



MVJCE CURRICULUM

FOR

ARTIFICIAL INTELLIGENCE &
MACHINE
LEARNING(Scheme 2022)

III SEMESTER

Mathematics for Computer Science		Semester	3
Course Code	MVJ22AI31	CIE Marks	50
Teaching Hours/Week (L: T:P: S)	2:2:0:0	SEE Marks	50
Total Hours of Pedagogy	30 hours Theory + 10 Hours Tutorial	Total Marks	100
Credits	03	Exam Hours	3
Examination type (SEE)	Theory		
<p>Course objectives: This course will enable the students to:</p> <ol style="list-style-type: none"> 1. To introduce the concept of random variables, probability distributions, specific discrete and continuous distributions with practical application in Computer Science Engineering and social life situations. 2. To Provide the principles of statistical inferences and the basics of hypothesis testing with emphasis on some commonly encountered hypotheses. 3. To Determine whether an input has a statistically significant effect on the system's response through ANOVA testing. 			
<p>Teaching-Learning Process Pedagogy (General Instructions): Teachers can use the following strategies to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the delivered lessons shall develop students' theoretical and applied Mathematical skills. 2. State the need for Mathematics with Engineering Studies and Provide real-life examples. 3. Support and guide the students for self-study. 4. You will assign homework, grading assignments and quizzes, and documenting students' progress. 5. Encourage the students to group learning to improve their creative and analytical skills. 6. Show short, related video lectures in the following ways: <ul style="list-style-type: none"> • As an introduction to new topics (pre-lecture activity). • As a revision of topics (post-lecture activity). • As additional examples (post-lecture activity). • As an additional material of challenging topics (pre-and post-lecture activity). • As a model solution of some exercises (post-lecture activity). 			
Module-1: Probability Distributions			
<p>Probability Distributions: Review of basic probability theory. Random variables (discrete and continuous), probability mass and density functions. Mathematical expectation, mean and variance. Binomial, Poisson and normal distributions- problems (derivations for mean and standard deviation for Binomial and Poisson distributions only)-Illustrative examples. Exponential distribution. (12 Hours)</p> <p>(RBT Levels: L1, L2 and L3)</p>			
Pedagogy	Chalk and Board, Problem-based learning		
Module-2: Joint probability distribution & Markov Chain			

<p>Joint probability distribution: Joint Probability distribution for two discrete random variables, expectation, covariance and correlation.</p> <p>Markov Chain: Introduction to Stochastic Process, Probability Vectors, Stochastic matrices, Regular stochastic matrices, Markov chains, Higher transition probabilities, Stationary distribution of Regular Markov chains and absorbing states. (12 Hours)</p> <p>(RBT Levels: L1, L2 and L3)</p>	
Pedagogy	Chalk and Board, Problem-based learning
Module-3: Statistical Inference 1	
<p>Introduction, sampling distribution, standard error, testing of hypothesis, levels of significance, test of significance, confidence limits, simple sampling of attributes, test of significance for large samples, comparison of large samples. (12 Hours)</p> <p>(RBT Levels: L1, L2 and L3)</p>	
Pedagogy	Chalk and Board, Problem-based learning
Module-4: Statistical Inference 2	
<p>Sampling variables, central limit theorem and confidences limit for unknown mean. Test of Significance for means of two small samples, students 't' distribution, Chi-square distribution as a test of goodness of fit. F-Distribution. (12 Hours)</p> <p>(RBT Levels: L1, L2 and L3)</p>	
Pedagogy	Chalk and Board, Problem-based learning
Module-5: Design of Experiments & ANOVA	
<p>Principles of experimentation in design, Analysis of completely randomized design, randomized block design. The ANOVA Technique, Basic Principle of ANOVA, One-way ANOVA, Two-way ANOVA, Latin-square Design, and Analysis of Co-Variance. (12 Hours)</p> <p>(RBT Levels: L1, L2 and L3)</p>	
Pedagogy	Chalk and Board, Problem-based learning

Test component, there are 25 marks.

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

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

Semester-End Examination:

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

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

Suggested Learning Resources:

Textbooks:

1. **Ronald E. Walpole, Raymond H Myers, Sharon L Myers & Keying Ye** "Probability & Statistics for Engineers & Scientists", Pearson Education, 9th edition, 2017.
2. Peter Bruce, Andrew Bruce & Peter Gedeck "Practical Statistics for Data Scientists" O'Reilly Media, Inc., 2nd edition 2020.

Reference Books: (Name of the author/Title of the Book/ Name of the publisher/Edition and Year)

1. **Erwin Kreyszig**, "Advanced Engineering Mathematics", John Wiley & Sons, 9th Edition, 2006.
2. **B. S. Grewal** "Higher Engineering Mathematics", Khanna publishers, 44th Ed., 2021.
3. **G Haribaskaran** "Probability, Queuing Theory & Reliability Engineering", Laxmi Publication, Latest Edition, 2006
4. **Irwin Miller & Marylees Miller**, John E. Freund's "Mathematical Statistics with Applications" Pearson. Dorling Kindersley Pvt. Ltd. India, 8th edition, 2014.
5. **S C Gupta and V K Kapoor**, "Fundamentals of Mathematical Statistics", S Chand and Company, Latest edition.
6. **Robert V. Hogg, Joseph W. McKean & Allen T. Craig**. "Introduction to Mathematical Statistics", Pearson Education 7th edition, 2013.
7. **Jim Pitman**. Probability, Springer-Verlag, 1993.
8. **Sheldon M. Ross**, "Introduction to Probability Models" 11th edition. Elsevier, 2014.
9. **A. M. Yaglom and I. M. Yaglom**, "Probability and Information". D. Reidel Publishing Company. Distributed by Hindustan Publishing
10. Reidel Publishing Company. Distributed by Hindustan Publishing

Corporation (India) Delhi, 1983.

11. P. G. Hoel, S. C. Port and C. J. Stone, "Introduction to Probability Theory", Universal Book Stall, (Reprint), 2003.
12. S. Ross, "A First Course in Probability", Pearson Education India, 6th Ed., 2002.
13. W. Feller, "An Introduction to Probability Theory and its Applications", Vol. 1, Wiley, 3rd Ed.,
14. N.P. Bali and Manish Goyal, A Textbook of Engineering Mathematics, Laxmi Publications, Reprint, 2010.
15. Veerarajan T, Engineering Mathematics (for semester III), Tata McGraw-Hill, New Delhi, 2010

OPERATINGSYSTEMS		Semester	3
Course Code	MVJ22A132	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 hours Theory + 20 hours practical	Total Marks	100
Credits	04	Exam Hours	3
Examination nature (SEE)	Theory		
<p>Course objectives:</p> <ul style="list-style-type: none"> ● To Demonstrate the need for OS and different types of OS ● To discuss suitable techniques for management of different resources ● To demonstrate different APIs/Commands related to processor, memory, storage and file system management. 			
<p>Teaching-Learning Process (General Instructions) Teachers can use the following strategies to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. Lecturer methods (L) need not to be only traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. 2. Use of Video/Animation to explain functioning of various concepts. 3. Encourage collaborative (Group Learning) Learning in the class. 4. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 5. Role play for process scheduling. 6. Demonstrate the installation of any one Linux OS on VMware/Virtual Box 			
MODULE-1			8 Hours
<p>Introduction to operating systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and Security; Distributed system; Special-purpose systems; Computing environments.</p> <p>Operating System Services: User - Operating System interface; System calls; Types of system calls; System programs; Operating system design and implementation; Operating System structure; Virtual machines; Operating System debugging, Operating System generation; System boot.</p> <p>Textbook 1: Chapter – 1 (1.1-1.12), 2 (2.2-2.11)</p>			
MODULE-2			8 Hours
<p>Process Management: Process concept; Process scheduling; Operations on processes; Inter process communication</p> <p>Multi-threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues.</p> <p>Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Thread scheduling; Multiple-processor scheduling,</p> <p>Textbook 1: Chapter – 3 (3.1-3.4), 4 (4.1-4.4), 5 (5.1 -5.5)</p>			
MODULE-3			8 Hours

Process Synchronization: Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization.

Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

Textbook 1: Chapter – 6 (6.1-6.6), 7 (7.1 -7.7)

MODULE-4

8 Hours

Memory Management: Memory management strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation.

Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.

Textbook 1: Chapter -8 (8.1-8.6), 9 (9.1-9.6)

MODULE-5

8 Hours

File System, Implementation of File System: File system: File concept; Access methods; Directory and Disk structure; File system mounting; File sharing; **Implementing File system:** File system structure; File system implementation; Directory implementation; Allocation methods; Free space management.

Secondary Storage Structure, Protection: Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; **Protection:** Goals of protection, Principles of protection, Domain of protection, Access matrix.

Textbook 1: Chapter – 10 (10.1-10.5) ,11 (11.1-11.5),12 (12.1-12.5), 14 (14.1-14.4)

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

Sl.N O	Experiments
1	Develop a c program to implement the Process system calls (fork (), exec(), wait(), create process, terminate process)
2	Simulate the following CPU scheduling algorithms to find turnaround time and waiting time a) FCFS b) SJF c) Round Robin d) Priority.
3	Develop a C program to simulate producer-consumer problem using semaphores.
4	Develop a C program which demonstrates inter-process communication between a reader process and a writer process. Use mk fifo, open, read, write and close APIs in your program.
5	Develop a C program to simulate Bankers Algorithm for Deadlock Avoidance.
6	Develop a C program to simulate the following contiguous memory allocation Techniques: a) Worst fit b) Best fit c) First fit.
7	Develop a C program to simulate page replacement algorithms: a) FIFO b) LRU
8	Simulate following File Organization Techniques a) Single level directory b) Two level directory

9	Develop a C program to simulate the Linked file allocation strategies.
10	Develop a C program to simulate SCAN disk scheduling algorithm.
11	<p>Debug a given C program //Moving Disk head to the inner most requested cylinder because this is Circular LOOK. queue[i]=queue2[0];</p> <pre> //Copying second array queue2[] after that first one is copied, into queue [] for(i=temp1+1,j=0;j<temp2;i++,j++) { queue[i]=queue2[j]; } //At this point, we have the queue[] with the requests in the //correct order of execution as per C-LOOK algorithm. //Now we have to set 0th index of queue[] to be the initial headposition. queue[0]=headposition; // Calculating SEEK TIME. seek is initially set to 0 in the declaration part. for(j=0; j<n; j++) //Loop starts from headposition. (ie. 0th index of queue) { // Finding the difference between next position and current position. difference = absoluteValue(queue[j+1]-queue[j]); // Adding difference to the current seek time value seek = seek + difference; // Displaying a message to show the movement of disk head printf("Disk head moves from position %d to %d with Seek %d \n", queue[j], queue[j+1], difference); } </pre>
<p>Course outcomes (Course Skill Set): At the end of the course, the student will be able to: CO 1. Explain the structure and functionality of operating system CO 2. Apply appropriate CPU scheduling algorithms for the given problem. CO 3. Analyze the various techniques for process synchronization and deadlock handling. CO 4. Apply the various techniques for memory management CO 5. Explain file and secondary storage management strategies. CO 6. Describe the need for information protection mechanisms</p>	

Assessment Details (both CIE and SEE)

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

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

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

CIE for the practical component of the IPCC

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

SEE for IPCC

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

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

The theory portion of the IPCC shall be for both CIE and SEE, whereas the practical portion will have a

- **CIE component only. Questions mentioned in the SEE paper may include questions from the practical component.**

