Semester: IV		
Complex Variables and Numerical Methods (Theory)		
Course Code	**MVJ21MA41A**	**CIE Marks: 50**
Credits	**L:T:P:: 2:2:0**	**SEE Marks: 50**
Hours	**20L+20T**	**SEE Duration: 3 Hrs.**
Course Learning Objectives: The students will be able to		
1	Understand the concepts of Complex variables and transformation for solving Engineering Problems.	
2	Understand the concepts of complex integration, Poles and Residuals in the stability analysis of engineering problems.	
3	Apply the concept to find extremal of functionals.	
4	Solve initial value problems using appropriate numerical methods.	
5	Students learn to obtain solutions of ordinary and partial differential equations numerically.	

UNIT-I	
Complex variables - I: Functions of complex variables, Analytic function, Cauchy-Riemann Equations in Cartesian and polar coordinates, Consequences of Cauchy-Riemann Equations, Construction of analytic functions (Using Milne-Thomson method). **Transformations:** Bilinear Transformation, Conformal transformation, Discussion of the transformations $w = z^2$, $w = e^z$ and $w = z + \frac{a}{z}, (z \neq 0)$. **Self Study topic :** Harmonic function and its properties **Web Link and Video Lectures:** https://nptel.ac.in/courses/111103070	8 Hrs
UNIT-II	
Complex variables-II: Complex integration - Cauchy theorem, Cauchy's Integral Theorem-Problems, Taylor & Laurent series- Problems, Singularities, Types of Singularities, Poles, Residues-definitions, Cauchy residue theorem - Problems. **Self Study topic:** Consequences of Cauchy's theorem, Cauchy residue theorem. **Web Link and Video Lectures:** https://nptel.ac.in/courses/111103070	8 Hrs
UNIT-III	
Numerical methods-I:	8 Hrs

Numerical solution of Ordinary Differential Equations of first order and first degree, Taylor's series method, Modified Euler's method, Runge-Kutta method of fourth order, Milne's and Adam-Bashforth Predictor and Corrector method. **Web Link and Video Lectures:** https://nptel.ac.in/courses/127106019	
UNIT-IV	
Numerical methods-II: Numerical solution of Ordinary Differential Equations of second order: Runge-Kutta method of fourth order, Milne's Predictor and Corrector method. **Calculus of variations:** Variation of function and Functional, variational problems, Euler's equation, Geodesics. **Applications** : Hanging Chain problem. **Self Study topic :** Adam-Bashforth Predictor and Corrector method. **Web Link and Video Lectures:** https://nptel.ac.in/courses/127106019 https://nptel.ac.in/courses/111107103	8 Hrs
UNIT-V	
Numerical methods-III: Numerical solution of Partial Differential Equations: Introduction, Finite difference approximations to derivatives, Explicit methods-Numerical Solution of Laplace Equation, Numerical solution of one-dimensional heat equation by Bender - Schmidt's method and by Crank-Nicholson Method, Implicit method-Numerical solution of one-dimensional wave equation. **Self Study topic:** Classification of Partial differential equations, Parabolic, Elliptic and Hyperbolic equations. **Web Link and Video Lectures:** https://nptel.ac.in/courses/111107063	8 Hrs

Course Outcomes: After completing the course, the students will be able to	
CO1	State and prove Cauchy - Riemann equation with its consequences and demonstrate Con-formal Transformation.
CO2	Illustrate Complex Integration using Cauchy's Integral theorem, Cauchy's Integral formula and Cauchy's Residue theorem.
CO3	Identify appropriate numerical methods to solve ODE.
CO4	Determine the extremals of functionals and solve the simple problems of the calculus of variations.

CO5	Choose appropriate numerical methods to solve Partial Differential Equations.

Reference Books	
1.	B.S. Grewal, "Higher Engineering Mathematics" Khanna Publishers, 43rd Edition, 2013.
2.	Erwin Kreyszig, "Advanced Engineering Mathematics", Wiley-India publishers, 10th edition,2014.
3.	Ramana B. V., "Higher Engineering Mathematics", Tata Mc Graw-Hill, 2006.
4.	Bali N. P. & Manish Goyal, "A text book of Engineering Mathematics", Laxmi Publications, 8th Edition.

Continuous Internal Evaluation (CIE):

Theory for 50 Marks

CIE is executed by way of quizzes (Q), tests (T) and assignments. A minimum of three quizzes are conducted along with tests. Test portion is evaluated for 50 marks and quiz is evaluated for 10 marks. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three (conduct additional quizzes and take best three). The three tests are conducted for 50 marks each and the average of all the tests are calculated for 50. The marks for the assignments are 20 (2 assignments for 10 marks each). The marks obtained in test, quiz and assignment are added to get marks out of 100 and report CIE for 50 marks.

