KONGU ENGINEERING COLLEGE, PERUNDURAI, ERODE- 638 060

MINUTES OF THE MEETING OF BOARD OF STUDIES IN MCA

Meeting No.	: 23	
Date	: 12-02-2022	
Time	: 10.00 A.M (Online / Offline)	
Google Meet Id	: https://meet.google.com/rtj-rccr-jfx	
Venue	: CTS Laboratory, Department of Computer Applications	

The following members were present for the meeting:

1.	Dr. R. Thamilselvan Professor and Head Department of Computer Applications Kongu Engineering College, Perundurai	Chairman
2.	Dr. A. Sankar Professor, Department of Computer Applications, PSG College of Technology, Coimbatore	University Nominee
3.	Dr.Arun Sahayadhas Professor, Department of Computer Science and Engineering, Vels Institute of Science, Technology and Advanced Studies, Chennai	Academic Council Nominee
4.	Dr. S. Chandrakala Professor, Department of Computer Science and Engineering, Sastra Deemed University, Thanjavur	Academic Council Nominee
5.	Mrs.Anuradha Subramanian Technical Project Lead, Wipro Limited, Bengaluru	Alumni Representative
6.	Mr.Mouleeswaran Subramaniyan Technical Consultant, Wipro Limited, Chennai	Industry Representative
7.	Dr. A. Tamilarasi Professor	Internal Member

8.	Dr. P.A.Selvaraj Associate Professor	Internal Member
9.	Dr. M. Jagadeesan Associate Professor	Internal Member
10. .	Dr. D.Sivabalaselvamani Associate Professor	Internal Member
11.	Ms.S.Hemalatha Assistant Professor (Sr.Gr.)	Internal Member
12.	Dr. M.Pyingkodi Assistant Professor (Sr.Gr.)	Internal Member
13.	Ms. K. Nanthini Assistant Professor (Sr.Gr.)	Internal Member
14.	Dr. T. Kavitha Assistant Professor (Sr.Gr.)	Internal Member
15.	Mr. M. Karthikeyan Assistant Professor (Sr.Gr.)	Internal Member

The following members were present as special invitees:
Nil

The following internal members have requested for leave of absence:
1. Dr. T.M. Saravanan, Associate Professor
2. Dr. L. Rahunathan, Associate Professor
3. Ms. K. Chitra, Assistant Professor (Sr. G)

Meeting of the MCA Board:

Dr. R. Thamilselvan, Chairman/BoS welcomed the members and briefed on curriculum, syllabi of courses to be added or modified under Regulation 2020 of MCA Programme.

The board discussed and approved the following points as per the agenda:

Item No. 23.1: Ratification of the following items under R2020 as given in Annexure-I.

a. Course and Syllabi for PhD Course work
b. One / Two credit courses
c. On line courses
d. Curriculum and Syllabi amendments under R2020 for MCA course

e. Introduction of new electives under R2020 for MCA course
f. Credit transfer from Foreign Universities, Change of Regulations for readmitted students, Transferred candidates
g. Other items if any

Item No. 23.2. Approval of the Syllabi from 3rd semester and 4th semester of MCA under R2020 as given in Annexure-II

Approved in 21st BoS meeting.

Item No. 23.3. Approval of the syllabi of courses to be studied for honours degree under R2020 as given in Annexure-III.

Not Applicable

Item No. 23.4. Approval for Value Added Courses (one / two credit courses), on-line courses with syllabi to be offered from first semester onwards, Transfer of credits from UGC & AICTE approved institutions and Credit transfer from foreign universities under R2020 as given in Annexure-IV.

The members discussed the value added courses (one/ two credit courses), on-line courses with syllabi to be offered from first semester onwards. Transfer of credits from UGC and AICTE approved institutions including NPTEL, SWAYAM, etc., and credit transfer from foreign universities under R2020 (from the year 2021-22 onwards) as given in Annexure – IV and approved the same.

The following two credits value added courses are approved by the members:
1. Mobile App Development with Flutter & Dart
2. Programming with GO
3. MATLAB Programming with Signal Processing
4. Optimization in Machine Learning

The following online NPTEL courses approved by the members:
1. Fundamental Algorithms: Design and Analysis (Noc22-cs01) – 1 credit
2. Foundations of Cryptography (Noc22-cs03) - 3 credits
3. Software Testing (Noc22-cs12) – 1 credit
4. Data Science for Engineers (Noc22-cs28) – 2 credits
5. Programming in Modern C++ (Noc22-cs43) – 3 credits
6. Introduction to Soft Computing (Noc22-cs54) – 2 credits

Item No. 23.5. Approval of Syllabus for PhD courses under R2020 as given in Annexure-V.

