2
OR				
14.a	A residential colony generates a large amount of household waste. Apply the principles of Zero Waste to recommend two strategies for reducing waste generation.	5	3	L3

14.b	Explain the concept of AI-driven energy optimization and its significance in sustainable development.	4	3	L2
Module IV				
Qn No.	Question	Marks	CO	BL
15.a	A coastal community is vulnerable to sea-level rise and flooding. Apply engineering solutions to improve its climate resilience.	5	4	L3
15.b	Summarize the opportunities and challenges associated with renewable energy adoption in India.	4	4	L2
OR				
16.a	A proposed industrial project is located near a sensitive ecosystem. Apply environmental policy and ethical principles to identify key considerations before project approval.	5	4	L3
16.b	Explain the importance of emerging sustainable technologies in achieving environmental sustainability.	4	4	L2



25UCSL305	DATA STRUCTURES LAB	Category	L	T	P	R	Credit	Year of Introduction
		PCL	0	0	3	0	2	2025

Preamble: The Data Structures Lab provides hands-on experience in implementing and analyzing fundamental data structures and algorithms for efficient problem solving. The course emphasizes logical reasoning, algorithmic thinking, and performance analysis, supporting the development of core computing and programming skills.

Prerequisite: PIC204 Programming in C

Course Outcomes: After the completion of the course, the student will be able to:

CO	Course Outcomes	Knowledge Level
CO1	Apply linear data structures such as arrays, stacks, queues, and linked	K3 Apply

	lists to perform fundamental operations and solve basic computational problems.	
CO2	Apply tree and graph data structures to perform traversal, insertion, deletion, and search operations.	K3 Apply
CO3	Apply searching and sorting algorithms to solve computational problems and determine their time and space requirements.	K3 Apply
CO4	Apply appropriate data structure to solve problems considering memory utilization and algorithmic performance.	K3 Apply

Mapping of course outcomes with program outcomes and program specific outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	–	2	–	–	1	–	–	2	3	2
CO2	3	3	2	2	2	–	–	2	–	–	2	3	2
CO3	3	3	3	2	2	–	–	2	–	–	2	3	3
CO4	3	3	3	3	3	–	–	3	–	–	3	3	3
Avg	3	3	2.5	2.33	2.25	–	–	2	–	–	2.25	3	2.5

Mapping of course outcomes with Knowledge and Attitude Profile (WK)

	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	*	*	*	*	*				
CO2	*	*	*	*	*				
CO3	*	*	*	*	*	*			
CO4	*	*	*	*	*	*			

Wks Mapped with the course

WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
*	*	*	*	*	*			

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	50	50	3 hrs

Continuous Internal Evaluation Pattern:

Attendance: 5 | Preparation/Pre-Lab Work, Viva, Timely completion of Lab Reports/Record (Continuous Assessment): 25 | Internal Examination: 20 | Total: 50

End Semester Examination (ESE): Procedure/Algorithm: 10 | Conduct/Programming: 15 | Result/Output: 10 | Viva voce: 10 | Record: 5 | Total: 50

Note: Students shall be allowed to attend end semester examination only upon submitting the duly certified record.

List of Experiments:

1. Represent polynomials using array-based data structures and perform the addition of two given polynomials. Display the input and output polynomials in standard mathematical notation.*
 2. Represent a sparse matrix using tuple (triplet) representation. Perform transpose operation on the sparse matrix and compute the sum of two sparse matrices, displaying the results clearly.
 3. Design and demonstrate stack operations using array representation, including push, pop, peek, and display, along with handling overflow and underflow conditions.*
 4. Design and demonstrate a linear queue using array representation. Perform insertion and deletion operations and display the status of the queue after each operation.*
 5. Design and demonstrate a circular queue using array representation and explain how it efficiently utilizes memory compared to a linear queue.**
 6. Convert a given infix arithmetic expression into postfix form using stack operations and explain the conversion process.*
 7. Evaluate a given postfix expression using stack representation and display the final computed result.*
 8. Apply linear search and binary search techniques on a dataset and analyze their performance characteristics.*
 9. Design a singly linked list and perform fundamental operations such as creation, insertion at various positions, deletion, and traversal of nodes.*
 10. Implement stack and queue data structures using singly linked list representation and demonstrate their operations.
 11. Represent polynomials using linked list structures and perform addition of two polynomials, displaying the resulting polynomial in standard form.
 12. Design a doubly linked list and demonstrate traversal of nodes in both forward and backward directions.*
 13. Design a circular linked list and perform insertion and deletion operations, demonstrating the circular nature of the structure.*
 14. Represent binary trees using suitable data structures and perform inorder, preorder, and postorder traversal operations.*
 15. Design a Binary Search Tree (BST) and perform insertion, searching, deletion, and traversal operations.*
 16. Represent a graph using adjacency matrix or adjacency list representation and perform Depth First Search (DFS) traversal starting from a given vertex.*
 17. Implementation and comparison of sorting techniques: Selection Sort, Insertion Sort, and Bubble Sort.*
 18. Implementation of Quick Sort algorithms.*
 19. Implementation of Merge Sort algorithms.*
- * Indicate the mandatory programs