Semester End Examination (SEE):

Total marks: 50+50=100

SEE for 50 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the entire syllabus. Part – B Students have to answer five questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have a maximum of three sub divisions. Each unit will have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom's taxonomy level.

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	0	3	0	0	0	0	0	0	1	1
CO2	3	3	0	3	0	0	0	0	0	0	1	0
CO3	3	2	0	2	0	0	0	0	0	0	0	0
CO4	3	3	0	3	0	0	0	0	0	0	0	1
CO5	3	3	0	3	0	0	0	0	0	0	1	0

High-3, Medium-2, Low-1

Semester: IV		
MICRO CONTROLLER AND EMBEDDED SYSTEMS (Theory)		
Course Code: MVJ21CS/CG/AI42		**CIE Marks:100**
Credits: L:T:P:S: 3:0:0:0		**SEE Marks: 100**

Hours: 40L+26T		SEE Duration: 3 Hrs
Course Learning Objectives: The students will be able to		
1	Explain the fundamentals of ARM based system, basic hardware components, selection methods and attributes of an ARM Controller.	
2	Program ARM controller using the various instructions.	
3	Explain the fundamentals of Exception, Interrupt Handling and Memory Management Unit of ARM Controller.	
4	Identify the Embedded System Design applications.	
5	Explain the real time operating system for the embedded system design.	

UNIT-I	
Microprocessors versus Microcontrollers, ARM Embedded Systems: The RISC design philosophy, The ARM Design Philosophy, Embedded System Hardware, Embedded System Software. **ARM Processor Fundamentals**: Registers, Current Program Status Register, Pipeline, Exceptions, Interrupts, and the Vector Table , Core Extensions	**8 Hrs**
UNIT-II	
Introduction to the ARM Instruction Set : Data Processing Instructions , Programme Instructions, Software Interrupt Instructions, Program Status Register Instructions, Coprocessor Instructions, Loading Constants **ARM programming using Assembly language:** Writing Assembly code, Profiling and cycle counting, instruction scheduling	**8 Hrs**
UNIT-III	
Exception, Interrupt Handling : Exception handling, Interrupts, Interrupt handling Schemes **Memory Management Unit :** The Memory Hierarchy and Cache Memory, Cache Architecture, Cache Policy, Moving from MPU to an MMU, How Virtual Memory Works, Details of ARM MMU	**8 Hrs**
UNIT-IV	
Embedded System Components: Embedded Vs General computing system, History of embedded systems, Classification of Embedded systems, Major applications areas of embedded systems, purpose of embedded systems Core of an Embedded System including all types of processor/controller, Memory, Sensors, Actuators, LED, 7 segment LED display, stepper motor, Keyboard, Push button switch, Communication Interface (on board and external types), Embedded firmware, Other system components.	**8 Hrs**
UNIT-V	
Real Time Operating System (RTOS) based Embedded System Design: Operating System basics, Types of operating systems, Task, process and threads (Only POSIX Threads with an example program), Thread pre-emption,	**8 Hrs**

Multiprocessing and Multitasking, Task Communication (without any program), Task synchronization issues – Racing and Deadlock, Concept of Binary and counting semaphores (Mutex example without any program), How to choose an RTOS	

Course Outcomes: After completing the course, the students will be able to	
CO1	Describe the architectural features and instructions of ARM microcontroller
CO2	Develop Assembly Programs in ARM for Embedded applications.
CO3	Describe the fundamentals of Exception, Interrupt Handling and Memory Management Unit of ARM Controller
CO4	Interface external devices and I/O with ARM microcontroller.
CO5	Demonstrate the need of real time operating system for embedded system applications

Reference Books	
1.	Andrew N Sloss, Dominic Symes and Chris Wright, ARM system developer's guide, Elsevier, Morgan Kaufman publishers, 2008.
2.	Shibu K V, "Introduction to Embedded Systems", Tata McGraw Hill Education, Private Limited, 2nd Edition.
3.	Raghunandan.G.H, Microcontroller (ARM) and Embedded System, Cengage learning Publication, 2019
4.	The Insider's Guide to the ARM7 Based Microcontrollers, Hitex Ltd., 1st edition, 2005.