Nil.

Reporting Item No. 23.6. Conventional examination system being followed for the November / December 2021 End Semester Examination as given in Annexure-VI.

The members appreciated the conventional examination system being followed for the November / December 2021 end semester examinations as given in Annexure-VI.

The members suggested and discussed the following points in the meeting to incorporate during next revision of the curriculum and syllabi:
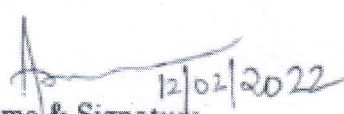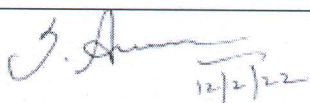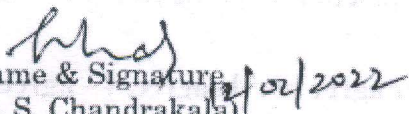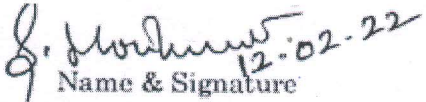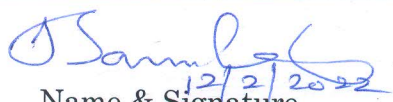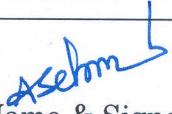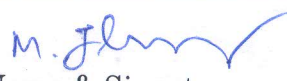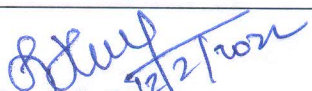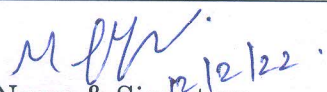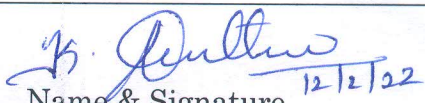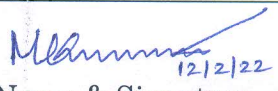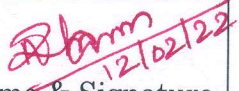
1. C Programming may be excluded from the first semester because most of the students studied this course during UG degree.

2. Python programming shall be moved from second semester to first semester.

3. For the programming and laboratory courses, theory and lab components may be integrated but derive the suitable assessment method to assess programming skill of the students.

4. Emerging courses like Data Science, Artificial Intelligence etc. can be added to fill the gap in the curriculum.

5. Allow the students to earn additional credits using online NPTEL, so that the students encouraged opting online courses.

6. A common methodology may be derived to transfer the grades of online NPTEL courses.

7. Suggested to include the probability in Mathematical Foundation of Computer Applications course. We have already added the probability in Discrete Mathematics Bridge course.

8. Linear algebra and distribution concepts should be covered in mathematics.

9. Professional elective course in fourth semester can be moved to third semester due to the student's project and placement aspects

10. Instead of offering all value added courses as two credit courses, one credit value added courses can also offered

11. Suggested to maintain uniformity in theory and practical hours between regular courses and value added courses.

12. To enhance the employability skills, Node.js has been suggested as value added course

The meeting was concluded with a vote of thanks to the members.

Name & Signature (Dr. A. Sankar)	Name & Signature (Dr.Arun Sahayadhas)
Name & Signature (Dr. S. Chandrakala)	Name & Signature (Ms.Anuradha Subramanian)
Name & Signature (Mr.Mouleeswaran Subramaniyan)	Name & Signature (Dr. A. Tamilarasi)
Name & Signature (Dr. P.A.Selvaraj)	Name & Signature (Dr. M. Jagadeesan)
Name & Signature (Dr. D. Sivabalaselvamani)	Name & Signature (Ms. S.Hemalatha)
Name & Signature (Dr. M. Pyingkodi)	Name & Signature (Ms. K. Nanthini)
Name & Signature (Dr. T.Kavitha)	Name & Signature (Mr. M. Karthikeyan)
	Name & Signature (Dr. R. Thamilselvan)

(i) Curriculum, syllabi of courses to be added newly and syllabi of courses to be modified from 3rd semester to 4th semester for MCA R2020.