Text Books:

1. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). Introduction to algorithms (4th ed.). MIT Press.
2. Thareja, R. (2021). Data structures using C (3rd ed.). Oxford University Press.
3. Horowitz, E., Sahni, S., & Anderson-Freed, S. (2007). Fundamentals of data structures in C (2nd ed.). Universities Press.

Reference Books:

1. Samanta, D. (2018). Classic data structures (2nd ed.). Prentice Hall India.

2. Gilberg, R. F., & Forouzan, B. A. (n.d.). Data structures: A pseudocode approach with C (2nd ed.). Cengage Learning.
3. Tremblay, J. P., & Sorenson, P. G. (2017). Introduction to data structures with applications (2nd ed.). Tata McGraw Hill.



25UCSL306	OPERATING SYSTEMS LAB	Category	L	T	P	R	Credit	Year of Introduction
		PCL	0	0	3	0	2	2025

Preamble: Operating Systems Lab provides practical exposure to core operating system concepts through hands-on experiments and programming exercises. Students implement and analyze algorithms related to process scheduling, memory management, file systems, synchronization, and inter-process communication. The lab strengthens system-level programming skills and helps learners understand how operating systems manage resources and processes in real-world environments.

Prerequisite: PIC204 Programming In C

Course Outcomes: After the completion of the course, the student will be able to:

CO	Course Outcomes	Knowledge Level
CO1	Apply Linux command-line utilities and shell scripting to perform file management, data extraction, sorting, filtering, system information retrieval, arithmetic operations, file handling, and string manipulation.	K3 Apply

CO2	Implement and demonstrate process creation, parallel processing, and inter-process communication using system calls, shared memory, and semaphores.	K3 Apply
CO3	Implement and execute CPU scheduling, deadlock avoidance, and page replacement algorithms using various methods.	K3 Apply
CO4	Simulate and compute disk scheduling algorithms using FCFS, SCAN, and C-SCAN and determine the head movement metrics.	K3 Apply

Mapping of course outcomes with program outcomes and programme specific outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	1	3	–	–	2	–	–	–	3	2
CO2	3	3	2	2	3	–	–	2	–	–	–	3	2
CO3	3	3	3	2	2	–	–	2	–	–	–	3	3
CO4	3	3	3	3	2	–	–	2	–	–	–	3	3
Avg	3	2.75	2.5	2	2.5	–	–	2	–	–	–	3	2.5

Mapping of course outcomes with Knowledge and Attitude Profile (WK)

	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	*	*	*	*	*	*	*	*	
CO2	*	*	*	*	*	*	*	*	
CO3	*	*	*	*	*	*	*	*	
CO4	*	*	*	*	*	*	*	*	

WKs Mapped with the course

WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
*	*	*	*	*	*	*	*	

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	50	50	3 hrs

Continuous Internal Evaluation Pattern:

Attendance: 5 | Preparation/Pre-Lab Work, Viva, Timely completion of Lab Reports/Record: 25 | Internal Examination: 20 | Total: 50

End Semester Examination (ESE): Procedure/Algorithm: 10 | Conduct/Programming: 15 | Result/Output: 10 | Viva voce: 10 | Record: 5 | Total: 50

Note: Students shall be allowed for the end semester examination only upon submitting the duly certified record.

List of Experiments:

1. Working with Linux Commands for File Management, Data Extraction, Sorting, Filtering, and Basic Terminal Navigation
2. Shell programming for System Information, Simple Arithmetic Operations, File Handling, and String Manipulation
3. Implementation of System Calls for Parallel Processing using fork, wait (join), exec, getpid, and pipe