Continuous Internal Evaluation (CIE):

Theory for 50 Marks

CIE is executed by way of quizzes (Q), tests (T) and assignments. A minimum of three quizzes are conducted along with tests. Test portion is evaluated for 50 marks and quiz is evaluated for 10 marks. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three (conduct additional quizzes and take best three). The three tests are conducted for 50 marks each and the average of all the tests are calculated for 50. The marks for the assignments are 20 (2 assignments for 10 marks each). The marks obtained in test, quiz and assignment are added to get marks out of 100 and report CIE for 50 marks.

Semester End Examination (SEE):

Total marks: 50+50=100

SEE for 50 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the entire syllabus. Part – B Students have to answer five questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have a maximum of

three sub divisions. Each unit will have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom's taxonomy level.

CO-PO Mapping												
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	**3**	**1**	**2**	**1**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
CO2	**3**	**2**	**1**	**3**	**3**	**2**	**-**	**-**	**2**	**-**	**1**	**-**
CO3	**3**	**2**	**1**	**3**	**-**	**2**	**-**	**-**	**2**	**-**	**-**	**-**
CO4	**3**	**3**	**2**	**3**	**3**	**2**	**-**	**-**	**2**	**2**	**2**	**-**
CO5	**3**	**2**	**3**	**3**	**3**	**2**	**-**	**-**	**2**	**2**	**2**	**2**

High-3, Medium-2, Low-1

Semester: IV		
COMPUTER ORGANIZATION AND ARCHITECTURE (Theory)		
Course Code: MVJ21CS/CG/AI43		**CIE Marks:100**
Credits: L:T:P:S: 3:0:0:0		**SEE Marks: 100**
Hours: 40L		**SEE Duration: 3 Hrs**
Course Learning Objectives: The students will be able to		
1	Learn the basic structure and operations of a computer.	
2	Learn the arithmetic and logic unit.	

3	Learn the different ways of communication with I/O devices & memories, memory hierarchies, cache memories and virtual memories.
4	Understand & implement arithmetic process.
5	Understand the processor and pipelining concepts.
6	Understand parallelism and multi-core processors.

UNIT-I	
Basic Structure of Computers: Basic Operational Concepts, Bus Structures, Performance –Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. **Machine Instructions and Programs:** Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions. **Arithmetic:** Numbers, Arithmetic Operations and Characters, Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed Operand Multiplication, Fast Multiplication, Integer Division. **Video link :** https://nptel.ac.in/courses/106105163/	**8 Hrs**
UNIT-II	
Input/output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCI Bus, SCSI Bus, USB **Videolink:**https://www.youtube.com/watch?v=RkAE4zE4uSE&list=PL13FD5F00C21BBC0B&index=11	**8 Hrs**
UNIT-III	
Memory: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Types of cache ,Cache miss management Mapping Functions, Replacement Algorithms, Performance Considerations,(ARM Cache and Pentium cache). **Video link :** https://nptel.ac.in/courses/106105163/	**8 Hrs**
UNIT-IV	
Processor : A Basic MIPS implementation – Building a Data path – Control Implementation Scheme –Pipelining – Pipelined data path and control – Handling Data Hazards & Control Hazards –Exceptions.	**8 Hrs**

Video link: https://nptel.ac.in/courses/106106166/	
UNIT-V	
Parallelism: Parallel processing challenges –Flynn's classification – SISD, MIMD, SIMD, SPMD, and Vector Architectures - Hardware multithreading – Multi-core processors and other Shared Memory Multiprocessors - Introduction to Graphics Processing Units, Clusters, Warehouse Scale Computers and other Message-Passing Multiprocessors. Video link: https://nptel.ac.in/courses/106102114/	**8 Hrs**

Course Outcomes: After completing the course, the students will be able to	
CO1	Explain the basic organization of a computer system.
CO2	Demonstrate functioning of different sub systems, such as processor, Input/output, and memory.
CO3	Design and analyses simple arithmetic and logical units.
CO4	Illustrate hardwired control and micro programmed control, pipelining, embedded and other Computing systems.
CO5	Design and analyses of simple Parallelism and Multithread.

Reference Books	
1.	Carl Hamacher, Zvonko Vranesic, SafwatZaky, Computer Organization, 5th Edition, Tata McGraw Hill, 2002. (Listed topics only from Chapters 1, 2, 4, 5, and 6).
2.	David A. Patterson and John L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, Fifth Edition, Morgan Kaufmann / Elsevier, 2014.(Listed topics only from Chapters 4and 6).
3.	John P. Hayes, Computer Architecture and Organization, Third Edition, Tata McGraw Hill, 2012.
4.	John L. Hennessey and David A. Patterson, Computer Architecture – A Quantitative Approach‖, Morgan Kaufmann / Elsevier Publishers, Fifth Edition, 2012.