Digital Design and Computer Organization		Semester	3
Course Code	MVJ22AI33	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:0:0	SEE Marks	50
Total Hours of Pedagogy	40 hours	Total Marks	100
Credits	03	Exam Hours	3
Examination nature (SEE)	Theory		
Course objectives:			
<ul style="list-style-type: none"> ● To demonstrate the functionalities of binary logic system ● To explain the working of combinational and sequential logic system ● To realize the basic structure of computer system ● To illustrate the working of I/O operations and processing unit 			
Teaching-Learning Process (General Instructions)			
These are sample Strategies; that teachers can use to accelerate the attainment of the various course outcomes.			
<ol style="list-style-type: none"> 1. Chalk and Talk 2. Live Demo with experiments 3. Power point presentation 			
MODULE-1		8 Hr	
Introduction to Digital Design: Binary Logic, Basic Theorems And Properties Of Boolean Algebra, Boolean Functions, Digital Logic Gates, Introduction, The Map Method, Four-Variable Map, Don't-Care Conditions, NAND and NOR Implementation, Other Hardware Description Language – Verilog Model of a simple circuit.			
Text book 1: 1.9, 2.4, 2.5, 2.8, 3.1, 3.2, 3.3, 3.5, 3.6, 3.9			
MODULE-2		8 Hr	
Combinational Logic: Introduction, Combinational Circuits, Design Procedure, Binary Adder- Subtractor, Decoders, Encoders, Multiplexers. HDL Models of Combinational Circuits – Adder, Multiplexer, Encoder.			
Sequential Logic: Introduction, Sequential Circuits, Storage Elements: Latches, Flip-Flops.			
Text book 1: 4.1, 4.2, 4.4, 4.5, 4.9, 4.10, 4.11, 4.12, 5.1, 5.2, 5.3, 5.4.			
MODULE-3		8 Hr	
Basic Structure of Computers: Functional Units, Basic Operational Concepts, Bus structure, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instruction and Instruction sequencing, Addressing Modes.			
Text book 2: 1.2, 1.3, 1.4, 1.6, 2.2, 2.3, 2.4, 2.5			
MODULE-4		8 Hr	
Input/output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access: Bus Arbitration, Speed, size and Cost of memory systems. Cache Memories – Mapping Functions.			
Text book 2: 4.1, 4.2.1, 4.2.2, 4.2.3, 4.4, 5.4, 5.5.1			
MODULE-5		8 Hr	

Basic Processing Unit: Some Fundamental Concepts: Register Transfers, Performing ALU operations, fetching a word from Memory, Storing a word in memory. Execution of a Complete Instruction. **Pipelining:** Basic concepts, Role of Cache memory, Pipeline Performance.

Text book 2: 7.1, 7.2, 8.1

Assessment Details (both CIE and SEE)

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

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

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

SEE for PCC

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

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

The theory portion of the PCC shall be for both CIE and SEE, whereas the practical portion will have a CIE component only.

Suggested Learning Resources:

Books

1. M. Morris Mano & Michael D. Ciletti, Digital Design With an Introduction to Verilog Design, 5e, Pearson Education.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer Organization, 5th Edition, Tata McGraw Hill.

Web links and Video Lectures (e-Resources):

<https://cse11-iiith.vlabs.ac.in/>

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

Assign the group task to Design the various types of counters and display the output accordingly

Methods

- Lab Assessment (25 Marks)
- GATE Based Aptitude Test

CO-PO/PSO Mapping																
CO/ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CO2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CO3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CO4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CO6	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

DATA STRUCTURES AND APPLICATIONS		Semester	3
Course Code	MVJ22AI34	CIE Marks	50
Teaching Hours/Week (L: T:P: S)	3:0:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	3
Examination type (SEE)	Theory		
<p>Course objectives:</p> <p>CLO 1. To explain fundamentals of data structures and their applications.</p> <p>CLO 2. To illustrate representation of Different data structures such as Stack, Queues, Linked Lists, Trees, and Graphs.</p> <p>CLO 3. To Design and Develop Solutions to problems using Linear Data Structures</p> <p>CLO 4. To discuss applications of Nonlinear Data Structures in problem solving.</p> <p>CLO 5. To introduce advanced Data structure concepts such as Hashing and Optimal Binary Search Trees</p>			
<p>Teaching-Learning Process (General Instructions)</p> <p>Teachers can use following strategies to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. Chalk and Talk with Black Board 2. ICT based Teaching 3. Demonstration based Teaching 			
Module-1		8Hours	
<p>INTRODUCTION TO DATA STRUCTURES: Data Structures, Classifications (Primitive & Non-Primitive), Data structure Operations</p> <p>Review of pointers and dynamic Memory Allocation,</p> <p>ARRAYS and STRUCTURES: Arrays, Dynamic Allocated Arrays, Structures and Unions, Polynomials, Sparse Matrices, representation of Multidimensional Arrays, Strings</p> <p>STACKS: Stacks, Stacks Using Dynamic Arrays, Evaluation and conversion of Expressions</p> <p>Text Book: Chapter-1:1.2 Chapter-2: 2.1 to 2.7 Chapter-3: 3.1,3.2,3.6</p> <p>Reference Book 1: 1.1 to 1.4</p>			
Module-2		8Hours	
<p>QUEUES: Queues, Circular Queues, Using Dynamic Arrays, Multiple Stacks and queues.</p> <p>LINKED LISTS : Singly Linked, Lists and Chains, Representing Chains in C, Linked Stacks and Queues, Polynomials</p> <p>Text Book: Chapter-3: 3.3, 3.4, 3.7 Chapter-4: 4.1 to 4.4</p>			
Module-3		8Hours	
<p>LINKED LISTS : Additional List Operations, Sparse Matrices, Doubly Linked List.</p> <p>TREES: Introduction, Binary Trees, Binary Tree Traversals, Threaded Binary Trees.</p> <p>Text Book: Chapter-4: 4.5,4.7,4.8 Chapter-5: 5.1 to 5.3, 5.5</p>			
Module-4		8Hours	
<p>TREES(Cont.): Binary Search trees, Selection Trees, Forests, Representation of Disjoint sets, Counting Binary Trees,</p> <p>GRAPHS: The Graph Abstract Data Types, Elementary Graph Operations</p> <p>Text Book: Chapter-5: 5.7 to 5.11 Chapter-6: 6.1, 6.2</p>			
Module-5		8Hours	

<p>HASHING: Introduction, Static Hashing, Dynamic Hashing PRIORITY QUEUES: Single and double ended Priority Queues, Leftist Trees INTRODUCTION TO EFFICIENT BINARY SEARCH TREES: Optimal Binary Search Trees Text Book: Chapter 8: 8.1 to 8.3 Chapter 9: 9.1, 9.2 Chapter 10: 10.1</p>
<p>Course outcome (Course Skill Set) At the end of the course the student will be able to: CO 1. Explain different data structures and their applications. CO 2. Apply Arrays, Stacks and Queue data structures to solve the given problems. CO 3. Use the concept of linked list in problem solving. CO 4. Develop solutions using trees and graphs to model the real-world problem. CO 5. Explain the advanced Data Structures concepts such as Hashing Techniques and Optimal Binary Search Trees.</p>
<p>Assessment Details (both CIE and SEE) <i>The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.</i></p> <p>Continuous Internal Evaluation:</p> <ul style="list-style-type: none"> ● <i>For the Assignment component of the CIE, there are 25 marks and for the Internal Assessment Test component, there are 25 marks.</i> ● <i>The first test will be administered after 40-50% of the syllabus has been covered, and the second test will be administered after 85-90% of the syllabus has been covered</i> ● <i>Any two assignment methods mentioned in the 220B2.4, if an assignment is project-based then only one assignment for the course shall be planned. The teacher should not conduct two assignments at the end of the semester if two assignments are planned.</i> ● <i>For the course, CIE marks will be based on a scaled-down sum of two tests and other methods of assessment.</i> <p>Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.</p> <p>Semester-End Examination: <i>Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (duration 03 hours).</i></p> <ol style="list-style-type: none"> 1. <i>The question paper will have ten questions. Each question is set for 20 marks.</i> 2. <i>There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), should have a mix of topics under that module.</i> 3. <i>The students have to answer 5 full questions, selecting one full question from each module.</i> 4. <i>Marks scored shall be proportionally reduced to 50 marks</i>
<p>Suggested Learning Resources:</p> <p>Textbook:</p> <ol style="list-style-type: none"> 1. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014

Reference Books:

1. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.
2. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed, Cengage Learning, 2014.
3. Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2012.
4. Jean-Paul Tremblay & Paul G. Sorenson, An Introduction to Data Structures with Applications, 2nd Ed, McGraw Hill, 2013
5. A M Tenenbaum, Data Structures using C, PHI, 1989
6. Robert Kruse, Data Structures and Program Design in C, 2nd Ed, PHI, 1996.

Web links and Video Lectures (e-Resources):

- <http://elearning.vtu.ac.in/econtent/courses/video/CSE/06CS35.html>
- <https://nptel.ac.in/courses/106/105/106105171/>
- <http://www.nptelvideos.in/2012/11/data-structures-and-algorithms.html>
- https://www.youtube.com/watch?v=3Xo6P_V-qns&t=201s
- <https://ds2-iiith.vlabs.ac.in/exp/selection-sort/index.html>
- <https://nptel.ac.in/courses/106/102/106102064/>
- <https://ds1-iiith.vlabs.ac.in/exp/stacks-queues/index.html>
- <https://ds1-iiith.vlabs.ac.in/exp/linked-list/basics/overview.html>
- <https://ds1-iiith.vlabs.ac.in/List%20of%20experiments.html>
- <https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/index.html>
- <https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/depth-first-traversal/dft-practice.html>
- https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01350159542807756812559/overview

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

- Role Play
- Flipped classroom
- Assessment Methods for 25 Marks (opt two Learning Activities)
 - Case Study
 - Programming Assignment
 - Gate Based Aptitude Test
 - MOOC Assignment for selected Module

DATA STRUCTURES LABORATORY			
SEMESTER – III			
Course Code	<i>MVJ22AIL35</i>	CIE Marks	50
Number of Contact Hours/Week	0:0:2	SEE Marks	50
Total Number of Lab Contact Hours	28	Exam Hours	03
Credits – 1			
Course Learning Objectives:			
<p>This laboratory course enables students to get practical experience in design, develop, implement, analyze and evaluation/testing of</p> <ul style="list-style-type: none"> ● Dynamic memory management ● Linear data structures and their applications such as stacks, queues and lists ● Non-Linear data structures and their applications such as trees and graphs 			
Descriptions (if any):			
<ul style="list-style-type: none"> ● Implement all the programs in “C” Programming Language and Linux OS. 			
Programs List:			
1.	Develop a Program in C for the following: <ol style="list-style-type: none"> a) Declare a calendar as an array of 7 elements (A dynamically Created array) to represent 7 days of a week. Each Element of the array is a structure having three fields. The first field is the name of the Day (A dynamically allocated String), The second field is the date of the Day (A integer), the third field is the description of the activity for a particular day (A dynamically allocated String). b) Write functions create (), read() and display(); to create the calendar, to read the data from the keyboard and to print weeks activity details report on screen. 		
2.	Develop a Program in C for the following operations on Strings. <ol style="list-style-type: none"> a. Read a main String (STR), a Pattern String (PAT) and a Replace String (REP) b. Perform Pattern Matching Operation: Find and Replace all occurrences of PAT in STR with REP if PAT exists in STR. Report suitable messages in case PAT does not exist in STR Support the program with functions for each of the above operations. Don't use Built-in functions.		
3.	Develop a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX) <ol style="list-style-type: none"> a. Push an Element on to Stack b. Pop an Element from Stack c. Demonstrate how Stack can be used to check Palindrome d. Demonstrate Overflow and Underflow situations on Stack e. Display the status of Stack f. Exit Support the program with appropriate functions for each of the above operations		