SEMESTER III		SUSTAINABLE DEVELOPMENT GOALS																
4. Implementation of CPU Scheduling Algorithms: (1) First Come First Served (FCFS), (2) Shortest Job First (SJF), (3) Round Robin, (4) Priority Scheduling																		
5. Implementation of Inter-Process Communication (IPC) using:	Course Code	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Semaphores																		
6. Implementation of Banker's Algorithm for Deadlock Avoidance	25UCST301																	
7. Implementation of Page Replacement Algorithms: FIFO, Optimal, LRU																		
8. Implementation of Disk Scheduling Algorithms: FCFS, SCAN, C-SCAN	25UCST302																	
Text Books:	25UAMT303																	
1. Silberschatz, A., Galvin, P. B., & Gagne, G. (2019). Operating system concepts (10th ed.). John Wiley & Sons.	25UCST304																	
2. Tanenbaum, A. S., & Bos, H. (2022). Modern operating systems (5th ed.). Pearson.																		
3. Stallings, W. (2019). Operating systems: Internal & design principles (9th ed.). Pearson.	25UCSL305																	
Reference Books:																		
1. Crowley, C. (2017). Operating systems: A design-oriented approach. McGraw Hill Education.																		





MINORS & HONOURS



25MAMT301	INTRODUCTION TO INTELLIGENT SYSTEMS	Category	L	T	P	R	Credit	Year of Introduction
		VAC	4	0	0	0	4	2025

EDUCATIO VINCERE TENEBRIS

Preamble: This course introduces the fundamental concepts, models, and techniques used in intelligent systems. It provides a foundation for understanding how machines perceive, reason, learn, and act intelligently. The course bridges classical artificial intelligence with modern intelligent system applications.

Prerequisite: Nil

Course Outcomes: After the completion of the course, the student will be able to:

CO	Course Outcomes	Knowledge Level
CO1	Explain the fundamentals, characteristics, and components of intelligent systems.	K2 Understand
CO2	Apply suitable knowledge representation techniques and reasoning methods to solve problems.	K3 Apply
CO3	Identify different learning paradigms used in intelligent systems.	K2 Understand

CO4	Develop basic intelligent system architectures for real-world applications.	K3 Apply
------------	---	----------

Mapping of course outcomes with program outcomes and program specific outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	1	1	2	-	-	-	-	-	-	3	-	-
CO2	3	3	3	2	-	-	-	-	-	-	3	-	-
CO3	3	2	2	2	-	-	-	-	-	-	3	-	-
CO4	3	3	3	2	-	-	-	-	-	-	3	-	-
Avg	3.00	2.25	2.25	2.00	-	-	-	-	-	-	3.00	-	-

Mapping of course outcomes with Knowledge and Attitude Profile (WK)

	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	*	*	*	*	*			*	
CO2	*	*	*	*	*			*	
CO3	*	*	*	*	*			*	
CO4	*	*	*	*	*			*	

Wks Mapped with the course

WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
*	*	*	*	*			*	

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	40	60	2.5 hrs

Continuous Internal Evaluation Pattern:

Attendance	: 5 marks
Continuous Assessment Test	: 20 marks
Continuous Assessment Assignment	: 10 marks

SYLLABUS

Module 1 (12 hrs)

Definition and characteristics of intelligent systems, Evolution of intelligent systems, Artificial Intelligence vs Intelligent Systems, Components of an intelligent system, Types of intelligent agents. Applications of intelligent systems in real-world domains.

Module 2 (15 hrs)

Knowledge representation techniques, Logic-based representation, Rule-based systems, Semantic networks, Frames. Reasoning methods: Deductive and inductive reasoning, Forward and backward chaining. Inference mechanisms.

Module 3 (15 hrs)

Concept of learning as knowledge acquisition and performance improvement in intelligent systems. Learning as an optimization and generalization process. Adaptation mechanisms in intelligent agents. Relationship between learning, environment, and agent behavior. Feedback-driven adaptation and system self-improvement.

Module 4 (14 hrs)

Architectural principles of intelligent systems: modularity, autonomy, scalability, and adaptability. Agent-based systems: autonomous agents, perception-action cycle, coordination, cooperation, and negotiation mechanisms. Expert systems: knowledge acquisition, inference engines, explanation facilities, and rule-based decision reasoning. Hybrid intelligent systems: integration of symbolic reasoning with data-driven learning; motivation for hybridization and system-level advantages.

Text Books:

1. Russell, S., & Norvig, P. (2021). *Artificial intelligence: A modern approach* (4th ed.). Pearson.
2. Negnevitsky, M. (2011). *Artificial intelligence: A guide to intelligent systems* (3rd ed.). Pearson Education.
3. Nilsson, N. J. (1998). *Principles of artificial intelligence*. Morgan Kaufmann.

Reference Books:

1. Rich, E., & Knight, K. (2009). *Artificial intelligence* (3rd ed.). Tata McGraw-Hill.
2. Luger, G. F. (2009). *Artificial intelligence: Structures and strategies for complex problem solving* (6th ed.). Pearson Education.
3. Mitchell, T. M. (1997). *Machine learning*. McGraw-Hill.