Continuous Internal Evaluation (CIE):

Theory for 50 Marks

CIE is executed by way of quizzes (Q), tests (T) and assignments. A minimum of three quizzes are conducted along with tests. Test portion is evaluated for 50 marks and quiz is evaluated for 10 marks. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three (conduct additional quizzes and take best three). The three tests are conducted for 50 marks each and the average of all the tests are calculated for 50. The marks for the assignments are 20 (2 assignments for 10 marks each). The marks obtained in test, quiz and assignment are added to get marks out of 100 and report CIE for 50 marks.

Semester End Examination (SEE):

Total marks: 50+50=100

SEE for 50 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the entire syllabus. Part – B Students have to answer five questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have a maximum of three sub divisions. Each unit will have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom's taxonomy level.

CO-PO Mapping												
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	**1**	**2**	**1**	**1**	**1**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
CO2	**2**	**2**	**1**	**1**	**1**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
CO3	**1**	**2**	**2**	**1**	**1**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
CO4	**2**	**2**	**2**	**1**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
CO5	**1**	**2**	**2**	**1**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**

High-3, Medium-2, Low-1

Semester: IV		
PYTHON PROGRAMMING (Theory and Practice)		
Course Code: MVJ21CS/CG/AI44		**CIE Marks:50+50**
Credits: L:T:P: 3:0:1		**SEE Marks: 50 +50**
Hours:40 L+ 26 P		**SEE Duration: 03+03 Hours**
Course Learning Objectives: The students will be able to		
1	Familiarize the students with the fundamentals and programming basics of Python Language	

UNIT-I	
Introduction to Python: Features of python, Applications of python, Syntax, Comments, Indentations, Number types, Variables and Data Types, Operators, conditional statement, Loops in Python.	**8 Hrs**

Python List: Create Python List, Access Python List, Slicing a Python List, slicing and dicing, Reassigning a Python List (Mutable), Reassigning the whole Python list, Deleting list and elements, Multidimensional Lists, List Operations, Built-in List Functions.	
UNIT-II	
Python Tuple: Create a Python Tuple, Tuples Packing, Tuples Unpacking, Creating a tuple with a single item, Access Python Tuple, Slicing a Tuple, Deleting a Python Tuple, Reassigning Tuples, Tuple Functions Tuple Operations. **Python Dictionary:** Create a Dictionary, Dictionaries with mixed keys, Access a Python Dictionary, Delete Python Dictionary, In-Built Functions on a Python Dictionary, In-Built Methods on a Python Dictionary, Dictionary Operations.	**8 Hrs**
UNIT-III	
Python Function: User-Defined Functions in Python, Python Built-in Functions, Python Lambda Expressions, Recursion Function, Range function. **Python Method:** Introduction to Method, __init__(), Self Parameter, Functions vs Method, Magic Methods	**8 Hrs**
UNIT-IV	
Python Class: Introduction to Python Class, Defining a Python Class, Accessing Python Class Members Python Object Attributes Belonging to Python Class, Delete Python Class, Attribute, Inheritance, Multiple inheritance.	**8 Hrs**
UNIT-V	
File Handling In Python: Read and Write File, Open File, Close File, File Methods, Data Base connections.	**8 Hrs**
LABORATORY EXPERIMENTS	

1. Write a Python program to encrypt the text using Caesar Cipher technique. Display the encrypted text. Prompt the user for input and the shift pattern.
2. Devise a Python program to implement the Rock-Paper-Scissor game.
3. Write a Python program to perform Jump Search for a given key and report success or failure. Prompt the user to enter the key and a list of numbers.
4. The celebrity problem is the problem of finding the celebrity among n people. A celebrity is someone who does not know anyone (including themselves) but is known by everyone. Write a Python program to solve the celebrity problem.
5. Write a Python program to construct a linked list. Prompt the user for input. Remove any duplicate numbers from the linked list.
6. Perform the following file operations using Python

   a) Traverse a path and display all the files and subdirectories in each level till the deepest level for a given path. Also, display the total number of files and subdirectories.

   b) Read a file content and copy only the contents at odd lines into a new file.

7. Create a menu drive Python program with a dictionary for words and their meanings. Write functions to add a new entry (word: meaning), search for a particular word and

retrieve meaning, given meaning find words with the same meaning, remove an entry, display all words sorted alphabetically.