4.	Develop a Program in C for converting an Infix Expression to Postfix Expression. Program should support for both parenthesized and free parenthesized expressions with the operators: +, -, *, /, % (Remainder), ^ (Power) and alphanumeric operands.
5.	Develop a Program in C for the following Stack Applications <ol style="list-style-type: none"> Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^ Solving Tower of Hanoi problem with n disks
6.	Develop a menu driven Program in C for the following operations on Circular QUEUE of Characters (Array Implementation of Queue with maximum size MAX) <ol style="list-style-type: none"> Insert an Element on to Circular QUEUE Delete an Element from Circular QUEUE Demonstrate Overflow and Underflow situations on Circular QUEUE Display the status of Circular QUEUE Exit Support the program with appropriate functions for each of the above operations
7.	Develop a menu driven Program in C for the following operations on Singly Linked List (SLL) of Student Data with the fields: <i>USN, Name, Programme, Sem, PhNo</i> <ol style="list-style-type: none"> Create a SLL of N Students Data by using <i>front insertion</i>. Display the status of SLL and count the number of nodes in it Perform Insertion / Deletion at End of SLL Perform Insertion / Deletion at Front of SLL(Demonstration of stack) Exit
8.	Develop a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: <i>SSN, Name, Dept, Designation, Sal, PhNo</i> <ol style="list-style-type: none"> Create a DLL of N Employees Data by using <i>end insertion</i>. Display the status of DLL and count the number of nodes in it Perform Insertion and Deletion at End of DLL Perform Insertion and Deletion at Front of DLL Demonstrate how this DLL can be used as Double Ended Queue. Exit
9.	Develop a Program in C for the following operations on Singly Circular Linked List (SCLL) with header nodes <ol style="list-style-type: none"> Represent and Evaluate a Polynomial $P(x,y,z) = 6x^2y^2z - 4yz^5 + 3x^3yz + 2xy^5z - 2xyz^3$ Find the sum of two polynomials POLY1(x,y,z) and POLY2(x,y,z) and store the result in POLYSUM(x,y,z) Support the program with appropriate functions for each of the above operations
10.	Develop a menu driven Program in C for the following operations on Binary Search Tree (BST) of Integers . <ol style="list-style-type: none"> Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2 Traverse the BST in Inorder, Preorder and Post Order Search the BST for a given element (KEY) and report the appropriate message Exit
11.	Develop a Program in C for the following operations on Graph(G) of Cities <ol style="list-style-type: none"> Create a Graph of N cities using Adjacency Matrix. Print all the nodes reachable from a given starting node in a digraph using DFS/BFS method

12.	<p>Given a File of N employee records with a set K of Keys (4-digit) which uniquely determine the records in file F. Assume that file F is maintained in memory by a Hash Table (HT) of m memory locations with L as the set of memory addresses (2-digit) of locations in HT. Let the keys in K and addresses in L are Integers. Develop a Program in C that uses Hash function H: $K \rightarrow L$ as $H(K)=K \text{ mod } m$ (remainder method), and implement hashing technique to map a given key K to the address space L. Resolve the collision (if any) using linear probing.</p>
<p>Note: During the lab sessions the data structures using python codes will be demonstrated.</p>	
<p>Laboratory Outcomes: The student should be able to:</p>	

- Analyze various linear and non-linear data structures
- Demonstrate the working nature of different types of data structures and their applications
- Use appropriate searching and sorting algorithms for the give scenario.
- Apply the appropriate data structure for solving real world problems

Conduct of Practical Examination:

- Experiment distribution
 - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
 - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (*Need to change in accordance with university regulations*)
 - c) For laboratories having only one part – Procedure + Execution + Viva-Voce: $15+70+15 = 100$ Marks
 - d) For laboratories having PART A and PART B
 - i. Part A – Procedure + Execution + Viva = $6 + 28 + 6 = 40$ Marks
 - ii. Part B – Procedure + Execution + Viva = $9 + 42 + 9 = 60$ Marks

Programming using Java		<i>Semester</i>	3
<i>Course Code</i>	MVJ22AI361	<i>CIE Marks</i>	50
<i>Teaching Hours/Week (L: T:P: S)</i>	2:0:2	<i>SEE Marks</i>	50
<i>Total Hours of Pedagogy</i>	28 Hours of Theory + 20 Hours of Practical	<i>Total Marks</i>	100
<i>Credits</i>	03	<i>Exam Hours</i>	03
<i>Examination type (SEE)</i>	Theory		
<p>Note - Students who have undergone " Basics of Java Programming-BPLCK105C/205C" in first year are not eligible to opt this course</p>			
<p>Course objectives:</p> <ul style="list-style-type: none"> To learn primitive constructs JAVA programming language. To understand Object Oriented Programming Features of JAVA. To gain knowledge on: packages, multi threaded programming and exceptions. 			
<p>Teaching-Learning Process (General Instructions) These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes and make Teaching –Learning more effective</p> <ol style="list-style-type: none"> Use Online Java Compiler IDE: https://www.jdoodle.com/online-java-compiler/ or any other. Demonstration of programming examples. Chalk and board, power point presentations Online material (Tutorials) and video lectures. 			
Module-1			
<p>An Overview of Java: Object-Oriented Programming (Two Paradigms, Abstraction, The Three OOP Principles), Using Blocks of Code, Lexical Issues (Whitespace, Identifiers, Literals, Comments, Separators, The Java Keywords).</p> <p>Data Types, Variables, and Arrays: The Primitive Types (Integers, Floating-Point Types, Characters, Booleans), Variables, Type Conversion and Casting, Automatic Type Promotion in Expressions, Arrays, Introducing Type Inference with Local Variables.</p> <p>Operators: Arithmetic Operators, Relational Operators, Boolean Logical Operators, The Assignment Operator, The ? Operator, Operator Precedence, Using Parentheses.</p> <p>Control Statements: Java’s Selection Statements (if, The Traditional switch), Iteration Statements (while, do-while, for, The For-Each Version of the for Loop, Local Variable Type Inference in a for Loop, Nested Loops), Jump Statements (Using break, Using continue, return).</p> <p>Chapter 2, 3, 4, 5</p>			
Module-2			
<p>Introducing Classes: Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this Keyword, Garbage Collection.</p> <p>Methods and Classes: Overloading Methods, Objects as Parameters, Argument Passing, Returning Objects, Recursion, Access Control, Understanding static, Introducing final, Introducing Nested and Inner Classes.</p> <p>Chapter 6, 7</p>			
Module-3			

<p>Inheritance: Inheritance Basics, Using super, Creating a Multilevel Hierarchy, When Constructors Are Executed, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, Local Variable Type Inference and Inheritance, The Object Class.</p> <p>Interfaces: Interfaces, Default Interface Methods, Use static Methods in an Interface, Private Interface Methods.</p> <p>Chapter 8, 9</p>
--

Module-4
<p>Packages: Packages, Packages and Member Access, Importing Packages.</p> <p>Exceptions: Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions, Creating Your Own Exception Subclasses, Chained Exceptions.</p> <p>Chapter 9, 10</p>

Module-5
<p>Multithreaded Programming: The Java Thread Model, The Main Thread, Creating a Thread, Creating Multiple Threads, Using isAlive() and join(), Thread Priorities, Synchronization, Interthread Communication, Suspending, Resuming, and Stopping Threads, Obtaining a Thread's State.</p> <p>Enumerations, Type Wrappers and Autoboxing: Enumerations (Enumeration Fundamentals, The values() and valueOf() Methods), Type Wrappers (Character, Boolean, The Numeric Type Wrappers), Autoboxing (Autoboxing and Methods, Autoboxing/Unboxing Occurs in Expressions, Autoboxing/Unboxing Boolean and Character Values).</p> <p>Chapter 11, 12</p>

Course outcome (Course Skill Set)

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

1. Demonstrate proficiency in writing simple programs involving branching and looping structures.
2. Design a class involving data members and methods for the given scenario.
3. Apply the concepts of inheritance and interfaces in solving real world problems.
4. Use the concept of packages and exception handling in solving complex problem
5. Apply concepts of multithreading, autoboxing and enumerations in program development

Programming Experiments (Suggested and are not limited to)

1. Develop a JAVA program to add TWO matrices of suitable order N (The value of N should be read from command line arguments).
2. Develop a stack class to hold a maximum of 10 integers with suitable methods. Develop a JAVA main method to illustrate Stack operations.
3. A class called Employee, which models an employee with an ID, name and salary, is designed as shown in the following class diagram. The method raiseSalary (percent) increases the salary by the given percentage. Develop the Employee class and suitable main method for demonstration.
4. A class called MyPoint, which models a 2D point with x and y coordinates, is designed as follows:
 - Two instance variables x (int) and y (int).
 - A default (or "no-arg") constructor that construct a point at the default location of (0, 0).
 - A overloaded constructor that constructs a point with the given x and y coordinates.
 - A method setXY() to set both x and y.
 - A method getXY() which returns the x and y in a 2-element int array.
 - A toString() method that returns a string description of the instance in the format "(x, y)".
 - A method called distance(int x, int y) that returns the distance from this point to another point at the given (x, y) coordinates
 - An overloaded distance(MyPoint another) that returns the distance from this point to the given MyPoint instance (called another)
 - Another overloaded distance() method that returns the distance from this point to the origin (0,0)
 Develop the code for the class MyPoint. Also develop a JAVA program (called TestMyPoint) to test all the methods defined in the class.
5. Develop a JAVA program to create a class named shape. Create three sub classes namely: circle, triangle and square, each class has two member functions named draw () and erase (). Demonstrate polymorphism concepts by developing suitable methods, defining member data and main program.
6. Develop a JAVA program to create an abstract class Shape with abstract methods calculateArea() and calculatePerimeter(). Create subclasses Circle and Triangle that extend the Shape class and implement the respective methods to calculate the area and perimeter of each shape.
7. Develop a JAVA program to create an interface Resizable with methods resizeWidth(int width) and resizeHeight(int height) that allow an object to be resized. Create a class Rectangle that implements the Resizable interface and implements the resize methods
8. Develop a JAVA program to create an outer class with a function display. Create another class inside the outer class named inner with a function called display and call the two functions in the main class.
9. Develop a JAVA program to raise a custom exception (user defined exception) for DivisionByZero using try, catch, throw and finally.
10. Develop a JAVA program to create a package named mypack and import & implement it in a suitable class.
11. Write a program to illustrate creation of threads using runnable class. (start method start each of the newly created thread. Inside the run method there is sleep() for suspend the thread for 500 milliseconds).
12. Develop a program to create a class MyThread in this class a constructor, call the base class constructor, using super and start the thread. The run method of the class starts after this. It can be observed that both main thread and created child thread are executed concurrently.

Assessment Details (both CIE and SEE)

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

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

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

CIE for the practical component of the IPCC

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

SEE for IPCC

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

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

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

Suggested Learning Resources:

Textbook:

Java: The Complete Reference, Twelfth Edition, by Herbert Schildt, November 2021, McGraw-

Hill, ISBN:9781260463422

Reference Books

1. Programming with Java, 6th Edition, by E Balagurusamy, Mar-2019, McGraw Hill Education, ISBN:9789353162337.
2. Thinking in Java, Fourth Edition, by Bruce Eckel, Prentice Hall, 2006 (https://sd.blackball.lv/library/thinking_in_java_4th_edition.pdf)

Web links and Video Lectures (e-Resources):

- Java Tutorial: <https://www.geeksforgeeks.org/java/>
- Introduction To Programming In Java (by Evan Jones, Adam Marcus and Eugene Wu): <https://ocw.mit.edu/courses/6-092-introduction-to-programming-in-java-january-iap-2010/>
- Java Tutorial: <https://www.w3schools.com/java/>
- Java Tutorial: <https://www.javatpoint.com/java-tutorial>

Activity Based Learning (Suggested Activities)/ Practical Based learning

1. Installation of Java (Refer: https://www.java.com/en/download/help/index_installing.html)
2. Demonstration of online IDEs like geeksforgeeks, jdoodle or any other Tools
3. Demonstration of class diagrams for the class abstraction, type visibility, composition and inheritance

