GOVERNMENT COLLEGE FOR WOMEN, PARADE GROUND, JAMMU
(An Autonomous College)

LEARNING OUTCOMES BASED CURRICULUM FRAMEWORK

FOR

POSTGRADUATE PROGRAMME IN

MASTERS OF COMPUTER APPLICATION

MCA

SEMESTER [I-II]

UNDER

CHOICE BASED CREDIT SYSTEM (CBCS)

EFFECTIVE FROM 2020-21, 2021-22, 2022-23

Rjamwal
11-7-20

Content of the Syllabus

1. Introduction

2. Curriculum Planning learning outcome based approach.

2.1. M.C.A as a curriculum in GCW Parade, Jammu

3. Types of Courses

3.1 Core Course (CC)

3.2 Electives

3.2.1.Discipline Specific Elective (DSE)

3.2.2.Ability Enhancement Courses (AEC)

3.2.2a Ability Enhancement Compulsory Courses (AECC)

3.2.2B Skill Enhancement Courses (SEC)

4. Practical/Tutorial

5. Aims of M.C.A

6. Programme Learning Outcomes for M.C.A

7. About the Programme

8. Passing Criterion

9. Scheme of Course Curriculum is as under:

10. Scheme of Examination / Assessment

11.Teaching-Learning Process

12. Instructions for paper setter for Semester(External) Examination.

13. Syllabus

Ramesh 11-7-20

Preamble

Raising the standards of higher education institutions in country has been the endeavour of UGC since the day it was formed. A number of initiatives have been taken to reform the system of imparting education in the institutions of higher education, by improving and upgrading the academic resources and learning environment, to better the quality of teaching and standards of achievement in terms of learning outcomes across various undergraduate programs under the Faculties of Sciences, Humanities, Commerce and Other Professional streams of higher education including Computer science.

One of the very significant reforms in the undergraduate education has been the introduction of Learning Outcomes-based Curriculum Framework (LOCF), which makes learning student-centric, interactive and outcome-oriented with well-defined aims, objectives and goals to achieve. LOCF aims to ensure uniform standards, of education and content delivery across the country irrespective of the institute and location.

Key components of the planning and development of LOCF are Graduate Attributes (GA), Qualification Descriptors (QD), Program Learning Outcomes (PLO) and Course Learning Outcomes (CLO) which are to be achieved at the successful completion of each undergraduate program.

The main objective of adopting this framework is to prepare a comprehensive course structure and detailed syllabi along with quality reading material in order to have uniform standards of education in undergraduate Computer Science programmes. It is a student centric framework where students get to learn fundamentals of computer science along with the latest trends and techniques like Artificial Intelligence, Internet of Things, Machine Intelligence along with advanced skill sets that include Mobile Application Development, Object Oriented Programming among many other courses.

1. Introduction

Computer Science (CS) has evolved as an important branch of science and engineering. It is a discipline that consists of theory and practice and requires abstract as well as concrete thinking. Nowadays, practically everyone is a computer user, and many people are even computer programmers. When seen from a different perspective, Study & application of Computer Science is science of problem solving. The ever evolving discipline of computer science also has strong connections to other disciplines. Applications of Computer Science, have made improvements, have introduced efficacious methods in the study and practice of other fields like engineering, health care, business, etc. Expertise and Knowledge of computer applications has been proving to be of immense benefit in solving the challenges faced in other fields.

Computer science has a wide range of specialties which are Computer Architecture, Software Systems, Graphics, Artificial Intelligence, Computational Science, and Software Engineering. Drawing from a common core, each specialty focuses on specific challenges. Computer Science is practiced by mathematicians, scientists and engineers. Computer Science traces its roots to Mathematics which is the study of reason and logic and these two concepts play a big role in Computer Science too. Science provides the methodology for learning and refinement. Engineering provides the techniques for building hardware and software.

In India, Computer Science was initially introduced at the Master (postgraduate) level as MCA and M.Tech. Later on, engineering programmes such as B.Tech and B.E in Computer Science & Engineering and in Information Technology were introduced in various engineering College/Institutions to cater to the growing demand of trained engineering manpower in IT

Ramwal
11.7.20

industries. Parallelly, MSc programmes with specialisation in Computer Science were introduced to train manpower. B.Sc and B.Sc(Hons) in Computer Science are also being planned and introduced in different colleges and institutions.