8. Using Regular Expressions, develop a Python program to

a) Identify a word with a sequence of one upper case letter followed by lower case letters.

b) Find all the patterns of "1(0+)1" in a given string.

c) Match a word containing 'z' followed by one or more o's.

Prompt the user for input.

9. Devise a Python program to implement the Hangman Game.

10. Write a Python program to print all the Disarium numbers between 1 and 100

Any 10 experiments to be conducted

Course Outcomes: After completing the course, the students will be able to	
CO1	Understand data types (like character strings, integers, and real numbers)and the Operations that can be Applied to each data type.
CO2	Write programs that get input, perform calculations, and provide output (using Conditional logic, loops, Functions).
CO3	Write well designed and well documented programs that are easily maintainable
CO4	Analyze String Formatting Options.
CO5	Enjoy the art and science of computer files using python.

Reference Books	
1.	Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser Data Structures and Algorithms in PythonJohn Wiley & Sons, Incorporated.
2.	Frank Kane (2017)Hands-On Data Science and Python Machine Learning 1st Edition, Kindle Edition
3.	Mark Smart,(2018), Introduction to Data Science with Python: Basics of Numpy and Pandas.
4.	VK Jain,Data Science & Analytics, Khanna Book Publishing ;edition (2018)

Continuous Internal Evaluation (CIE):

Theory for 50 Marks

CIE is executed by way of quizzes (Q), tests (T) and assignments. A minimum of three quizzes are conducted along with tests. Test portion is evaluated for 50 marks and quiz is evaluated for 10 marks. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three (conduct additional quizzes and take best three). The three tests are conducted for 50 marks each and the average of all the tests are calculated for 50. The marks for the assignments are 20 (2 assignments for 10 marks each). The marks obtained in test, quiz and assignment are added to get marks out of 100 and report CIE for 50 marks.

Laboratory- 50 Marks

The laboratory session is held every week as per the time table and the performance of the student is evaluated in every session. The average of the marks over number of weeks is considered for 30 marks. At the end of the semester a test is conducted for 10 marks. The students are encouraged to implement additional innovative experiments in the lab and are awarded 10 marks. Total marks for the laboratory is 50.

Semester End Examination (SEE):

Total marks: 50+50=100

SEE for 50 marks are executed by means of an examination.

The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the entire syllabus. Part – B Students have to answer five questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have a maximum of three sub divisions. Each unit will have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom's taxonomy level.

Semester: IV		
DESIGN AND ANALYSIS OF ALGORITHMS & LAB (Theory and Practice)		
Course Code: MVJ21CS/CG/AI45		**CIE Marks:50+50**
Credits: L:T:P: 3:0:1		**SEE Marks: 50 +50**
Hours:40 L+ 26 P		**SEE Duration: 03+03 Hours**
Course Learning Objectives: The students will be able to		
1	Identify the importance of different asymptotic notation.	
2	Determine the complexity of recursive and non-recursive algorithms.	
3	Compare the efficiency of various design techniques like greedy method, backtracking etc.	
4	Apply appropriate method to solve a given problem.	

UNIT-I	
Basic Concept of Algorithms: Introduction-What is an Algorithm, Algorithm Specification, Analysis Framework, Performance Analysis: Space complexity, Time complexity. Asymptotic Notations: Big-Oh notation (O), Omega notation (Ω), Theta notation (Θ), and Little-oh notation (o), Mathematical analysis of Non-	**8 Hrs**

Recursive and recursive Algorithms with Examples . Important Problem Types. Fundamental Data Structures.	
UNIT-II	
Simple Design Techniques – Brute force : Selection sort, Bubble sort, Sequential Search and Brute-Force String Matching , Exhaustive search –Traveling Salesman problem, Knapsack problem , Assignment Problem. **Divide and Conquer:** General method, Binary search, Recurrence equation for divide and conquer, Finding the maximum and minimum , Merge sort, Quick sort , Strassen's matrix multiplication , Advantages and Disadvantages of divide and conquer.	**8 Hrs**
UNIT-III	
Decrease and Conquer approach: Topological Sort, Decrease-by-a-Constant-Factor Algorithms: Josephus Problem. **Greedy Method**: General method, Coin Change Problem, Knapsack Problem, Job sequencing with deadlines. Minimum cost spanning trees: Prim's Algorithm, Kruskal's Algorithm. Single source shortest paths: Dijkstra's Algorithm. Huffman Trees and Codes.	**8 Hrs**
UNIT-IV	
Dynamic Programming: General method with Examples, Multistage Graphs. Transitive Closure: Warshall's Algorithm, All Pairs Shortest Paths: Floyd's Algorithm, Optimal Binary Search Trees, Knapsack problem, Bellman-Ford Algorithm , Travelling Sales Person problem , Reliability design.	**8 Hrs**
UNIT-V	
Backtracking: General method, N-Queens problem, Sum of subsets problem, Graph coloring, Hamiltonian cycles Programme and Bound: Assignment Problem, Travelling Sales Person problem, 0/1 Knapsack problem. **LC Programme and Bound solution :** FIFO Programme and Bound solution. NP-Complete and NP-Hard problems: Basic concepts, non-deterministic algorithms, P, NP, NP-Complete, and NP-Hard classes.	**8 Hrs**
LABORATORY EXPERIMENTS	
1.Create a Java class called Student with the following details as variables within it. (i) USN (ii) Name (iii) Branch (iv) Phone Write a Java program to create nStudent objects and print the USN, Name, Branch, and Phoneof these objects with suitable headings. 2.Write a Java program to read two integers a andb. Compute a/b and print, when b is not	