Assessment Method

- Programming Assignment / Course Project

<i>Data Analytics with R Programming</i>		<i>Semester</i>	3
Course Code	MVJ22AI363	CIE Marks	50
Teaching Hours/Week (L: T:P: S)	0:0:2:0	SEE Marks	50
Credits	01	Exam Hours	02
Examination type (SEE)	Practical		
Course objectives:			
<ul style="list-style-type: none"> To explore and understand how R and R Studio interactive environment. To understand the different data Structures, data types in R. To learn and practice programming techniques using R programming. To import data into R from various data sources and generate visualizations. To draw insights from datasets using data analytics techniques. 			
SLN	Experiments		
O			
1	<p>Demonstrate the steps for installation of R and R Studio. Perform the following:</p> <ol style="list-style-type: none"> Assign different type of values to variables and display the type of variable. Assign different types such as Double, Integer, Logical, Complex and Character and understand the difference between each data type. Demonstrate Arithmetic and Logical Operations with simple examples. Demonstrate generation of sequences and creation of vectors. Demonstrate Creation of Matrices Demonstrate the Creation of Matrices from Vectors using Binding Function. Demonstrate element extraction from vectors, matrices and arrays <p>Suggested Reading – Text Book 1 – Chapter 1 (What is R, Installing R, Choosing an IDE – RStudio, How to Get Help in R, Installing Extra Related Software), Chapter 2 (Mathematical Operations and Vectors, Assigning Variables, Special Numbers, Logical Vectors), Chapter 3 (Classes, Different Types of Numbers, Other Common Classes, Checking and Changing Classes, Examining Variables)</p>		
2	<p>Assess the Financial Statement of an Organization being supplied with 2 vectors of data: Monthly Revenue and Monthly Expenses for the Financial Year. You can create your own sample data vector for this experiment) Calculate the following financial metrics:</p> <ol style="list-style-type: none"> Profit for each month. Profit after tax for each month (Tax Rate is 30%). Profit margin for each month equals to profit after tax divided by revenue. Good Months – where the profit after tax was greater than the mean for the year. Bad Months – where the profit after tax was less than the mean for the year. The best month – where the profit after tax was max for the year. The worst month – where the profit after tax was min for the year. <p>Note:</p> <ol style="list-style-type: none"> All Results need to be presented as vectors Results for Dollar values need to be calculated with \$0.01 precision, but need to be presented in Units of \$1000 (i.e 1k) with no decimal points Results for the profit margin ratio need to be presented in units of % with no decimal point. It is okay for tax to be negative for any given month (deferred tax asset) Generate CSV file for the data. <p>Suggested Reading – Text Book 1 – Chapter 4 (Vectors, Combining Matrices)</p>		
3	<p>Develop a program to create two 3 X 3 matrices A and B and perform the following operations a) Transpose of the matrix b) addition c) subtraction d) multiplication</p> <p>Suggested Reading – Text Book 1 – Chapter 4 (Matrices and Arrays – Array Arithmetic)</p>		
4	<p>Develop a program to find the factorial of given number using recursive function calls.</p> <p>Suggested Reading – Reference Book 1 – Chapter 5 (5.5 – Recursive Programming) Text Book 1 – Chapter 8 (Flow Control and Loops – If and Else, Vectorized If, while loops, for loops), Chapter 6 (Creating and Calling Functions, Passing Functions to and from other functions)</p>		

5	<p>Develop an R Program using functions to find all the prime numbers up to a specified number by the method of Sieve of Eratosthenes.</p> <p>Suggested Reading – Reference Book 1 - Chapter 5 (5.5 – Recursive Programming) Text Book 1 – Chapter 8 (Flow Control and Loops – If and Else, Vectorized If, while loops, for loops), Chapter 6 (Creating and Calling Functions, Passing Functions to and from other functions)</p>																		
6	<p>The built-in data set mammals contain data on body weight versus brain weight. Develop R commands to:</p> <p>a) Find the Pearson and Spearman correlation coefficients. Are they similar? b) Plot the data using the plot command. c) Plot the logarithm (log) of each variable and see if that makes a difference.</p> <p>Suggested Reading – Text Book 1 –Chapter 12 – (Built-in Datasets) Chapter 14 – (Scatterplots) Reference Book 2 – 13.2.5 (Covariance and Correlation)</p>																		
7	<p>Develop R program to create a Data Frame with following details and do the following operations.</p> <table border="1" data-bbox="272 646 1101 982"> <thead> <tr> <th>itemCode</th> <th>itemCategory</th> <th>itemPrice</th> </tr> </thead> <tbody> <tr> <td>1001</td> <td>Electronics</td> <td>700</td> </tr> <tr> <td>1002</td> <td>Desktop Supplies</td> <td>300</td> </tr> <tr> <td>1003</td> <td>Office Supplies</td> <td>350</td> </tr> <tr> <td>1004</td> <td>USB</td> <td>400</td> </tr> <tr> <td>1005</td> <td>CD Drive</td> <td>800</td> </tr> </tbody> </table> <p>a) Subset the Data frame and display the details of only those items whose price is greater than or equal to 350. b) Subset the Data frame and display only the items where the category is either “Office Supplies” or “Desktop Supplies” c) Create another Data Frame called “item-details” with three different fields itemCode, ItemQtyonHand and ItemReorderLvl and merge the two frames</p> <p>Suggested Reading –Textbook 1: Chapter 5 (Lists and Data Frames)</p>	itemCode	itemCategory	itemPrice	1001	Electronics	700	1002	Desktop Supplies	300	1003	Office Supplies	350	1004	USB	400	1005	CD Drive	800
itemCode	itemCategory	itemPrice																	
1001	Electronics	700																	
1002	Desktop Supplies	300																	
1003	Office Supplies	350																	
1004	USB	400																	
1005	CD Drive	800																	
8	<p>Let us use the built-in dataset air quality which has Daily air quality measurements in New York, May to September 1973. Develop R program to generate histogram by using appropriate arguments for the following statements.</p> <p>a) Assigning names, using the air quality data set. b) Change colors of the Histogram c) Remove Axis and Add labels to Histogram d) Change Axis limits of a Histogram e) Add Density curve to the histogram</p> <p>Suggested Reading –Reference Book 2 – Chapter 7 (7.4 – The ggplot2 Package), Chapter 24 (Smoothing and Shading)</p>																		
9	<p>Design a data frame in R for storing about 20 employee details. Create a CSV file named “input.csv” that defines all the required information about the employee such as id, name, salary, start_date, dept. Import into R and do the following analysis.</p> <p>a) Find the total number rows & columns b) Find the maximum salary c) Retrieve the details of the employee with maximum salary d) Retrieve all the employees working in the IT Department. e) Retrieve the employees in the IT Department whose salary is greater than 20000 and write these</p>																		

	<p>details into another file “output.csv”</p> <p>Suggested Reading – Text Book 1 – Chapter 12(CSV and Tab Delimited Files)</p>
10	<p>Using the built in dataset mtcars which is a popular dataset consisting of the design and fuel consumption patterns of 32 different automobiles. The data was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973-74 models). Format A data frame with 32 observations on 11 variables : [1] mpg Miles/(US) gallon, [2] cyl Number of cylinders [3] disp Displacement (cu.in.), [4] hp Gross horsepower [5] drat Rear axle ratio,[6] wt Weight (lb/1000) [7] qsec 1/4 mile time, [8] vs V/S, [9] am Transmission (0 = automatic, 1 = manual), [10] gear Number of forward gears, [11] carb Number of carburetors</p> <p>Develop R program, to solve the following:</p> <ol style="list-style-type: none"> What is the total number of observations and variables in the dataset? Find the car with the largest hp and the least hp using suitable functions Plot histogram / density for each variable and determine whether continuous variables are normally distributed or not. If not, what is their skewness? What is the average difference of gross horse power(hp) between automobiles with 3 and 4 number of cylinders(cyl)? Also determine the difference in their standard deviations. Which pair of variables has the highest Pearson correlation? <p>References (Web links):</p> <ol style="list-style-type: none"> https://cran.r-project.org/web/packages/explore/vignettes/explore_mtcars.html https://www.w3schools.com/r/r_stat_data_set.asp https://rpubs.com/BillB/217355
11	<p>Demonstrate the progression of salary with years of experience using a suitable data set (You can create your own dataset). Plot the graph visualizing the best fit line on the plot of the given data points. Plot a curve of Actual Values vs. Predicted values to show their correlation and performance of the model.</p> <p>Interpret the meaning of the slope and y-intercept of the line with respect to the given data. Implement using lm function. Save the graphs and coefficients in files. Attach the predicted values of salaries as a new column to the original data set and save the data as a new CSV file.</p> <p>Suggested Reading – Reference Book 2 – Chapter 20 (General Concepts, Statistical Inference, Prediction)</p>
<p>Note: Data analytics part is also included in the mathematics.</p>	
<p>Course outcomes (Course Skill Set):</p> <p>At the end of the course the student will be able to:</p> <ul style="list-style-type: none"> Explain the fundamental syntax of R data types, expressions and the usage of the R-Studio IDE Develop a program in R with programming constructs: conditionals, looping and functions. Apply the list and data frame structure of the R programming language. Use visualization packages and file handlers for data analysis.. 	

Assessment Details (both CIE and SEE)

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

Continuous Internal Evaluation (CIE):

CIE marks for the practical course are 50 Marks.

The split-up of CIE marks for record/ journal and test are in the ratio 60:40.

- *Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.*
- *Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.*
- *Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).*
- *Weightage to be given for neatness and submission of record/write-up on time.*
- *Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.*
- *In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.*
- *The suitable rubrics can be designed to evaluate each student's performance and learning ability.*
- *The marks scored shall be scaled down to 20 marks (40% of the maximum marks).*

The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student.

Semester End Evaluation (SEE):

- *SEE marks for the practical course are 50 Marks.*
- *SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.*
- *The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University.*
- *All laboratory experiments are to be included for practical examination.*
- *(Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. OR based on the course requirement evaluation*

rubrics shall be decided jointly by examiners.

- *Students can pick one question (experiment) from the questions lot prepared by the examiners jointly.*
- *Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.*

General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)

Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero.

The minimum duration of SEE is 02 hours

Suggested Learning Resources:

Book:

1. Cotton, R. (2013). Learning R: A Step by Step Function Guide to Data Analysis. 1st ed. O'Reilly Media Inc.

References:

1. Jones, O., Maillardet, R. and Robinson, A. (2014). Introduction to Scientific Programming and Simulation Using R. Chapman & Hall/CRC, The R Series.
2. Davies, T.M. (2016) The Book of R: A First Course in Programming and Statistics. No Starch Press.

MVJ22SCR37– Social Connect & Responsibility		Semester	3rd
Course Code	MVJ22SCR37 Common for all dept	CIE Marks	100
Teaching Hours/Week (L:T:P: S)	0:0:3:1	SEE Marks	-----
Total Hours of Pedagogy	40 hour Practical Session +15 hour Planning	Total Marks	100
Examination nature (No SEE – Only CIE)	For CIE Assessment - Activities Report Evaluation by College NSS Officer / HOD / Sports Dept / Any Dept.		
Credits	01 - Credit		
<p>Course objectives: The course will enable the students to:</p> <ol style="list-style-type: none"> 1. Provide a formal platform for students to communicate and connect to the surrounding. 2. create a responsible connection with the society. 3. Understand the community in general in which they work. 4. Identify the needs and problems of the community and involve them in problem –solving. 5. Develop among themselves a sense of social & civic responsibility & utilize their knowledge in finding practical solutions to individual and community problems. 6. Develop competence required for group-living and sharing of responsibilities & gain skills in mobilizing community participation to acquire leadership qualities and democratic attitudes. 			
<p>General Instructions - Pedagogy : These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the activities will develop students’ theoretical and applied social and cultural skills. 2. State the need for activities and its present relevance in the society and Provide real-life examples. 3. Support and guide the students for self-planned activities. 4. You will also be responsible for assigning homework, grading assignments and quizzes, and documenting students’ progress in real activities in the field. 5. Encourage the students for group work to improve their creative and analytical skills. 			
<p>Contents : The course is mainly activity-based that will offer a set of activities for the student that enables them to connect with fellow human beings, nature, society, and the world at large. The course will engage students for interactive sessions, open mic, reading group, storytelling sessions, and semester-long activities conducted by faculty mentors. In the following a set of activities planned for the course have been listed:</p>			
Social Connect & Responsibility – Contents			
<p>Part I: Plantation and adoption of a tree: Plantation of a tree that will be adopted for four years by a group of BE / B.Tech students. (ONE STUDENT ONE TREE) They will also make an excerpt either as a documentary or a photo blog describing the plant’s origin, its usage in daily life, its appearance in folklore and literature - – Objectives, Visit, case study, report, outcomes.</p>			
<p>Part II : Heritage walk and crafts corner: Heritage tour, knowing the history and culture of the city, connecting to people around through their history, knowing the city and its craftsman, photo blog and documentary on evolution and practice of various craft forms - – Objectives, Visit, case study, report, outcomes.</p>			
<p>Part III : Organic farming and waste management: Usefulness of organic farming, wet waste management in neighboring villages, and implementation in the campus – Objectives, Visit, case study, report, outcomes.</p>			

Part IV:

Water conservation:

Knowing the present practices in the surrounding villages and implementation in the campus, documentary or photoblog presenting the current practices – Objectives, Visit, case study, report, outcomes.