B.Sc with CS and BCA are aimed at undergraduate level training facilitating multiple career paths. Students so graduated, can take up postgraduate programmes in CS leading to research as well as R&D, can be employable at IT industries, or can pursue a teachers' training programme such BEd in Computer Education, or can adopt a business management career. There are several employment opportunities available after thesuccessful completion of an undergraduate programme in CS, graduating students can fetch employment directly in companies as Web Developer, Software Engineer, Network Administrator, Data Scientist, or AI/ML personnel.

The Learning Outcome-based Curriculum Framework in Computer Science is aimed at allowing flexibility and innovation in design and development of course content, in method of imparting training, in teaching learning process and in assessment procedures of the learning outcomes. The emphasis in computer science courses, in outcome-based curriculum framework, help students learn problem solving, accomplishing IT tasks, and expressing creativity, both individually and collaboratively. The proposed framework will help Students learn programming techniques and the syntax of one or more programming languages.

Many of the learning outcomes of Computer Science can be achieved only by programming a computer for several different meaningful purposes. All students must, therefore, have access to a computer with a modern programming language installed. The computer science framework does not prescribe a specific language. The teacher and students will decide which modern programming languages students will learn. More importantly, students will learn to adapt to changes in programming languages and learn new languages as they are developed.

The present Learning Outcome-based Curriculum Framework for bachelor's degrees in CS is intended to facilitate the students to achieve the following.

- To develop an understanding and knowledge of the basic theory of Computer Science and Information Technology with good foundation on theory, systems and applications such as algorithms, data structures, data handling, data communication and computation.
- To develop the ability to use this knowledge to analyse new situations
- To acquire necessary and state-of-the-art skills to take up industry challenges. The objectives and outcomes are carefully designed to suit to the above-mentioned purpose.
- The ability to synthesize the acquired knowledge, understanding and experience for a better and improved comprehension of the real-life problems
- To learn skills and tools like mathematics, statistics, physics and electronics to find the solution, interpret the results and make predictions for the future developments.

2. Curriculum Planning learning outcome based approach.

At Postgraduate level GCW, Parade offers Masters of Computer Application.

2.1. M.C.A. as a curriculum in GCW Parade, Jammu

M.C.A or Masters of Computer Application is a general multidiscipline bachelor programme introduced in Govt. College for Women, Parade in 2018. The programme has a balanced emphasis on three science subjects, one of which is computer science. A student studying B.Sc. with Computer Science is required to choose two other subjects from a pool of subjects which include Physics, Mathematics, Statistics, Electronics, Chemistry. Different institutions offer different choice of combinations of subjects. Most popular combinations are Physics and

Rjamwal 11.7.20

Mathematics, Physics and Electronics, Mathematics and Electronics, but there are also combinations like Statistics and Economics or Commerce and Economics alongwith Computer Science.

3. Types of Courses

3.1 Core Course (CC)

A core course is a mandatory course required in degree. Core course of study refers to a series or selection of courses that all students are required to complete before they can move on to the next level in their education or earn a diploma. The general educational purpose of a core course of study is to ensure that all students take and complete courses that are academically and culturally essential. These are the courses that teach students the foundational knowledge and skills they will need in securing the specific degree or diploma. The core courses are designed with an aim to cover the basics that is expected of a student to imbibe in that particular discipline.

Thus, a course, which should compulsorily be studied by a candidate as a core requirement is termed as a Core course. The present document specifies the core courses for M.C.A. The purpose of fixing core papers is to ensure that all the institutions follow a minimum common curriculum so that each institution/ university adheres to common minimum standard.

3.2 Electives

Generally a course which can be chosen from a pool of courses and which may be very specific or specialized or advanced or supportive to the discipline/ subject of study or which provides an extended scope or which enables an exposure to some other discipline/subject/domain or nurtures the candidate's proficiency/skill is called an Elective Course. Different types of elective courses mandated in the present framework are the following.

- Domain Specific Elective (DSE)
- Generic Elective (GE)
- Ability Enhancement Elective (AEEC)

3.2.1.Discipline Specific Elective (DSE)

Elective courses offered under the main discipline/subject of study is referred to as Discipline Specific Elective. The list provided under this category are suggestive in nature and HEI has freedom to suggest its own papers under this category based on their expertise, specialization, requirements, scope and need. The University/Institute may also offer discipline related elective courses of interdisciplinary nature (to be offered by main discipline/subject of study).