zero. Raise an exception when b is equal to zero.
3.Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer for every 1 second; second thread computes the square of the number and prints; third thread will print the value of cube of the number.
4.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 non graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate using Java how the divide-and-conquer method works along with its time complexity analysis: worst case, average case and best case.
5.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 non graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate using Java how the divide-and-conquer method works along with its time complexity analysis: worst case, average case and best case.

6. Implement in Java, the 0/1 Knapsack problem using (a) Dynamic Programming method

(b) Greedy method.

7. From a given vertex in a weighted connected graph, find shortest paths to other vertices

using Dijkstra's algorithm. Write the program in Java.

8. Find Minimum Cost Spanning Tree of a given connected undirected graph using

Kruskal's algorithm. Use Union-Find algorithms in your program.
9.Find Minimum Cost Spanning Tree of a given connected undirected graph using Prim's algorithm.
10. Write Java programs to (a) Implement All-Pairs Shortest Paths problem using Floyd's algorithm. (b) Implement Travelling Sales Person problem using Dynamic programming.
11. Design and implement in Java to find all Hamiltonian Cycles in a connected undirected Graph G of n vertices using backtracking principle.
12. Design and implement in Java to find a subset of a given set S = {Sl, S2,.....,Sn}

of n positive integers whose SUM is equal to a given positive integer d.

For example, if S ={1, 2, 5, 6, 8} and d= 9, there are two solutions {1,2,6}and

{1,8}. Display a suitable message, if the given problem instance doesn't have a solution.

Any 10 experiments to be conducted

Course Outcomes: After completing the course, the students will be able to	
CO1	Describe the need of algorithm and the notations used in design analysis.

CO2	Compare the efficiency of brute force, divide and conquer techniques for problem solving.
CO3	Ability to apply greedy algorithms, hashing and string matching algorithms.
CO4	Ability to design efficient algorithms using various design techniques.
CO5	Ability to apply the knowledge of complexity classes P, NP, and NP Complete and prove certain problems are NP-Complete.

Reference Books	
1.	Introduction to the Design and Analysis of Algorithms, Anany Levitin:, 2rd Edition, 2009. Pearson.
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).
4.	Computer Algorithms/C++, Ellis Horowitz, Satraj Sahni and Rajasekaran, 2nd Edition, 2014, Universities Press.

Continuous Internal Evaluation (CIE):

Theory for 50 Marks

CIE is executed by way of quizzes (Q), tests (T) and assignments. A minimum of three quizzes are conducted along with tests. Test portion is evaluated for 50 marks and quiz is evaluated for 10 marks. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three (conduct additional quizzes and take best three). The three tests are conducted for 50 marks each and the average of all the tests are calculated for 50. The marks for the assignments are 20 (2 assignments for 10 marks each). The marks obtained in test, quiz and assignment are added to get marks out of 100 and report CIE for 50 marks.

Laboratory- 50 Marks

The laboratory session is held every week as per the time table and the performance of the student is evaluated in every session. The average of the marks over number of weeks is considered for 30 marks. At the end of the semester a test is conducted for 10 marks. The students are encouraged to implement additional innovative experiments in the lab and are awarded 10 marks. Total marks for the laboratory is 50.

Semester End Examination (SEE):

Total marks: 50+50=100

SEE for 50 marks are executed by means of an examination.

The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the entire syllabus. Part – B Students have to answer five questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have a maximum of three sub divisions. Each unit will have internal choice in

which both questions cover entire unit having same complexity in terms of COs and Bloom's taxonomy level.