Part V :

Food walk:

City's culinary practices, food lore, and indigenous materials of the region used in cooking – Objectives, Visit, case study, report, outcomes.

Course outcomes (Course Skill Set):

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

CO1: Communicate and connect to the surrounding. CO2: Create a responsible connection with the society.

CO3: Involve in the community in general in which they work.

CO4: Notice the needs and problems of the community and involve them in problem –solving.

CO5: Develop among themselves a sense of social & civic responsibility & utilize their knowledge in finding practical solutions to individual and community problems.

CO6: Develop competence required for group-living and sharing of responsibilities & gain skills in mobilizing community participation to acquire leadership qualities and democratic attitudes.

Activities:

Jamming session, open mic, and poetry: Platform to connect to others. Share the stories with others. Share the experience of Social Connect. Exhibit the talent like playing instruments, singing, one-act play, art-painting, and fine art.

PEDAGOGY:

The pedagogy will include interactive lectures, inspiring guest talks, field visits, social immersion, and a course project. Applying and synthesizing information from these sources to define the social problem to address and take up the solution as the course project, with your group. Social immersion with NGOs/social sections will be a key part of the course. Will all lead to the course project that will address the needs of the social sector?

COURSE TOPICS:

The course will introduce social context and various players in the social space, and present approaches to discovering and understanding social needs. Social immersion and inspiring conversational will culminate in developing an actual, idea for problem-based intervention, based on an in-depth understanding of a key social problem.

Duration :

A total of 40 - 50 hrs engagement per semester is required for the 3rd semester of the B.E. /B.Tech. program. The students will be divided into groups. Each group will be handled by faculty mentor. Faculty mentor will design the activities (particularly Jamming sessions open mic ,and poetry) Faculty mentors has to design the evaluation system as per VTU guidelines of scheme & syllabus.

Guideline for Assessment Process:**Continuous Internal Evaluation (CIE):**

After completion of the course, the student shall prepare, with daily diary as reference, a comprehensive report in consultation with the mentor/s to indicate what he has observed and learned in the social connect period. The report should be signed by the mentor. The report shall be evaluated on the basis of the following criteria and/or other relevant criteria pertaining to the activity completed. Marks allotted for the diary are out of 50. Planning and scheduling the social connect Information/Data collected during the social connect Analysis of the information/data and report writing Considering all above points allotting the marks as mentioned below

Excellent	: 80 to 100
Good	: 60 to 79
Satisfactory:	40 to 59
Unsatisfactory and fail :	<39

Special Note :**NO SEE – Semester End Exam – Completely Practical and activities based evaluation****Pedagogy – Guidelines :**

It may differ depending on local resources available for the study as well as environment and climatic differences, location and time of execution.

Sl No	Topic	Group size	Location	Activity execution	Reporting	Evaluation Of the Topic
1.	Plantation and adoption of a tree:	May be individual or team	Farmers land/ parks / Villages / roadside/ community area / College campus etc.....	Site selection /proper consultation/Continuous monitoring/ Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
2.	Heritage walk and crafts corner:	May be individual or team	Temples / monumental places / Villages/ City Areas / Grama panchayat/ public associations/Government Schemes officers/ campus etc.....	Site selection /proper consultation/Continuous monitoring/ Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
3.	Organic farming and waste management:	May be individual or team	Farmers land / parks / Villages visits / roadside/ community area / College campus etc.....	Group selection / proper consultation / Continuous monitoring / Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
4.	Water conservation: & conservation techniques	May be individual or team	Villages/ City Areas / Grama panchayat/ public associations/Government Schemes officers / campus etc.....	site selection / proper consultation/Continuous monitoring/ Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
5.	Food walk: Practices in society	May be individual or team	Villages/ City Areas / Grama panchayat/ public associations/Government Schemes officers/ campus etc.....	Group selection / proper consultation / Continuous monitoring / Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty

Plan of Action (Execution of Activities)

Sl.NO	Practice Session Description
1	Lecture session in field to start activities
2	Students Presentation on Ideas
3	Commencement of activity and its progress
4	Execution of Activity
5	Execution of Activity
6	Execution of Activity
7	Execution of Activity
8	Case study based Assessment, Individual performance
9	Sector/ Team wise study and its consolidation
10	Video based seminar for 10 minutes by each student At the end of semester with Report.

- Each student should do activities according to the scheme and syllabus.
 - At the end of semester student performance has to be evaluated by the faculty for the assigned activity progress and its completion.
 - At last consolidated report of all activities from 1st to 5th, compiled report should be submitted as per the instructions and scheme.
-

Assessment Details for CIE (both CIE and SEE)

Weightage	CIE – 100%	
Field Visit, Plan, Discussion	10 Marks	<ul style="list-style-type: none"> • Implementation strategies of the project (NSS work). • The last report should be signed by NSS Officer, the HOD and principal. • At last report should be evaluated by the NSS officer of the institute. • Finally the consolidated marks sheet should be sent to the university and also to be made available at LIC visit.
Commencement of activities and its progress	20 Marks	
Case study based Assessment Individual performance with report	20 Marks	
Sector wise study & its consolidation 5*5 = 25	25 Marks	
Video based seminar for 10 minutes by each student At the end of semester with Report. <u>Activities 1 to 5, 5*5 = 25</u>	25 Marks	
Total marks for the course in each semester	100 Marks	

For each activity, 20 marks CIE will be evaluated for IA marks at the end of semester, Report and assessment copy should be made available in the department.

Students should present the progress of the activities as per the schedule in the prescribed practical session in the field. There should be positive progress in the vertical order for the benefit of society in general through activities.



MVJCE CURRICULUM

FOR

ARTIFICIAL INTELLIGENCE & MACHINE LEARNING(Scheme 2022)

IV SEMESTER

Analysis & Design of Algorithms		Semester	4
Course Code	MVJ22AI41	CIE Marks	50
Teaching Hours/Week (L: T:P: S)	3:0:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Examination type (SEE)	Theory		
<p>Course objectives:</p> <ul style="list-style-type: none"> • To learn the methods for analyzing algorithms and evaluating their performance. • To demonstrate the efficiency of algorithms using asymptotic notations. • To solve problems using various algorithm design methods, including brute force, greedy, divide and conquer, decrease and conquer, transform and conquer, dynamic programming, backtracking, and branch and bound. • To learn the concepts of P and NP complexity classes. 			
<p>Teaching-Learning Process (General Instructions) These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. Lecturer method (L) does not mean only the traditional lecture method, but different types of teaching methods may be adopted to achieve the outcomes. 2. Utilize video/animation films to illustrate the functioning of various concepts. 3. Promote collaborative learning (Group Learning) in the class. 4. Pose at least three HOT (Higher Order Thinking) questions in the class to stimulate critical thinking. 5. Incorporate Problem-Based Learning (PBL) to foster students' analytical skills and develop their ability to evaluate, generalize, and analyze information rather than merely recalling it. 6. Introduce topics through multiple representations. 7. Demonstrate various ways to solve the same problem and encourage students to devise their own creative solutions. 8. Discuss the real-world applications of every concept to enhance students' comprehension. 			
Module-1			
<p>INTRODUCTION: What is an Algorithm?, Fundamentals of Algorithmic Problem Solving. FUNDAMENTALS OF THE ANALYSIS OF ALGORITHM EFFICIENCY: Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non recursive Algorithms, Mathematical Analysis of Recursive Algorithms. BRUTE FORCE APPROACHES: Selection Sort and Bubble Sort, Sequential Search and Brute Force String Matching. Chapter 1 (Sections 1.1,1.2), Chapter 2(Sections 2.1,2.2,2.3,2.4), Chapter 3(Section 3.1,3.2)</p>			
Module-2			
<p>BRUTE FORCE APPROACHES (contd.): Exhaustive Search (Travelling Salesman problem and Knapsack Problem). DECREASE-AND-CONQUER: Insertion Sort, Topological Sorting. DIVIDE AND CONQUER: Merge Sort, Quick Sort, Binary Tree Traversals, Multiplication of Large Integers and Strassen's Matrix Multiplication.</p>			

Chapter 3(Section 3.4), Chapter 4 (Sections 4.1,4.2), Chapter 5 (Section 5.1,5.2,5.3, 5.4)

Module-3

TRANSFORM-AND-CONQUER: Balanced Search Trees, Heaps and Heapsort.

SPACE-TIME TRADEOFFS: Sorting by Counting: Comparison counting sort, Input Enhancement in String Matching: Horspool's Algorithm.

Chapter 6 (Sections 6.3,6.4), Chapter 7 (Sections 7.1,7.2)

Module-4

DYNAMIC PROGRAMMING: Three basic examples, The Knapsack Problem and Memory Functions, Warshall's and Floyd's Algorithms.

THE GREEDY METHOD: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees and Codes.

Chapter 8 (Sections 8.1,8.2,8.4), Chapter 9 (Sections 9.1,9.2,9.3,9.4)

Module-5

LIMITATIONS OF ALGORITHMIC POWER: Decision Trees, P, NP, and NP-Complete Problems.

COPING WITH LIMITATIONS OF ALGORITHMIC POWER: Backtracking (n-Queens problem, Subset-sum problem), Branch-and-Bound (Knapsack problem), Approximation algorithms for NP-Hard problems (Knapsack problem).

Chapter 11 (Section 11.2, 11.3), Chapter 12 (Sections 12.1,12.2,12.3)

Course outcome (Course Skill Set)

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

1. Apply asymptotic notational method to analyze the performance of the algorithms in terms of time complexity.
2. Demonstrate divide & conquer approaches and decrease & conquer approaches to solve computational problems.
3. Make use of transform & conquer and dynamic programming design approaches to solve the given real world or complex computational problems.
4. Apply greedy and input enhancement methods to solve graph & string based computational problems.
5. Analyse various classes (P, NP and NP Complete) of problems
6. Illustrate backtracking, branch & bound and approximation methods.

Assessment Details (both CIE and SEE)

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

Continuous Internal Evaluation:

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

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

Semester-End Examination:

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

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

Suggested Learning Resources:

Textbooks

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

Reference books

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

Web links and Video Lectures (e-Resources):

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

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

- Promote real-world problem-solving and competitive problem solving through group discussions to engage students actively in the learning process.
- Encourage students to enhance their problem-solving skills by implementing algorithms and solutions through programming exercises, fostering practical application of theoretical concepts.