END SEMESTER EXAMINATION – MODEL QUESTION PAPER

Course Code	25MAMT301
Course Name	INTRODUCTION TO INTELLIGENT SYSTEMS
Max. Marks: 60	Duration: 2.5 Hours

Part A (Answer all questions. Each question carries 3 marks)				
Qn No.	Question	Marks	CO	BL
1.	Define intelligent systems and explain their characteristics.	3	CO1	L2
2.	Differentiate Artificial Intelligence and Intelligent Systems with examples.	3	CO1	L2
3.	Explain logic-based knowledge representation with an example.	3	CO2	L2
4.	Describe forward chaining and backward chaining with an example.	3	CO2	L2
5.	Explain the concept of learning as knowledge acquisition in intelligent systems.	3	CO3	L2
6.	Describe feedback-driven adaptation in intelligent agents with an example.	3	CO3	L2
7.	Explain the architectural principles of intelligent systems.	3	CO4	L2
8.	Describe the components and working of an expert system.	3	CO4	L2

Part B (Answer any one full question from each Module. Each carry 9 marks)				
Qn No.	Question	Marks	CO	BL
Module I				
9.a	Explain the definition, characteristics, and types of intelligent agents in detail.	5	CO1	L2
9.b	Describe the evolution of intelligent systems and their real-world applications.	4	CO1	L2
OR				
10	Compare artificial intelligence and intelligent systems. Explain the components of an intelligent system and types of intelligent agents.	9	CO1	L2
Module II				
11.a	Explain semantic networks and frames as knowledge representation techniques with examples.	5	CO2	L2
11.b	Describe deductive and inductive reasoning and explain	4	CO2	L2

	inference mechanisms.			
OR				
12	Explain rule-based systems and production rules. Apply forward and backward chaining to solve a given problem.	9	CO2	L3
Module III				
13	Explain learning as an optimization and generalization process. Describe adaptation mechanisms in intelligent agents, feedback loops, and self-improving systems.	9	CO3	L3
OR				
14.a	Describe the relationship between learning, environment, and agent behavior with examples.	5	CO3	L3
14.b	Explain knowledge revision and belief updating in intelligent systems.	4	CO3	L2
Module IV				
15.a	Explain agent-based systems: autonomous agents, perception-action cycle, coordination, and cooperation mechanisms.	5	CO4	L2
15.b	Describe hybrid intelligent systems and explain the motivation for integrating symbolic reasoning with data-driven learning.	4	CO4	L2
OR				
16	Explain expert systems: knowledge acquisition, inference engines, explanation facilities, and rule-based decision reasoning. Develop a basic architecture for an intelligent system for a real-world application.	9	CO4	L3

EDUCATIO VINCERE TENEBRIS

25MAMT302	PYTHON PROGRAMMING ESSENTIALS	Category	L	T	P	R	Credit	Year of Introduction
		VAC	4	0	0	0	4	2025

Preamble: This course provides a comprehensive introduction to Python programming, covering fundamental programming constructs, data structures, object-oriented concepts, file handling, and essential Python libraries. The course emphasizes both theoretical understanding and hands-on implementation to develop problem-solving skills and real-world application development capabilities.

Prerequisite: Nil

Course Outcomes: After the completion of the course, the student will be able to:

CO	Course Outcomes	Knowledge Level
CO1	Apply programming fundamentals such as control structures, functions, and data structures to develop Python applications.	K3 Apply
CO2	Implement data structures and functions to solve computational problems.	K3 Apply
CO3	Apply object-oriented concepts to develop modular Python applications.	K3 Apply
CO4	Apply file handling techniques and Python libraries to develop real-world applications.	K3 Apply

Mapping of course outcomes with program outcomes and program specific outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	-	1	-	-	-	-	-	1	2	1
CO2	3	3	2	1	1	-	-	-	-	-	1	2	1
CO3	3	3	3	1	1	-	-	-	-	-	1	2	1
CO4	2	3	3	2	1	-	-	-	-	-	1	2	1
Avg	2.8	2.8	2.3	1.3	1.0	-	-	-	-	-	1	2.0	1.0

Mapping of course outcomes with Knowledge and Attitude Profile (WK)

	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	*	*	*	*	*	*		*	
CO2	*	*	*	*	*	*		*	
CO3	*	*	*	*	*	*		*	
CO4	*	*	*	*	*	*		*	

Wks Mapped with the course

WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
*	*	*	*	*	*		*	

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	40	60	2.5 hrs

Continuous Internal Evaluation Pattern:

Attendance	: 5 marks
Continuous Assessment Test	: 20 marks
Continuous Assessment Assignment	: 15 marks

SYLLABUS

Module 1 (15 Hrs)

Introduction to Programming and Python. Features of Python, Installation and IDEs. Python Syntax and Structure. Variables, Data Types (int, float, string, Boolean). Operators and Expressions. Input and Output Statements. Control Statements: Conditional Statements (if, elif, else), Looping Statements (for, while), Break, Continue, Pass.