CO-PO Mapping												
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	**3**	**3**	**2**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**1**
CO2	**3**	**3**	**2**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**1**
CO3	**3**	**3**	**3**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**1**
CO4	**3**	**3**	**2**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**1**
CO5	**3**	**3**	**3**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**1**

High-3, Medium-2, Low-1

Semester: IV		
C# AND .NET FRAMEWORK **(Theory)**		
Course Code: MVJ21AEC47		**CIE Marks:100**
Credits: L:T:P:S: 2:0:0:0		**SEE Marks: 100**
Hours: 40L		**SEE Duration: 3 Hrs**
Course Learning Objectives: The students will be able to		
1	Understand the basics of C# and .NET	
2	Learn the variables and constants of C#	
3	Know the object-oriented aspects and applications.	
4	Learn the basic structure of .NET framework.	
5	Learn to create a simple project of .NET Core	
UNIT-I		
Introduction to C# Part-I: Understanding C#, .NET, overview of C#, Variables, Data Types, Operators, Expressions, Branching, Looping, Methods, implicit and explicit casting.		**6 Hrs**
UNIT-II		
Part-II: Constants, Arrays, Array Class, Array List, String, String Builder, Structure, Enumerations, boxing and unboxing.		**6 Hrs**
UNIT-III		

Object Oriented Concepts-I: Class, Objects, Constructors and its types, inheritance, properties, indexers, index overloading, polymorphism	**6 Hrs**
UNIT-IV	
Object Oriented Concepts-II: Sealed class and methods, interface, abstract class,abstract and interface, operator overloading, delegates, events, errors and exception, Threading.	**6 Hrs**
UNIT-V	
Introduction to .NET FRAMEWORK: Assemblies, Versoning, Attributes, reflection, viewing meta data, remoting, security in .NET, Environment Setup of .NET Coreand create a small project.	**6 Hrs**

Course Outcomes: After completing the course, the students will be able to	
CO1	Able to explain how C# fits into the .NET platform
CO2	Describe the utilization of variables and constants of C#
CO3	Use the implementation of object-oriented aspects in applications.
CO4	Analyze and Set up Environment of .NET Core.
CO5	Evaluate and create a simple project application

Reference Books	
1.	Herbert Schildt, "The Complete Reference: C# 4.0", Tata McGraw Hill, 2012
2.	Christian Nagel et al. "Professional C# 2012 with .NET 4.5", Wiley India, 2012.
3.	Andrew Troelsen , "Pro C# 2010 and the .NET 4 Platform, Fifth edition, A Press, 2010.
4.	Ian Griffiths, Matthew Adams, Jesse Liberty, "Programming C# 4.0", Sixth Edition, O"Reilly, 2010.

Continuous Internal Evaluation (CIE):

Theory for 50 Marks

CIE is executed by way of quizzes (Q), tests (T) and assignments. A minimum of three quizzes are conducted along with tests. Test portion is evaluated for 50 marks and quiz is evaluated for 10 marks. Faculty may adopt innovative methods for conducting quizzes

effectively. The number of quizzes may be more than three (conduct additional quizzes and take best three). The three tests are conducted for 50 marks each and the average of all the tests are calculated for 50. The marks for the assignments are 20 (2 assignments for 10 marks each). The marks obtained in test, quiz and assignment are added to get marks out of 100 and report CIE for 50 marks.

Semester End Examination (SEE):

Total marks: 50+50=100

SEE for 50 marksis executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the entire syllabus. Part – B Students have to answer five questions, one from each unit for 16 marks adding up to 80 marks. Each main question mayhave a maximum of three sub divisions. Each unit will have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom's taxonomy level.

CO-PO Mapping												
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	**3**	**1**	**2**	**1**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
CO2	**3**	**2**	**1**	**3**	**3**	**2**	**-**	**-**	**2**	**-**	**1**	**-**
CO3	**3**	**2**	**1**	**3**	**-**	**2**	**-**	**-**	**2**	**-**	**-**	**-**
CO4	**3**	**3**	**2**	**3**	**3**	**2**	**-**	**-**	**2**	**2**	**2**	**-**
CO5	**3**	**2**	**3**	**3**	**3**	**2**	**-**	**-**	**2**	**2**	**2**	**2**

High-3, Medium-2, Low-1

Semester: IV		
Additional Mathematics-II (Common to all branches)		
Course Code:	**MVJ21MATDIP2**	**CIE Marks:50**
Credits:	**L:T:P:S: 4:0:0:0**	**SEE Marks: 50**
Hours:	**40L**	**SEE Duration: 3 Hrs**
Course Learning Objectives: The students will be able to		
1	To familiarize the important concepts of linear algebra.	
2	Aims to provide essential concepts differential calculus, beta and gamma functions.	
3	Introductory concepts of three-dimensional geometry along with methods to solve them.	
4	Linear differential equations	
5	Formation of partial differential equations.	