Assessment Methods -

1. Problem Solving Assignments (Hacker Rank/ Hacker Earth / Leadcode)
2. Gate Based Aptitude Test

ARTIFICIAL INTELLIGENCE		Semester	IV
Course Code	MVJ22AI42	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 hours Theory + 8-10 Lab slots	Total Marks	100
Credits	04	Exam Hours	
Examination nature (SEE)	Theory/		
<p>Course objectives:</p> <ul style="list-style-type: none"> ● Gain a historical perspective of AI and its foundations. ● Become familiar with basic principles of AI toward problem solving ● Get to know approaches of inference, perception, knowledge representation, and learning 			
<p>Teaching-Learning Process (General Instructions) These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. Lecturer methods (L) need not to be only traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. 2. Use of Video/Animation to explain functioning of various concepts. 3. Encourage collaborative (Group Learning) Learning in the class. 4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking. 5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 6. Introduce Topics in manifold representations. 7. Demonstrate ways to solve the same problem and encourage the students to come up with their own creative solutions. 8. Discuss application of every concept to solve the real world problems. 			
MODULE-1			
<p>Introduction: What is AI? Foundations and History of AI Intelligent Agents: Agents and environment, Concept of Rationality, The nature of environment, The structure of agents. Text book 1: Chapter 1- 1.1, 1.2, 1.3 Chapter 2- 2.1, 2.2, 2.3, 2.4</p>			
MODULE-2			
<p>Problem-solving: Problem-solving agents, Example problems, Searching for Solutions Uninformed Search Strategies: Breadth First search, Depth First Search, Iterative deepening depth first search;</p> <p>Text book 1: Chapter 3- 3.1, 3.2, 3.3, 3.4</p>			
MODULE-3			

Informed Search Strategies: Heuristic functions, Greedy best first search, A*search. Heuristic Functions
Logical Agents: Knowledge-based agents, The Wumpus world, Logic, Propositional logic, Reasoning patterns in Propositional Logic

Text book 1: Chapter 3-3.5,3.6
 Chapter 4 – 4.1, 4.2 Chapter 7- 7.1, 7.2, 7.3, 7.4, 7.5

MODULE-4

First Order Logic: Representation Revisited, Syntax and Semantics of First Order logic, Using First Order logic. **Inference in First Order Logic :**Propositional Versus First Order Inference, Unification, Forward Chaining, Backward Chaining, Resolution

Text book 1: Chapter 8- 8.1, 8.2, 8.3 Chapter 9- 9.1, 9.2, 9.3, 9.4, 9.5

MODULE-5

Uncertain Knowledge and Reasoning: Quantifying Uncertainty: Acting under Uncertainty, Basic Probability Notation, Inference using Full Joint Distributions, Independence, Baye’s Rule and its use. Wumpus World Revisited

Expert Systems: Representing and using domain knowledge, ES shells. Explanation, knowledge acquisition

Text Book 1: Chapter 13-13.1, 13.2, 13.3, 13.4, 13.5, 13.6

Text Book 2: Chapter 20

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

NOTE: Programs need to be implemented in python

Sl.N O	Experiments
1	Implement and Demonstrate Depth First Search Algorithm on Water Jug Problem
2	Implement and Demonstrate Best First Search Algorithm on Missionaries-Cannibals Problems using Python
3	Implement A* Search algorithm
4	Implement AO* Search algorithm
5	Solve 8-Queens Problem with suitable assumptions
6	Implementation of TSP using heuristic approach
7	Implementation of the problem solving strategies: either using Forward Chaining or Backward Chaining
8	Implement resolution principle on FOPL related problems
9	Implement Tic-Tac-Toe game using Python

10	Build a bot which provides all the information related to text in search box
11	Implement any Game and demonstrate the Game playing strategies

Course outcomes (Course Skill Set):

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

CO1: Apply knowledge of agent architecture, searching and reasoning techniques for different applications.

CO 2. Compare various Searching and Inferencing Techniques.

CO 3. Develop knowledge base sentences using propositional logic and first order logic

CO 4. Describe the concepts of quantifying uncertainty.

CO5: Use the concepts of Expert Systems to build applications.

Assessment Details (both CIE and SEE)

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

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

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

CIE for the practical component of the IPCC

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

- The student has to secure 40% of 25 marks to qualify in the CIE of the practical component of the IPCC.

SEE for IPCC

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

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

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

Suggested Learning Resources:

Text Books

1. Stuart J. Russell and Peter Norvig , Artificial Intelligence, 3rd Edition, Pearson,2015
2. Elaine Rich, Kevin Knight, Artificial Intelligence, 3rd edition,Tata McGraw Hill,2013

Reference:

1. George F Lugar, Artificial Intelligence Structure and strategies for complex, Pearson Education, 5th Edition, 2011
2. Nils J. Nilsson, Principles of Artificial Intelligence, Elsevier, 1980
3. Saroj Kaushik, Artificial Intelligence, Cengage learning, 2014

Web links and Video Lectures (e-Resources)

1. <https://www.kdnuggets.com/2019/11/10-free-must-read-books-ai.html>
2. <https://www.udacity.com/course/knowledge-based-ai-cognitive-systems--ud409>
3. <https://nptel.ac.in/courses/106/105/106105077/>

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

1. Group discussion on Real world examples
2. Project based learning
3. Simple strategies on gaming, reasoning and uncertainty etc

DATABASE MANAGEMENT SYSTEM		Semester	4
Course Code	MVJ22AI43	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 hours Theory + 8-10 Lab slots	Total Marks	100
Credits	04	Exam Hours	
Examination nature (SEE)	Theory		
<p>Course objectives:</p> <ul style="list-style-type: none"> ● To Provide a strong foundation in database concepts, technology, and practice. ● To Practice SQL programming through a variety of database problems. ● To Understand the relational database design principles. ● To Demonstrate the use of concurrency and transactions in database. ● To Design and build database applications for real world problems. ● To become familiar with database storage structures and access techniques. 			
<p>Teaching-Learning Process</p> <p>These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. Lecturer method (L) needs not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. 2. Use of Video/Animation to explain functioning of various concepts. 3. Encourage collaborative (Group Learning) Learning in the class. 4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking. 5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 6. Introduce Topics in manifold representations. 7. Show the different ways to solve the same problem with different circuits/logic and encourage the students to come up with their own creative ways to solve them. 8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 9. Use any of these methods: Chalk and board, Active Learning, Case Studies 			
MODULE-1		No. of Hours: 8	
<p>Introduction to Databases: Introduction, Characteristics of database approach, Advantages of using the DBMS approach, History of database applications.</p> <p>Overview of Database Languages and Architectures: Data Models, Schemas, and Instances. Three schema architecture and data independence, database languages, and interfaces, The Database System environment.</p> <p>Conceptual Data Modelling using Entities and Relationships: Entity types, Entity sets and structural constraints, Weak entity types, ER diagrams, Specialization and Generalization.</p> <p>Textbook 1:Ch 1.1 to 1.8, 2.1 to 2.6, 3.1 to 3.10 RBT: L1, L2, L3</p>			
MODULE-2		No. of Hours: 8	

Relational Model: Relational Model Concepts, Relational Model Constraints and relational database schemas, Update operations, transactions, and dealing with constraint violations.
Relational Algebra: Unary and Binary relational operations, additional relational operations (aggregate, grouping, etc.) Examples of Queries in relational algebra.
Mapping Conceptual Design into a Logical Design: Relational Database Design using ER-to-Relational mapping.

Textbook 1: Ch 5.1 to 5.3, Ch 8.1 to 8.5; Ch 9.1 to 9.2 Textbook 2: 3.5
RBT: L1, L2, L3

MODULE-3

No. of Hours:8

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

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

Textbook 1: Ch 14.1 to 14.7, Ch 6.1 to 6.5
RBT: L1, L2, L3

MODULE-4

No. of Hours:8

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

Transaction Processing: Introduction to Transaction Processing, Transaction and System concepts, Desirable properties of Transactions, Characterizing schedules based on recoverability, Characterizing schedules based on Serializability, Transaction support in SQL.

Textbook 1: Ch 7.1 to 7.3, Ch 20.1 to 20.6
RBT: L1, L2, L3

MODULE-5

No. of Hours:08

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

NOSQL Databases and Big Data Storage Systems: Introduction to NOSQL Systems, The CAP Theorem, Document-Based NOSQL Systems and MongoDB, NOSQL Key-Value Stores, Column-Based or Wide Column NOSQL Systems, NOSQL Graph Databases and Neo4j

Textbook 1:Chapter 21.1 to 21.5, Chapter 24.1 to 24.6
RBT: L1, L2, L3

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

Sl.NO	Experiments
1	<p>Create a table called Employee & execute the following.</p> <p>Employee(EMPNO,ENAME,JOB, MANAGER_NO, SAL, COMMISSION)</p> <ol style="list-style-type: none"> 1. Create a user and grant all permissions to the user. 2. Insert the any three records in the employee table contains attributes EMPNO,ENAME JOB, MANAGER_NO, SAL, COMMISSION and use rollback. Check the result. 3. Add primary key constraint and not null constraint to the employee table. 4. Insert null values to the employee table and verify the result.
2	<p>Create a table called Employee that contain attributes EMPNO,ENAME,JOB, MGR,SAL & execute the following.</p> <ol style="list-style-type: none"> 1. Add a column commission with domain to the Employee table. 2. Insert any five records into the table. 3. Update the column details of job 4. Rename the column of Employ table using alter command. 5. Delete the employee whose Empno is 105.
3	<p>Queries using aggregate functions(COUNT,AVG,MIN,MAX,SUM),Group by,Orderby.</p> <p>Employee(E_id, E_name, Age, Salary)</p> <ol style="list-style-type: none"> 1. Create Employee table containing all Records E_id, E_name, Age, Salary. 2. Count number of employee names from employeetable 3. Find the Maximum age from employee table. 4. Find the Minimum age from employeetable. 5. Find salaries of employee in Ascending Order. 6. Find grouped salaries of employees.
4	<p>Create a row level trigger for the customers table that would fire for INSERT or UPDATE or DELETE operations performed on the CUSTOMERS table. This trigger will display the salary difference between the old & new Salary.</p> <p>CUSTOMERS(ID,NAME,AGE,ADDRESS,SALARY)</p>
5	<p>Create cursor for Employee table & extract the values from the table. Declare the variables ,Open the cursor & extract the values from the cursor. Close the cursor.</p> <p>Employee(E_id, E_name, Age, Salary)</p>
6	<p>Write a PL/SQL block of code using parameterized Cursor, that will merge the data available in the newly created table N_RollCall with the data available in the table O_RollCall. If the data in the first table already exist in the second table then that data should be skipped.</p>
7	<p>Install an Open Source NoSQL Data base MangoDB & perform basic CRUD(Create, Read, Update & Delete) operations. Execute MangoDB basic Queries using CRUD operations.</p>
<p>Course outcomes (Course Skill Set): At the end of the course, the student will be able to:</p> <ul style="list-style-type: none"> ● Describe the basic elements of a relational database management system ● Design entity relationship for the given scenario. ● Apply various Structured Query Language (SQL) statements for database manipulation. ● Analyse various normalization forms for the given application. ● Develop database applications for the given real world problem. ● Understand the concepts related to NoSQL databases. 	
<p>Assessment Details (both CIE and SEE) The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum</p>	

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

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

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

CIE for the practical component of the IPCC

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

SEE for IPCC

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

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

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

Suggested Learning Resources:

Text Books:

1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.
2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill

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

Mini Project:

- Project Based Learning

Analysis & Design of Algorithms Lab		Semester	4
Course Code	MVJ22AIL 44	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	0:0:2:0	SEE Marks	50
Credits	01	Exam Hours	2
Examination type (SEE)	Practical		
Course objectives:			
<ul style="list-style-type: none"> • To design and implement various algorithms in C/C++ programming using suitable development tools to address different computational challenges. • To apply diverse design strategies for effective problem-solving. • To Measure and compare the performance of different algorithms to determine their efficiency and suitability for specific tasks. 			
Sl.No	Experiments		
1	Design and implement C/C++ Program to find Minimum Cost Spanning Tree of a given connected undirected graph using Kruskal's algorithm.		
2	Design and implement C/C++ Program to find Minimum Cost Spanning Tree of a given connected undirected graph using Prim's algorithm.		
3	a. Design and implement C/C++ Program to solve All-Pairs Shortest Paths problem using Floyd's algorithm. b. Design and implement C/C++ Program to find the transitive closure using Warshal's algorithm.		
4	Design and implement C/C++ Program to find shortest paths from a given vertex in a weighted connected graph to other vertices using Dijkstra's algorithm.		
5	Design and implement C/C++ Program to obtain the Topological ordering of vertices in a given digraph.		
6	Design and implement C/C++ Program to solve 0/1 Knapsack problem using Dynamic Programming method.		
7	Design and implement C/C++ Program to solve discrete Knapsack and continuous Knapsack problems using greedy approximation method.		
8	Design and implement C/C++ Program to find a subset of a given set $S = \{s_1, s_2, \dots, s_n\}$ of n positive integers whose sum is equal to a given positive integer d .		
9	Design and implement C/C++ Program to sort a given set of n integer elements using Selection Sort method and compute its time complexity. Run the program for varied values of $n > 5000$ and record the time taken to sort. Plot a graph of the time taken versus n . The elements can be read from a file or can be generated using the random number generator.		
10	Design and implement C/C++ Program to sort a given set of n integer elements using Quick Sort method and compute its time complexity. Run the program for varied values of $n > 5000$ and record the time taken to sort. Plot a graph of the time taken versus n . The elements can be read from a file or can be generated using the random number generator.		
11	Design and implement C/C++ Program to sort a given set of n integer elements using Merge Sort method and compute its time complexity. Run the program for varied values of $n > 5000$, and record the time taken to sort. Plot a graph of the time taken versus n . The elements can be read from a file or can be generated using the random number generator.		
12	Design and implement C/C++ Program for N Queen's problem using Backtracking.		