Approved in 21st BoS Meeting

Syllabi of the courses to be studied for MCA with Honours in Computer Applications under R2020

NA

List of One / Two credit courses, on-line courses and syllabi, Transfer of credits from UGC and AICTE approved institutions and Credit transfer from foreign universities under R2020 (from the year 2021-22 onwards)

Value added courses

Mobile App Development with Flutter & Dart

L	T	P	Credit
1	0	3	2

Preamble	This course provides the ability to build beautiful cross platform mobile apps for iOS and Android devices using Flutter Framework

Unit - I	Dart Programming Language	10

Dart Programming Language: Installation - Class, function, variable skeleton - Local variable, global variable - Parameters and arguments - Return type, return statement and reusability – Data types (Number, Strings, Boolean, Lists, Maps, Runes, Symbols) - Operators (Arithmetic, Assignment, Relational, Type test, Logical, Conditional) - Enumeration - Constant (final, const)- Conditions (if, if..else, if .. else if… else, switch) - Loops (for, for.. in, while, do... while) - OOPs [classes & object, Constructor, this keyword, static keyword, super keyword, inheritance, super constructor, method overriding, getters & setters, abstract classes, interfaces] - Exception - Typedef - Metadata – Collection- Generics - Packages - Libraries - Generators - Callable classes - Isolates – Async.

Unit - II	Flutter Frameworks	10

Installation [Flutter SDK, Android Studio, Dart] - Hello World application - Flutter widget: Scaffold -Container -Row & Column - Text - TextField - Buttons - Stack - Forms - AlertDialogue - Icons - Images - Card - Tabbar - Drawer - Lists - GridView - Toast - CheckBox - RadioButton - Progress Bar - SnackBar - Tooltip - Slider - Switch - Charts - Navigation Bar - Themes - Table - Calendar - Database – Sqlite (CRUD App Creation)

List of Exercises / Experiments:

1.	To do simple programs using dart language
2.	To do string and arrays programs using dart language
3.	Create a simple program using OOPS concept
4.	Create a simple program using list and set objects
5.	Create a dart programs using Iterating Collections
6.	How to create a simple application in android studio to understand the basics of the Flutter application
7.	To apply the basic Widgets used in Flutter App Development
8.	To implement form elements, validation and submission
9.	To create program using flutter layouts and user management
10	To perform navigation through screens in flutter App
11.	Create simple application using importing external libraries
12.	Working with JSoN data in flutter APP Development
13.	Implement session management using packages for login page
14.	To use cloud firestore to a flutter application and perform different read, write operation and use some queries to retrieve data.
15.	To apply firebase cloud realtime database with Flutter APP

Lecture: 20, Practical: 20, Total: 40

REFERENCES:

1.	Alessandro Biessek, "Flutter for Beginners: An introductory guide to building cross-platform mobile applications with Flutter and Dart 2", Packt Publishing, First Edition, 2019.
2.	Marco L. Napoli, "Beginning Flutter: A Hands On Guide to App Development", Wrox Publisher, First Edition, 2019

COURSE OUTCOMES: On completion of the course, the students will be able to		BT Mapped (Highest Level)
CO1	use various features in Dart Programming Language	Applying (K3)
CO2	implement simple applications using various components in Flutter Framework	Applying (K3), Precision (S3)
CO3	impart database connectivity and cloud databases	Applying (K3), Precision (S3)
CO4	build interactive mobile applications for iOS and Android devices	Applying (K3) Precision (S3)

Mapping of COs with POs and PSOs

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	2	3	2					2				3	2
CO2	2	2	3	2					2				3	2
CO3	3	2	3	2					2				3	3

1 – Slight, 2 – Moderate, 3 – Substantial, BT- Bloom's Taxonomy

ASSESSMENT PATTERN

Test / Bloom's Category*	Remembering (K1) %	Understanding (K2) %	Applying (K3) %	Analyzing (K4) %	Evaluating (K5) %	Creating (K6) %	Total %
CAT – I (Written Test MCQ)		50	50				100
CAT- II (App Development)		30	70				100

* As per the approval

PROGRAMMING WITH GO

L	T	P	Credit
1	0	3	2

Preamble	This course provides the ability to enhance programming skills in GO Framework	
Unit - I	**Program Structure:**	**10**
Names- Declaration-Variables-Assignments-Type Declarations – Basic Data Types: integers – Floating Point Numbers – Complex numbers – Booleans – Strings – Constants – Control flows.		
Unit - II	**Composite Types, Functions and Methods**	**10**
Arrays - Slices – Maps – Structs – Functions: Function Declarations - Recursion – Multiple Return Values - Errors - Function Values – Anonymous Functions – Variadic Functions- Methods.		