UNIT-I	
Linear Algebra: Introduction - Rank of matrix by elementary row operations - Echelon form. Consistency of system of linear equations - Gauss elimination method. Eigen values and eigen vectors of a square matrix. Diagonalization of a square matrix of order two. **Self study**: Application of Cayley-Hamilton theorem (without proof) to compute the inverse of a matrix-Examples. **Video Link**: 1. http://nptel.ac.in/courses.php?disciplineID=111	**8 Hrs**
UNIT-II	
Differential calculus: Indeterminate forms: L-Hospital rule (without proof), Total derivatives, and Composite functions. Maxima and minima for a function of two variables. **Beta and Gamma functions:** Beta and Gamma functions, Relation between Beta and Gamma function-simple problems. **Self study**: Curve tracing. **Video Link**: 1. http://nptel.ac.in/courses.php?disciplineID=111	**8Hrs**
UNIT-III	
Analytical solid geometry : Introduction –Directional cosine and Directional ratio of a line, Equation of line in space- differentforms, Angle between two line, shortest distance between two line, plane and equation of plane in different	**8Hrs**

forms and problems. **Self study**: Volume tetrahedron. **Video Link**: 1. http://nptel.ac.in/courses.php?disciplineID=111	
UNIT-IV	
Differential Equations of higher order: Linear differential equations of second and higher order equations with constant coefficients. Inverse Differential operator, Operators methods for finding particular integrals , and Euler –Cauchy equation. **Self study**: Method of variation of parameters **Video Link**: 1. http://nptel.ac.in/courses.php?disciplineID=111	**8 Hrs**
UNIT-V	
Partial differential equation: Introduction- Classification of partial differential equations, formation of partial differential equations. Method of elimination of arbitrary constants and functions. Solutions of non-homogeneous partial differential equations by direct integration. Solution of Lagrange's linear PDE. **Self study:** One dimensional heat and wave equations and solutions by the method of separable of variable **Video Link**: 1. http://nptel.ac.in/courses.php?disciplineID=111	**8 Hrs**

Course Outcomes: After completing the course, the students will be able to	
CO1	Make use of matrix theory for solving system of linear equations and compute eigenvalues and eigen vectors required for matrix diagonalization process.
CO2	Learn the notion of partial differentiation to calculate rates of change of multivariate functions and solve problems related to composite functions and Jacobians.
CO3	Understand the Three-Dimensional geometry basic, Equation of line in space-different forms, Angle between two line and studying the shortest distance .
CO4	Demonstrate various physical models through higher order differential equations and solve such linear ordinary differential equations.
CO5	Construct a variety of partial differential equations and solution by exact methods.

Reference Books	
1.	B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 43rd Edition, 2013, .

2.	G. B. Gururajachar, Calculus and Linear Algebra, Academic Excellent Series Publication 2018-19
3.	Chandrashekar K. S, Engineering Mathematics-I, Sudha Publications, 2010.

Continuous Internal Evaluation (CIE):

Theory for 50 Marks

CIE is executed by way of quizzes (Q), tests (T) and assignments. A minimum of three quizzes are conducted along with tests. Test portion is evaluated for 50 marks and quiz is evaluated for 10 marks. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three (conduct additional quizzes and take best three). The three tests are conducted for 50 marks each and the average of all the tests are calculated for 50. The marks for the assignments are 20 (2 assignments for 10 marks each). The marks obtained in test, quiz and assignment are added to get marks out of 100 and report CIE for 50 marks.

Semester End Examination (SEE):

Total marks: 50+50=100

SEE for 50 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the entire syllabus. Part – B Students have to answer five questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have a maximum of three sub divisions. Each unit will have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom's taxonomy level.

CO-PO Mapping												
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	3	3	0	2	0	0	0	0	0	0	1	1
CO2	3	3	0	2	0	0	0	0	0	0	1	1
CO3	3	3	0	3	0	0	0	0	0	0	0	1
CO4	2	2	0	3	0	0	0	0	0	0	1	1
CO5	2	2	0	2	0	0	0	0	0	0	0	1

High-3, Medium-2, Low-1