Course outcomes (Course Skill Set):

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

1. Develop programs to solve computational problems using suitable algorithm design strategy.
2. Compare algorithm design strategies by developing equivalent programs and observing running times for analysis (Empirical).
3. Make use of suitable integrated development tools to develop programs
4. Choose appropriate algorithm design techniques to develop solution to the computational and complex problems.
5. Demonstrate and present the development of program, its execution and running time(s) and record the results/inferences.

Assessment Details (both CIE and SEE)

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

Continuous Internal Evaluation (CIE):

CIE marks for the practical course are **50 Marks**.

The split-up of CIE marks for record/ journal and test are in the ratio **60:40**.

- Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to **30 marks** (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The marks scored shall be scaled down to **20 marks** (40% of the maximum marks).

The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student.

Semester End Evaluation (SEE):

- SEE marks for the practical course are 50 Marks.

- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.
- The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University.
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. **OR** based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question (experiment) from the questions lot prepared by the examiners jointly.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.

General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)

Change of experiment is allowed only once and 15% of Marks allotted to the procedure part are to be made zero.

The minimum duration of SEE is 02 hours

Suggested Learning Resources:

- Virtual Labs (CSE): <http://cse01-iiith.vlabs.ac.in/>

DISCRETE MATHEMATICAL STRUCTURES		Semester	IV
Course Code	MVJ22AI4 51	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	2:2:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Examination type (SEE)	Theory		
<p>Course objectives:</p> <ol style="list-style-type: none"> To help students to understand discrete and continuous mathematical structures. To impart basics of relations and functions. To facilitate students in applying principles of Recurrence Relations to find the generating functions and solve the Recurrence relations. To have the knowledge of groups and their properties to understand the importance of algebraic properties relative to various number systems. 			
<p>Teaching-Learning Process Pedagogy (General Instructions): These are sample Strategies, teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the delivered lessons shall develop students' theoretical and applied Mathematical skills. State the need for Mathematics with Engineering Studies and Provide real-life examples. Support and guide the students for self-study. You will assign homework, grading assignments and quizzes, and documenting students' progress. Encourage the students to group learning to improve their creative and analytical skills. Show short related video lectures in the following ways: <ul style="list-style-type: none"> As an introduction to new topics (pre-lecture activity). As a revision of topics (post-lecture activity). As additional examples (post-lecture activity). As an additional material of challenging topics (pre-and post-lecture activity). As a model solution for some exercises (post-lecture activity). 			
Module-1: Fundamentals of Logic			
Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference. The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems. <p style="text-align: right;">(8 hours)</p> <p>(RBT Levels: L1, L2 and L3)</p>			
Module-2: Properties of the Integers			
Mathematical Induction, The Well Ordering Principle – Mathematical Induction, Recursive Definitions. <p>Fundamental Principles of Counting: The Rules of Sum and Product, Permutations, Combinations – The Binomial Theorem, Combinations with Repetition. <p style="text-align: right;">(8 Hours)</p> (RBT Levels: L1, L2 and L3)</p>			
Module-3: Relations and Functions			
Cartesian Products and Relations, Functions – Plain and One-to-One, Onto Functions. The Pigeon-hole Principle, Function Composition and Inverse Functions. <p>Properties of Relations, Computer Recognition – Zero-One Matrices and Directed Graphs, Partial Orders – Hasse Diagrams, Equivalence Relations and Partitions. <p style="text-align: right;">(8 hours)</p> (RBT Levels: L1, L2 and L3)</p>			
Module-4: The Principle of Inclusion and Exclusion			

The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements – Nothing is in its Right Place, Rook Polynomials.

Recurrence Relations: First Order Linear Recurrence Relation, The Second Order Linear Homogeneous Recurrence Relation with Constant Coefficients. **(8 Hours)**

(RBT Levels: L1, L2 and L3)

Module-5: Introduction to Groups Theory

Definitions and Examples of Particular Groups Klein 4-group, Additive group of Integers modulo n , Multiplicative group of Integers modulo- p and permutation groups, Properties of groups, Subgroups, cyclic groups, Cosets, Lagrange's Theorem. **(8 Hours)**

(RBT Levels: L1, L2 and L3)

Course outcome (Course Skill Set)

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

1. Apply concepts of logical reasoning and mathematical proof techniques in proving theorems and statements.
2. Demonstrate the application of discrete structures in different fields of computer science.
3. Apply the basic concepts of relations, functions and partially ordered sets for computer representations.
4. Solve problems involving recurrence relations and generating functions.
5. Illustrate the fundamental principles of Algebraic structures with the problems related to computer science & engineering.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE, the minimum passing mark is 35% of the maximum marks (18 out of 50 marks). The student is declared as a pass in the course if he/she secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- There are 25 marks for the CIE's Assignment component and 25 for the Internal Assessment Test component.
- Each test shall be conducted for 25 marks. The first test will be administered after 40-50% of the coverage of the syllabus, and the second test will be administered after 85-90% of the coverage of the syllabus. The average of the two tests shall be scaled down to 25 marks
- Any two assignment methods mentioned in the 220B2.4, if an assignment is project-based then only one assignment for the course shall be planned. The schedule for assignments shall be planned properly by the course teacher. The teacher should not conduct two assignments at the end of the semester if two assignments are planned. Each assignment shall be conducted for 25 marks. (If two assignments are conducted then the sum of the two assignments shall be scaled down to 25 marks)

The final CIE marks of the course out of 50 will be the sum of the scale-down marks of tests and assignment/s marks.

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

Semester-End Examination:

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

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

Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:

Books (Name of the author/Title of the Book/Name of the publisher/Edition and Year)

Text Books:

1. **Ralph P. Grimaldi, B V Ramana: "Discrete Mathematical Structures an Applied Introduction"**, 5th Edition, Pearson Education, 2004.
2. **Ralph P. Grimaldi: "Discrete and Combinatorial Mathematics"**, 5th Edition, Pearson Education. 2004.

Reference Books:

1. **Basavaraj S Anami and Venakanna S Madalli: "Discrete Mathematics – A Concept-based approach"**, Universities Press, 2016
2. **Kenneth H. Rosen: "Discrete Mathematics and its Applications"**, 6th Edition, McGraw Hill, 2007.
3. **Jayant Ganguly: "A Treatise on Discrete Mathematical Structures"**, Sanguine-Pearson, 2010.
4. **D.S. Malik and M.K. Sen: "Discrete Mathematical Structures Theory and Applications"**, Latest Edition, Thomson, 2004.
5. **Thomas Koshy: "Discrete Mathematics with Applications"**, Elsevier, 2005, Reprint 2008.

Web links and Video Lectures (e-Resources):

- <http://nptel.ac.in/courses.php?disciplineID=111>
- [http://www.class-central.com/subject/math\(MOOCs\)](http://www.class-central.com/subject/math(MOOCs))
- <http://academicearth.org/>
- VTU e-Shikshana Program
- VTU EDUSAT Program.
- <http://www.themathpage.com/>
- <http://www.abstractmath.org/>
- <http://www.ocw.mit.edu/courses/mathematics/>

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

- Quizzes
- Assignments
- Seminar

METRIC SPACES		Semester	IV
Course Code	MVJ22A I452	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	2:2:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Examination type (SEE)	Theory		
<p>Course objectives:</p> <ul style="list-style-type: none"> • Provide insight into the theory of sets • Learn basic concepts of metric spaces • Understand the concepts of connected sets and compact spaces 			
<p>Teaching-Learning Process Pedagogy (General Instructions): These are sample Strategies, teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the delivered lessons shall develop students' theoretical and applied Mathematical skills. 2. State the need for Mathematics with Engineering Studies and Provide real-life examples. 3. Support and guide the students for self-study. 4. You will assign homework, grading assignments and quizzes, and documenting students' progress. 5. Encourage the students to group learning to improve their creative and analytical skills. 6. Show short related video lectures in the following ways: <ul style="list-style-type: none"> • As an introduction to new topics (pre-lecture activity). • As a revision of topics (post-lecture activity). • As additional examples (post-lecture activity). • As an additional material of challenging topics (pre-and post-lecture activity). • As a model solution for some exercises (post-lecture activity). 			
Module-1: Theory of Sets			
Finite and infinite sets, countable and uncountable sets, cardinality of sets, Schroder-Bernstein theorem, cantor's theorem, Order relation in cardinal numbers, Arithmetic of cardinal numbers, Partially ordered set, Zorn's lemma and axioms of choice, various set-theoretic paradoxes. <p style="text-align: right;">(8 hours)</p>			
(RBT Levels: L1, L2 and L3)			
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation		
Module-2: Concepts in Metric Spaces			
Definition and examples of metric spaces, Open spheres and Closed spheres, Neighborhoods, Open sets, Interior, Exterior and boundary points, Closed sets, Limit points and isolated points, Interior and closure of a set, Boundary of a set, Bounded sets, Distance between two sets, Diameter of a set. <p style="text-align: right;">(8 hours)</p>			
(RBT Levels: L1, L2 and L3)			
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation		
Module-3: Complete Metric Spaces and Continuous Functions			

<p>Cauchy and Convergent sequences, Completeness of metric spaces, Cantor's intersection theorem, Dense sets and separable spaces, Nowhere dense sets and Baire's category theorem, continuous and uniformly continuous functions, Homeomorphism. Banach contraction principle. (8 hours) (RBT Levels: L1, L2 and L3)</p>	
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation
Module-4: Compactness	
<p>Compact spaces, Sequential compactness, Bolzano-Weierstrass property, Compactness and finite intersection property, Heine-Borel theorem, Totally bounded set, equivalence of compactness and sequential compactness. (8 hours) (RBT Levels: L1, L2 and L3)</p>	
Module-5: Connectedness	
<p>Separated sets, Disconnected and connected sets, components, connected subsets of R, Continuous functions on connected sets. Local connectedness and arc-wise connectedness. (8 hours) (RBT Levels: L1, L2 and L3)</p>	
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation
<p>Course outcome (Course Skill Set) At the end of the course, the student will be able to:</p> <ol style="list-style-type: none"> 1. Explain basic facts about the cardinality of a set and various set-theoretic paradoxes. 2. Apply the concepts of open and closed spheres and bounded sets to solve problems. 3. Demonstrate standard concepts of metric spaces and their properties. 4. Identify the continuity of a function defined on metric spaces and homomorphism. 	
<p>Assessment Details (both CIE and SEE) The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE, the minimum passing mark is 35% of the maximum marks (18 out of 50 marks). The student is declared as a pass in the course if he/she secures a minimum of 40% (40 marks out of 100) in the total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.</p>	
<p>Continuous Internal Evaluation:</p> <ul style="list-style-type: none"> • There are 25 marks for the CIE's Assignment component and 25 for the Internal Assessment Test component. • Each test shall be conducted for 25 marks. The first test will be administered after 40-50% of the coverage of the syllabus, and the second test will be administered after 85-90% of the coverage of the syllabus. The average of the two tests shall be scaled down to 25 marks • Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The schedule for assignments shall be planned properly by the course teacher. The teacher should not conduct two assignments at the end of the semester if two assignments are planned. Each assignment shall be conducted for 25 marks. (If two assignments are conducted then the sum of the two assignments shall be scaled down to 25 marks) 	

- The final CIE marks of the course out of 50 will be the sum of the scale-down marks of tests and assignment/s marks.

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

Semester-End Examination:

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

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

Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:

Books (Name of the author/Title of the Book/Name of the publisher/Edition and Year)

Text Books

1. P.K. Jain & Khalil Ahamad, "*Metric Spaces*". Narosa, 2019.
2. Micheal O; Searcoid, "Metric spaces". Springer-Verlag, 2009.