List of Exercises / Experiments:

1.	Write a simple Go code using variables and operators.
2.	Write a simple Go code to illustrate the basic data types.
3.	Write a simple Go program using Decision making.
4.	Write a Go program to illustrate String handling.
5.	Write a program in Go for Array composite type.
6.	Implement a Go program using Slice type.
7.	Write a Go program using Map.
8.	Implement a Program in Go using Struct.
9.	Develop Go programs using functions.
10.	Design a Go program with Recursion.
11.	Write a Go program using scope rules.
12.	Design Go programs using methods.
13.	Write a Go program using pointers.
14.	Write a Go program in Type casting.
15.	Design simple Go program using Interface.

Lecture: 20, Practical: 20, Total: 40

REFERENCES:

1.	Alan A. A. Donovan, Brian W. Kernighan,"The Go Programming Language Paperback" 1st Edition, Pearson Education, 2016.
2.	Vladimir Vivien, "Learning Go Programming", 1st Edition, Ingram short title, 2016.
3.	William Kennedy, Brian Ketelsen, Erik St. Martin ,"Go in Action", 1st Edition, Manning Publications , 2015.

COURSE OUTCOMES: On completion of the course, the students will be able to		BT Mapped (Highest Level)
CO1	construct simple logics using Go basics.	Applying (K3), Precision (S3)
CO2	implement simple programs using Control flow and composite types, functions, methods to develop simple applications.	Applying (K3), Precision (S3)
CO3	design applications to implement functions, methods and pointers	Applying (K3), Precision (S3)
CO4	design simple applications using interface.	Applying (K3), Precision (S3)

Mapping of COs with POs and PSOs

COs/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1										3	3
CO2	3	2	1										2	2
CO3	3	2	1										3	2

1 – Slight, 2 – Moderate, 3 – Substantial, BT- Bloom's Taxonomy

ASSESSMENT PATTERN

Test / Bloom's Category*	Remembering (K1) %	Understanding (K2) %	Applying (K3) %	Analyzing (K4) %	Evaluating (K5) %	Creating (K6) %	Total %
CAT – I (MCQ)	10	30	60				100
CAT- II (MCQ)	10	30	60				100

* As per the approval

MATLAB Programming with Signal Processing

L	T	P	Credit
2	0	2	2

Unit - I	MATLAB Programming Techniques	16

Structuring Data-Manipulating Heterogeneous Data-Optimizing Code-Creating Flexible Functions-Creating Robust Applications-Verifying Application behavior-Debugging Code- Organizing Projects

Unit - II	Object Oriented Programming:	16

Creating Custom Data Types-Designing a MATLAB Class-Building Class Hierarchies-Facilitating Multiple References-Writing Unit Tests-Synchronizing Objects

Unit - III	Signal Processing:	9

Signal Processing Onramp-Generating Signals and Common Signal Operations-Estimating Power Spectral Density-Improving the Power Spectral Density Estimate-Characterizing Digital Filters-Designing Digital Filters-Streaming Signal Processing

Total Hours: 41

Optimization in Machine Learning

L	T	P	Credit
2	0	2	2

Unit - I	Math and Optimization:	14

Solving an Unconstrained Optimization Problem in MATLAB-Solving Constrained Optimization Problems in MATLAB-Creating Symbolic Variables-Mathematical Expressions with Symbolic Variables-Creating and Solving Symbolic Equations-Algebraic Manipulation and Simplification-Working with Assumptions-Working with Units of Measurement-Creating Symbolic Functions-Visualizing Symbolic Functions and Equations-Calculus with Symbolic Math-Solving Nonlinear Equations: Rooting Finding – Bisection Method-fzero function-Solving Ordinary Differential Equations-Linear Algebra-Statistical Methods: Exploring Data-Fitting a Curve to Data-Interpolating Data