3.2.2.Ability Enhancement Courses (AEC)

The Ability Enhancement Courses may be of two kinds:

3.2.2a **Ability Enhancement Compulsory Courses (AECC):** AECC are the courses based upon the content that leads to knowledge enhancement. These are mandatory for all disciplines. Ability Enhancement Compulsory Courses (AECC) are the following.

- AECC-I English
- AECC-II English/Hindi/ MIL Communications
- AECC-III Environment Science

3.2.2B Skill Enhancement Courses (SEC): SEC courses are value-based and/or skill-based and are aimed at providing hands-on-training, competencies, skills, etc. SEC are at least 2 courses for Honours courses and 4 courses for General bachelor programmes. These courses may be chosen

Rjamwal 11.7.20

from a pool of courses designed to provide value-based and/or skill-based knowledge and should contain both theory and lab/hands-on/training/field work. The main purpose of these courses is to provide students life-skills in hands-on mode to increase their employability. The list provided under this category are suggestive in nature and each university has freedom to suggest their own papers under this category based on their expertise, specialization, requirements, scope and need.

4.Practical/Tutorial

For each core course and DSE course there will be one practical. The list of practical provided is suggestive in nature and each university has the freedom to add/subtract/edit practical from the list depending on their faculty and infrastructure available. Addition will however be of similar nature.

5. Aims of Masters of Computer Application.

The Masters of Science degree in Computer Science emphasizes problem solving in the context of algorithm development and software implementation and prepares students for effectively using modern computer systems in various applications. The curriculum provides required computer science courses such as programming languages, data structures, computer architecture and organization, algorithms, database systems, operating systems, and software engineering; as well as elective courses in artificial intelligence, computer-based communication networks, distributed computing, information security, graphics, human-computer interaction, multimedia, scientific computing, web technology, and other current topics in computer science. The main aim of this Bachelor's degree is to deliver a modern curriculum that will equip graduates with strong theoretical and practical backgrounds to enable them to excel in the workplace and to be lifelong learners. The purpose of the MCA programs are two fold: (1) to prepare the student for a position involving the design, development and implementation of computer software/hardware, and (2) to prepare the student for entry into a program of postgraduate study in computer science/engineering and related fields.

The Masters of Computer Applications focus on the concepts and techniques used in the design and development of software systems. Students in this program explore the conceptual underpinnings of Computer Science -- its fundamental algorithms, programming languages, operating systems, and software engineering techniques. In addition, students choose from a rich set of electives that includes data science, computer graphics, artificial intelligence, database systems, computer architecture, and computer networks, among other topics. A generous allotment of free electives allows students to combine study in computer science with study in auxiliary fields to formulate a program that combines experiences across disciplines.

6. Programme Learning Outcomes for MCA

The Masters of Computer Applications program enables students to attain, by the time of graduation:

- Demonstrate the aptitude of Computer Programming and Computer based problem solving skills.
- Display the knowledge of appropriate theory, practices and tools for the specification, design, implementation
- Ability to learn and acquire knowledge through online courses available at different

Rjanwal 11.7.20

- MOOC Providers.
- Ability to link knowledge of Computer Science with other two chosen auxiliary disciplines of study.
- Display ethical code of conduct in usage of Internet and Cyber systems.
- Ability to pursue higher studies of specialization and to take up technical employment.
- Ability to formulate, to model, to design solutions, procedure and to use software tools to solve real world problems and evaluate .
- Ability to operate, manage, deploy, configure computer network, hardware, software operation of an organization.
- Ability to present result using different presentation tools.
- Ability to appreciate emerging technologies and tools.

7. About the Programme

The Masters of Computer Application is an undergraduate programme of three years duration based on Semester System and consist of six semester. Each semester will be approximately 5 months duration (minimum 90 working days in a semester). A candidate admitted to theprogramme will be required to pass the course within the prescribed academic years from the year of admission to the first semester with an aggregate criteria of 40 percentage in theory as well as Practical subjects.

8. Passing Criterion

The minimum Grade /Grade Point required to pass each paper in a semester examination under CBCS shall be Grade E / Grade Point 5 in each theory paper/ Practical/Project (wherever applicable) in External Examination and Internal Assessment separately i,e.40(Forty) percentage of marks in theory and practical separately.