Reference Books:

1. Satish Shirali & Harikishan L. Vasudeva, "*Metric Spaces*", Springer-Verlag, 2006.
2. E.T. Copson, "*Metric spaces*", Cambridge University Press, 1988.
3. P.R. Halmos, "*Naive Set Theory*". Springer, 1974.
4. S. Kumaresan, "*Topology of Metric spaces*", 2nd edition, Narosa, 2011.
5. G.F. Simmons, "Introduction to Topology and Modern Analysis". McGraw-Hill, 2004.

Web links and Video Lectures (e-Resources):

- <http://nptel.ac.in/courses.php?disciplineID=111>
- [http://www.class-central.com/subject/math\(MOOCs\)](http://www.class-central.com/subject/math(MOOCs))
- <http://academicearth.org/>
- VTU e-Shikshana Program
- VTU EDUSAT Program.

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

- Quizzes
- Assignments
- Seminar

OPTIMIZATION TECHNIQUE		Semester	IV
Course Code	BCS405C	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	2:2:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Examination type (SEE)	Theory		
<p>Course objectives: The objectives of the course are to facilitate the learners to:</p> <ul style="list-style-type: none"> • Appreciate the importance of linear algebra in computer science and allied engineering science. • Gain the knowledge of linear algebra tools and concepts to implement them in their core domain. • Improve their mathematical thinking and acquire skills required for sustained lifelong learning. 			
<p>Teaching-Learning Process Pedagogy (General Instructions): These are sample Strategies, teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the delivered lessons shall develop students' theoretical and applied Mathematical skills. 2. State the need for Mathematics with Engineering Studies and Provide real-life examples. 3. Support and guide the students for self-study. 4. You will assign homework, grading assignments and quizzes, and documenting students' progress. 5. Encourage the students to group learning to improve their creative and analytical skills. 6. Show short related video lectures in the following ways: <ul style="list-style-type: none"> • As an introduction to new topics (pre-lecture activity). • As a revision of topics (post-lecture activity). • As additional examples (post-lecture activity). • As an additional material of challenging topics (pre-and post-lecture activity). • As a model solution of some exercises (post-lecture activity). 			
Module-1: VECTOR CALCULUS			
Functions of several variables, Differentiation and partial differentials, gradients of vector-valued functions, gradients of matrices, useful identities for computing gradients, linearization and multivariate Taylor series. (8 hours) (RBT Levels: L1, L2 and L3)			
Module-2: APPLICATIONS OF VECTOR CALCULUS			
Backpropagation and automatic differentiation, gradients in a deep network, The Gradient of Quadratic Cost, Descending the Gradient of Cost, The Gradient of Mean Squared Error. <p style="text-align: right;">(8 hours)</p> (RBT Levels: L1, L2 and L3)			
Module-3: Convex Optimization-1			

<p>Local and global optima, convex sets and functions separating hyperplanes, application of Hessian matrix in optimization, Optimization using gradient descent, Sequential search 3-point search and Fibonacci search.</p> <p>(8 hours) (RBT Levels: L1, L2 and L3)</p>
<p>Module-4: Convex Optimization-2</p>
<p>Unconstrained optimization -Method of steepest ascent/descent, NR method, Gradient descent, Mini batch gradient descent, Stochastic gradient descent. (8 hours)</p> <p>(RBT Levels: L1, L2 and L3)</p>
<p>Module-5: Advanced Optimization</p>
<p>Momentum-based gradient descent methods: Adagrad, RMSprop and Adam. Non-Convex Optimization: Convergence to Critical Points, Saddle-Point methods.</p> <p>(8 hours) (RBT Levels: L1, L2 and L3)</p>
<p>Course outcome (Course Skill Set)</p> <p>At the end of the course, the student will be able to:</p> <ol style="list-style-type: none"> 1. Apply the concepts of vector calculus to solve the given problem. 2. Apply the concepts of partial differentiation in machine learning and deep neural networks. 3. Analyze the convex optimization algorithms and their importance in computer science & engineering. 4. Apply the optimization algorithms to solve the problem. 5. Analyze the advanced optimization algorithms for machine learning .
<p>Assessment Details (both CIE and SEE)</p> <p>The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE, the minimum passing mark is 35% of the maximum marks (18 out of 50 marks). The student is declared as a pass in the course if he/she secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.</p>

Continuous Internal Evaluation:

- There are 25 marks for the CIE's Assignment component and 25 for the Internal Assessment Test component.
- Each test shall be conducted for 25 marks. The first test will be administered after 40-50% of the coverage of the syllabus, and the second test will be administered after 85-90% of the coverage of the syllabus. The average of the two tests shall be scaled down to 25 marks
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The schedule for assignments shall be planned properly by the course teacher. The teacher should not conduct two assignments at the end of the semester if two assignments are planned. Each assignment shall be conducted for 25 marks. (If two assignments are conducted then the sum of the two assignments shall be scaled down to 25 marks)
- The final CIE marks of the course out of 50 will be the sum of the scale-down marks of tests and assignment/s marks.

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

Semester-End Examination:

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

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

Marks scored shall be proportionally reduced to 50 marks.

Suggested Learning Resources:

Books (Name of the author/Title of the Book/Name of the publisher/Edition and Year)

Text Books:

1. Mathematics for Machine learning, Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong, 2020, Cambridge University Press.
2. S. Bubeck, Convex Optimization: Algorithms and Complexity, Foundations and Trends in Optimization, 2015.
3. S. Boyd, N. Parikh, and E. Chu, "Distributed optimization and statistical learning via the alternating direction method of multipliers", Foundations and Trends in Machine Learning, Now Publishers Inc.

Reference Books:

1. Linear Algebra and Optimization for Machine Learning, Charu C. Aggarwal, Springer, 2020.
2. A. Beck, First-Order Methods in Optimization, MOS-SIAM Series on Optimization, 2017.
3. F. Bach, "Learning with Submodular Functions: A Convex Optimization Perspective", Foundations and Trends in Machine Learning, Now Publishers Inc.

Web links and Video Lectures (e-Resources):

- <https://ocw.mit.edu/courses/mathematics/18-06sc-linear-algebra-fall-2011/index.htm>
- <https://www.math.ucdavis.edu/~linear/linear.pdf>
- <https://www.coursera.org/learn/linear-algebra-machine-learning>
- <https://nptel.ac.in/syllabus/111106051/>
- https://github.com/epfml/OptML_course
- <https://www.youtube.com/playlist?list=PL4O4bXkI-fAeYrsBqTUYn2xMjJAqlFQzX>

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

- Quizzes
- Assignments
- Seminar

ALGORITHMIC GAME THEORY		Semester	IV
Course Code	MVJ22AI 454	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	2:2:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Examination type (SEE)	Theory		
<p>Course objectives:</p> <ul style="list-style-type: none"> • Comprehend the basics of strategic gaming and mixed strategic equilibrium. • Enable students to develop skills on extensive gaming strategies. • Analyze and discuss various gaming models. • Illustrate some real-time situations. 			
<p>Teaching-Learning Process Pedagogy (General Instructions): These are sample Strategies, teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the delivered lessons shall develop students' theoretical and applied Mathematical skills. 2. State the need for Mathematics with Engineering Studies and Provide real-life examples. 3. Support and guide the students for self-study. 4. You will assign homework, grading assignments and quizzes, and documenting students' progress. 5. Encourage the students to group learning to improve their creative and analytical skills. 6. Show short related video lectures in the following ways: <ul style="list-style-type: none"> • As an introduction to new topics (pre-lecture activity). • As a revision of topics (post-lecture activity). • As additional examples (post-lecture activity). • As an additional material of challenging topics (pre-and post-lecture activity). • As a model solution for some exercises (post-lecture activity). 			
Module-1			
<p>Introduction to Strategic Games: What is game theory? The theory of rational choice, Strategic games; Examples: The prisoner's dilemma, Bach or Stravinsky, Matching pennies; Nash equilibrium; Examples of Nash equilibrium; Best response functions; Dominated actions.</p> <p style="text-align: right;">(8 hours)</p> <p>(RBT Levels: L1, L2 and L3)</p>			
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation		
Module-2			
<p>Introduction; Strategic games in which players may randomize; Mixed strategy Nash equilibrium; Dominated actions; Pure equilibrium when randomization is allowed. Illustration: Expert Diagnosis; Equilibrium in a single population.</p> <p style="text-align: right;">(8 hours)</p> <p>(RBT Levels: L1, L2 and L3)</p>			
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation		
Module-3			

<p>Extensive games with perfect information; Strategies and outcomes; Nash equilibrium; Sub-game perfect equilibrium; Finding sub-game perfect equilibria of finite horizon games; Backward induction; Illustrations: The ultimatum game, Stackelberg's model of duopoly. (8 hours) (RBT Levels: L1, L2 and L3)</p>	
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation
Module-4	
<p>Bayesian Games, Motivational examples; General definitions; Two examples concerning information; Illustrations: Cournot's duopoly game with imperfect information, Providing a public good; Auctions: Auctions with an arbitrary distribution of valuations. (8 hours) (RBT Levels: L1, L2 and L3)</p>	
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation
Module-5	
<p>Competative Games: Strictly competitive games and maximization. Repeated games: The main idea; Preferences; Repeated games; Finitely and infinitely repeated Prisoner's dilemma; Strategies in an infinitely repeated Prisoner's dilemma; Nash equilibrium of an infinitely repeated Prisoner's dilemma, Nash equilibrium payoffs of an infinitely repeated Prisoner's dilemma. (8 hours) (RBT Levels: L1, L2 and L3)</p>	
Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation
<p>Course outcome (Course Skill Set) At the end of the course, the student will be able to:</p> <ol style="list-style-type: none"> 1. Interpret the basics of strategic gaming and extensive games. 2. Analyze gaming strategies on real-time incidence. 3. Develop the models of gaming on real-time incidence. 4. Apply game theory in the real world problems. 	
<p>Assessment Details (both CIE and SEE) The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE, the minimum passing mark is 35% of the maximum marks (18 out of 50 marks). The student is declared as a pass in the course if he/she secures a minimum of 40% (40 marks out of 100) in the total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.</p>	
<p>Continuous Internal Evaluation:</p> <ul style="list-style-type: none"> • There are 25 marks for the CIE's Assignment component and 25 for the Internal Assessment Test component. • Each test shall be conducted for 25 marks. The first test will be administered after 40-50% of the coverage of the syllabus, and the second test will be administered after 85-90% of the coverage of the syllabus. The average of the two tests shall be scaled down to 25 marks 	

- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the course shall be planned. The schedule for assignments shall be planned properly by the course teacher. The teacher should not conduct two assignments at the end of the semester if two assignments are planned. Each assignment shall be conducted for 25 marks. (If two assignments are conducted then the sum of the two assignments shall be scaled down to 25 marks)
- The final CIE marks of the course out of 50 will be the sum of the scale-down marks of tests and assignment/s marks.

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

Semester-End Examination:

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

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

Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:

Books (Name of the author/Title of the Book/Name of the publisher/Edition and Year)

Text Books:

1. **Martin Osborne: "An Introduction to Game Theory"**, Oxford University Press, First Indian Edition, 2009, 7th impression, ISBN – 0195128958.

Reference Books:

1. **Roger B. Myerson: "Analysis of Conflict Game Theory"**, Re-print Edition, Harvard University Press, 2008, ISBN – 978-0674341166.
2. **Frederick S. Hillier and Gerald J. Lieberman: "Introduction to Operations Research, Concepts and Cases"**, 9th Edition; Tata McGraw Hill, 2010, ISBN – 0073376299.
3. **Joel Watson: "An Introduction to Game Theory" Strategy**, 2nd Edition, W.W. Norton & Company, 2007, ISBN – 9780393929348.

Web links and Video Lectures (e-Resources):

- <http://nptel.ac.in/courses.php?disciplineID=111>
- [http://www.class-central.com/subject/math\(MOOCs\)](http://www.class-central.com/subject/math(MOOCs))
- <http://academicearth.org/>
- VTU e-Shikshana Program
- VTU EDUSAT Program.

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

- Quizzes
- Assignments
- Seminar