Unit - II	Optimization Techniques:	8

Synchronizing Objects-Running an Optimization Problem-Specifying Objective Functions and Constraints-Choosing a Solver and Improving Performance-Global and Multi objective Optimization

Unit - III	Machine Learning:	20

Finding Natural Patterns in Data-Classification Methods-Improving Predictive Models-Regression Methods-Neural Networks-Deep Learning: Classifying Images with Convolutional Networks-Interpreting Network Behavior-Creating and Training Networks-Improving Performance-Performing Regression-Using Deep Learning for Computer Vision-Classifying Sequence Data with Recurrent Networks-Classifying Categorical Sequences-Generating Sequences of Output.

Total Hours: 42

Noc22-cs01 - Fundamental Algorithms: Design and Analysis

PRE-REQUISITES: Nil

COURSE PLAN:

Week 1: Insertion Sort, Asymptotic Notation, Merge Sort, Quick Sort, Divide and Conquer

Week 2: Heap Sort, Linear-time Sorting, Order Statistics

Week 3: Hashing, BST Sort, Augmenting Data Structures, Dynamic Programming

Week 4: Graphs: BFS, DFS, Prim's, Shortest Paths

Books and references:

1. Cormen, Thomas, Charles Leiserson, Ronald Rivest, and Clifford Stein. Introduction to Algorithms. 3rd edition.

Noc22-cs03 - Foundations of Cryptography

PRE-REQUISITES: Nil

COURSE PLAN:

Week 1: Course Overview, Symmetric-key Encryption, Historical Ciphers, Perfect Security and Its Limitations

Week 2: Computational Security, Semantic Security and Pseudorandom Generators (PRGs)

Week 3: Stream Ciphers, Provably-secure Instantiation of PRG, Practical Instantiation of PRG, CPA-security and Pseudo-random Functions (PRFs)

Week 4: CPA-Secure Ciphers from PRF, Modes of Operations of Block Ciphers, Theoretical Constructions of Block Ciphers and Practical Constructions of Block Ciphers

Week 5: DES, AES and Message Authentication Codes (MAC)

Week 6: Information-theoretic Secure MAC, Cryptographic Hash Functions, Ideal-Cipher Model, Davies-Meyer construction and Merkle-Damgård Paradigm

Week 7: Birthday Attacks on Cryptographic Hash Functions, Applications of Hash Functions, Random Oracle Model and Authenticated Encryption

Week 8: Generic Constructions of Authenticated Encryption Schemes, Key-exchange Problem, One-way Trapdoor Functions and Cyclic Groups

Week 9: Discrete-Logarithm Problem, Computational Diffie-Hellman Problem, Decisional Diffie-Hellman Problem, Elliptic-Curve Based Cryptography and Public-Key Encryption

Week 10: El Gamal Encryption Scheme, RSA Assumption, RSA Public-key Cryptosystem, KEM-DEM Paradigm and CCA-security in the Public-key Domain

Week 11: CCA-secure Public-key Hybrid Ciphers Based on Diffie-Hellman Problems and RSA-assumption, Digital Signatures, RSA Signatures and Schnorr Identification Scheme

Week 12: Schnorr Signature, Overview of TLS/SSL, Number Theory, Interactive Protocols and Farewell

Books and references:

1. Introduction to Modern Cryptography by Jonathan Katz and Yehuda Lindell
2. Cryptography Theory and Practice by Douglas Stinson

Noc22-cs12 - Software Testing

PRE-REQUISITES: Programming, Algorithms, Discrete Mathematics (basics)

COURSE PLAN:

Week 1: Introduction to software testing and test process

Week 2: Black box testing

Week 3: White box testing

Week 4: Integration, regression and system testing, test automation

Books and references

1. Software Engineering by Rajib Mall, PHI 2014

2. Software Testing: A Craftsman's Approach, by Paul C. Jorgensen, Third Edition

Noc22-cs28 - Data Science for Engineers

- **PRE-REQUISITES:** 10 hrs of pre-course material will be provided, learners need to practice this to be ready to take the course.