Module 2 (13 Hrs)

Strings and String Operations. Lists and List Methods. Tuples and Sets. Dictionaries and Dictionary Operations. Functions: Function Definition and Calling, Parameters and Return Values, Default and Keyword Arguments, Recursive Functions, Lambda Functions.

Module 3 (15 Hrs)

Introduction to OOP Concepts, Classes and Objects, Constructors and Destructors, Instance and Class Variables, Methods and Types of Methods, Inheritance, Polymorphism, Encapsulation, Exception Handling, Modules and Packages.

Module 4 (13 Hrs)

File Handling (Read, Write, Append). Working with CSV and JSON Files. Introduction to Python Libraries. Basic GUI using Tkinter (Introduction). Introduction to NumPy and Pandas. Mini Project Development.

Text Books:

1. Matthes, E. (2023). *Python crash course: A hands-on, project-based introduction to programming* (3rd ed.). No Starch Press.
2. Lutz, M. (2013). *Learning Python* (5th ed.). O'Reilly Media.
3. Sweigart, A. (2019). *Automate the boring stuff with Python* (2nd ed.). No Starch Press.

Reference Books:

4. Beazley, D. M., & Jones, B. K. (2013). *Python cookbook* (3rd ed.). O'Reilly Media.
5. Ramalho, L. (2022). *Fluent Python* (2nd ed.). O'Reilly Media.
6. McKinney, W. (2022). *Python for data analysis* (3rd ed.). O'Reilly Media.



END SEMESTER EXAMINATION – MODEL QUESTION PAPER

Course Code	25MAMT302
Course Name	PYTHON PROGRAMMING ESSENTIALS
Max. Marks: 60	Duration: 2.5 Hours

Part A (Answer all questions. Each question carries 3 marks)

Qn No.	Question	Marks	CO	BL
1.	Explain the features of Python and its advantages over other programming languages.	3	CO1	L2
2.	Write a Python program to find the factorial of a number using recursive functions.	3	CO1	L3
3.	Explain Python list methods with examples.	3	CO2	L2
4.	Write a Python program using lambda functions and dictionaries to count word frequency in a string.	3	CO2	L3
5.	Explain the concept of inheritance in Python with an example.	3	CO3	L2
6.	Describe exception handling in Python with try, except, else, and finally blocks.	3	CO3	L2
7.	Write a Python program to read and write data to a CSV file.	3	CO4	L3
8.	Explain the use of NumPy arrays with basic operations.	3	CO4	L2

Part B (Answer any one full question from each Module. Each carry 9 marks)

Qn No.	Question	Marks	CO	BL
Module I				
9.a	Explain Python data types, operators, and expressions with examples. Describe conditional and looping statements.	5	CO1	L2
9.b	Write a Python program implementing control statements (if-elif-else, for, while, break, continue, pass) to solve a given problem.	4	CO1	L3
OR				
10	Explain functions in Python: definition, calling, parameters, return values, default/keyword arguments, recursive functions, and lambda functions with examples.	9	CO1	L2
Module II				
11.a	Explain Python strings and string operations with examples. Describe tuples and sets with their methods.	5	CO2	L2
11.b	Write a Python program using dictionaries to implement a phonebook application with add, search, and delete	4	CO2	L3

	operations.			
OR				
12	Explain Python lists and list methods in detail. Write a Python program to implement a student grade management system using lists and functions.	9	CO2	L3
Module III				
13	Explain OOP concepts in Python: classes, objects, constructors, destructors, instance/class variables, inheritance, polymorphism, and encapsulation with examples.	9	CO3	L2
OR				
14.a	Explain different types of methods in Python: instance methods, class methods, and static methods with examples.	5	CO3	L2
14.b	Describe modules and packages in Python. Write a Python program using exception handling.	4	CO3	L3
Module IV				
15.a	Explain file handling in Python: read, write, and append operations. Describe working with CSV and JSON files.	5	CO4	L2
15.b	Explain the Pandas library with DataFrame operations and demonstrate its use with an example dataset.	4	CO4	L3
OR				
16	Develop a mini project in Python demonstrating file handling, Python libraries (NumPy/Pandas), and GUI using Tkinter. Describe the design and implementation.	9	CO4	L3