Ramwa
11-7-20

9. Scheme of Course Curriculum is as under:

M.C.A-Semester-I

	Course No.	Title	Credits	Total Marks
Core Courses	MCACC101	OBJECT ORIENTED PROGRAMMING USING C++	04	100
	MCACC102	DISCRETE MATHEMATICS	04	100
	MCACC103	OPERATING SYSTEM	04	100
	MCAPC150	LAB BASED ON C++ AND OS	08	200
Elective-I (any one)	MCAEC104	OPERATIONS RESEARCH	04	100
	MCAEC105	NUMERICAL COMPUTING		
	MCAEC106	E-COMMERCE		
Foundation Compulsory	MCAFC105	STATISTICAL FOUNDATION FOR COMPUTER SCIENCE	02	50
TOTAL			26	650

M.C.A-Semester-2

	Course No.	Title	Credits	Total Marks
Core Courses	MCACC201	DATA STRUCTURES	04	100
	MCACC202	COMPUTER SYSTEM ARCHITECTURE	04	100
	MCACC203	JAVA PROGRAMMING	04	100
	MCAPC250	LAB BASED ON DS AND JAVA	08	200
Inter-Disciplinary Elective-I	MCAEC204	ELECTIVES OFFERED BY OTHER DEPARTMENTS (COMMUNICATION SKILLS IN ENGLISH)	04	100
Foundation Elective	MCAFC205	R PROGRAMMING	02	50
	MCAFC206	COMPUTER HARDWARE AND TROUBLESHOOTING		
TOTAL			26	650

Rjamwal
11.7.20

10. Scheme of Examination / Assessment

The evaluation of each course shall contain two parts: Internal or In Semester Assessment (IA) and External or End-Semester Assessment (EA). The internal grade awarded to the students in each course in a semester shall be published on the notice board at least one week before the commencement of end semester examination. The responsibility of evaluating the internal assessment is vested on the teacher(s) who teaches the course. There will be external Examinations at the end of each semester for both theory and Practical. 20 of the marks allotted to each theory paper and 50of the marks allotted to each practical paper including field work, wherever prescribed, shall be reserved for internal assessment. The evaluation of a candidate shall be awarded and record thereof maintained in accordance with the Regulations prescribed for the purpose under the CBCS as per the following:

THEORY	Syllabus to be covered in the examination	Time allotted	Weightage
Internal Assessment Test (Pattern: One long answer type question of 06 marks and Five short answer type questions of 03 marks each)	Upto 50 (after 45 days))	1 hour	20
External End Semester University Exam	Upto 100 (after 90 days))	3 hours	80
Total			100

PRACTICAL

Internal Practical Examination+Viva Voce	25 Marks	Total=50 Marks
External Practical Examination+Viva Voce	25 Marks	

Note: In case of failure/re-appear category, the Internal Assessment earned by the candidate as a regular student shall be carried forward to the subsequent examination.

Jamwal 11-7-20

11.Teaching-Learning Process

To meet the set objectives of the course and enable students achieve the expected outcomes of the course from the teaching-learning process any options are appropriately chosen by the teacher.

CLASSROOM TEACHING - Regular classroom and face to face teaching and tutorials can be primarily used for imparting theoretical foundations of Computer Science. Applications of the same may be explained from time to time so that the student can appreciate the theory.

LABORATORY - Lab exercises in programming and usage of package / software tools are an integral part. Open source software/Packages should be preferred over proprietary tools wherever available.

SEMINARS - Guest lectures and seminars involving industry experts and eminent teachers are arranged to help the students understand the practices in the industry and developments in the field.

MOOCS - Teacher choose the appropriate lecture materials and videos on similar courses available online through Massive Open Courses Online in the world wide web (such as NPTEL) to provide good perspective of the course and use cases and promote blended learning.

ASSIGNMENTS - Home assignments should be designed to make student collect information from various sources and solve unfamiliar problems and make comparisons of solutions

SIMULATION - Packages to provide simulated environments to teach various components of networking and hardware working are used wherever feasible.

12. Instructions for paper setter for Semester(External) Examination.

The question paper will be divided into the following three sections. No question will be repeated in the question paper.

Section A

Total of 5 short answer questions (one from each Unit) shall be set and the candidates are required answer all questions. Answer to a question should not exceed 80 words. Each question shall be of marks.

(5 x 3 = 15 marks)

Section B

Total of 5 medium answer questions (one from each Unit) shall be set and the candidates are required to answer all questions. Answer to a question should not exceed 300 words. Each

Rjamwal 11.7.20

question shall be of 7 marks.

(5 x 7 = 35 marks)

Section C

It will contain five long answer questions (one from each Unit). The candidates will be required to answer any two questions. Answer to each question should not exceed 600 words. Each question shall be of 15 marks.