COURSE PLAN:

Week 1: Course philosophy and introduction to R

Week 2: Linear algebra for data science

Algebraic view - vectors, matrices, product of matrix & vector, rank, null space, solution of over-determined set of equations and pseudo-inverse)

Geometric view - vectors, distance, projections, eigenvalue decomposition

Week 3: Statistics (descriptive statistics, notion of probability, distributions, mean, variance, covariance, covariance matrix, understanding univariate and multivariate normal distributions, introduction to hypothesis testing, confidence interval for estimates)

Week 4: Optimization

Week 5: 1. Optimization

2. Typology of data science problems and a solution framework

Week 6: 1. Simple linear regression and verifying assumptions used in linear regression

2. Multivariate linear regression, model assessment, assessing importance of different variables, subset selection

Week 7: Classification using logistic regression

Week 8: Classification using KNN and k-means clustering

Books and references

1. Introduction to linear algebra - by Gilbert Strang

2. Applied statistics and probability for engineers – by Douglas Montgomery

Noc22-cs43 - Programming in Modern C++

PRE-REQUISITES: Programming & Data Structure (mandatory), Programming in C(optional). Design and Analysis of Algorithms (optional)

COURSE PLAN:

Week1: Programming in C++ is Fun.

Week2: C++ as Better C.

Week3: OOP in C++.

Week4: OOP in C++.

Week5: Inheritance.

Week6: Polymorphism.

Week7: Type Casting.

Week8: Exceptions and Templates.

Week9: Streams and STL.

Week10: Modern C++.

Week11: Lambda and Concurrency.

Week12: Move, Rvalue and STL Containers.

Books and references

Online Material:

1. C++ reference - C++98 and C++03, C++11, C++14.

2. Overview of the New C++ (C++11/14) by Scott Meyers, 2015.

3. ISO C++ Standards.

4. Presentations used in the Course.

Books:

1. C++ Move Semantics - The Complete Guide by Nicolai M. Josuttis, 2020.

2. C++ Concurrency in Action, 2nd Edition by Anthony Williams, 2019.

3. C++17 - The Complete Guide by Nicolai M. Josuttis, 2020.

4. C++17 In Detail: Learn the Exciting Features of The New C++ Standard by Bartlomiej Filipek, 2019.

5. Professional C++, 4th Edition by Marc Gregoire, 2018.

6. Functional Programming in C++ by Ivan Čukić, 2018.

7. Effective Modern C++: 42 Specific Ways to Improve Your Use of C++11 and C++14 by Scott Meyers, 2015.

Noc22-cs54 - Introduction to Soft Computing

PRE-REQUISITES: Nil

COURSE PLAN:

Week 1: Introduction to Soft Computing, Introduction to Fuzzy logic,Fuzzy membership functions, Operations on Fuzzy sets

Week 2: Fuzzy relations, Fuzzy propositions, Fuzzy implications, Fuzzy inferences

Week 3: Defuzzyfication Techniques-I, Defuzzyfication Techniques-II, Fuzzy logic controller-I, Fuzzy logic controller-II

Week 4: Solving optimization problems, Concept of GA, GA Operators: Encoding,GA Operators: Selection-I

Week 5: GA Operators: Selection-II, GA Operators: Crossover-I, GA Operators: Crossover-II, GA Operators: Mutation

Week 6: Introduction to EC-I, Introduction to EC-II, MOEA Approaches: Non-Pareto, MOEA Approaches: Pareto-I

Week 7: MOEA Approaches: Pareto-II, Introduction to ANN, ANN Architecture

Week 8: ANN Training-I, ANN Training-II, ANN Training-III, Applications of ANN

Books and references

1. An Introduction to Genetic Algorithm Melanic Mitchell (MIT Press)

2. Evolutionary Algorithm for Solving Multi-objective, Optimization Problems (2nd Edition), Collelo, Lament, Veldhnizer(Springer)

3. Fuzzy Logic with Engineering Applications Timothy J. Ross (Wiley)

4. Neural Networks and Learning Machines Simon Haykin (PHI)

Annexure – V

Syllabi for PhD courses under R2020 from the academic year 2021-2022 onwards
Nil

REPORTING ITEM

Online/ conventional method of examination system being followed for the November / December 2021 (both regular and arrear exams) End Semester / Trimester Examinations to be held in February 2022.

a) As per the directions issued by Anna University and guidelines issued by Higher Education Department, Government of Tamilnadu, MCA End Semester Examinations will be conducted through conventional method (paper and pen) in campus.

c) The above examination procedure shall also be followed for the maximum period exhausted students