(2 X 15 = 30 marks)

Note:-The paper setter shall ensure that the questions are uniformly distributed over entire syllabus.

Department of Computer Science
MCA First Semester

Course Title: Object Oriented Programming Using C++ **Course No:MCACC101**
Duration of Exam: 3 hrs **Lectures: 4 Hours per Week**
Internal Assessment: 20 **End-Semester Assessment:80** **Total Marks: 100**

Learning Outcomes:

1. Learn the concepts of data, abstraction and encapsulation
2. Be able to write programs using classes and objects, packages.
3. Understand conceptually principles of Inheritance and Polymorphism and their use and program level implementation.
4. Learn exception and basic event handling mechanisms in a program
5. To learn typical object-oriented constructs of specific object oriented programming language

Unit-I

Paradigms of Programming Languages, Procedural programming, Need of OOP, Evolution of OO Methodology and C++, Basic Concepts of OO Approach, Comparison of Object Oriented and Procedure Oriented Approaches, Benefits of OOPs, APPLICATIONSs of OOPs, Objects, classes, encapsulation, abstraction, inheritance, reusability, polymorphism and overloading.

Unit-II

Basic program construction, Data types, reference variables, Input output statements, comments, escape sequence, manipulators, type conversion, arithmetic logical and relational operators, For loop, while loop & do loop and if, if.....else, switch & other control statements, arrays and Strings, new and delete operator.

Unit-III

Functions: passing arguments to functions, returning values from functions, reference arguments, static functions, inline functions, default arguments, variables and storage class and returning by reference. Class and visibility modes, C++ objects, this pointer, object as function argument, function overloading, Operator overloading, Overloading unary and binary operators.

Rjamwal 11-7-20

Unit-IV

Constructors and its types, overloaded constructors, copy constructors, destructor, Memory management, passing and returning Objects from functions, Structures and classes, static class members, Inheritance derived class and base class, derived class constructors, types of inheritance: single level, multiple, multiplevel, hierarchical, hybrid inheritance, function overriding.

Unit-V

Exception handing, file handling, Streams stream classes, stream errors, disk file I/O with streams, file pointers and their manipulations, file handling in text and binary modes.

Practicum:8 credits

Students are required to understand the object-oriented concepts using C++. They are required to practice the concepts learnt in the theory . Some of the programs to be implemented are listed as follows:

1. Number of vowels and number of characters ina string.
2. Write a function called zeros maller () that is passed with two introduce arguments by reference and set the smaller of the number to zero. Write a man() program to access this function.
3. Demonstration of array of object.
4. Using this pointer to return a value (return by reference).
5. Demonstration of virtual function.
6. Demonstration of static function.
7. Accessing a particular record in a student's file.
8. Demonstration of operator overloading.

Suggested Readings:

1. Herbert Schildt, C++ The Complete Reference, McGraw Hill.
2. Robert Lafore, Object Oriented Programming in C++, Galgotia publ.
3. H.M. Deitel and P.J. Deitel, C++: How to Program, Prentice Hall.
4. Bajarne Stroustrup, The C++ Programming Language, (3rd edition), Addision Wesley.
5. Object Oriented Programming and C++, Balaguruswamy, TMH.

Department of Computer Science
MCA First Semester

Course Title: Discrete Mathematics | **Course No: MCACC102**
Duration of Exam: 3 hrs | **Lectures: 4 Hours per Week**
Internal Assessment: 20 | **End-Semester Assessment: 80** | **Total Marks: 100**

Learning Outcomes:

Rjamwal 11.7.20

1. Understand the notion of mathematical thinking, mathematical proofs, and algorithmic thinking, and be able to apply them in problem solving.
2. Understand the basics of combinatorics, and be able to apply the methods from these subjects in problem solving.
3. Be able to use effectively algebraic techniques to analyse basic discrete structures and algorithms.
4. Understand asymptotic notation, its significance, and be able to use it to analyse asymptotic performance for some basic algorithmic examples.
5. Understand some basic properties of graphs and related discrete structures, and be able to relate these to practical examples.

UNIT-I

Fundamentals of Set Theory: Set operations, Algebra of sets, combination of sets, Finite and Infinite sets, Classes of sets, Power Sets, Multi sets, Cartesian Product, Representation of relations, Types of relations, Binary Relations, Equivalence relations and partitions, Partial ordering relations and lattices, Mathematics Induction, Principle of Inclusion & Exclusion, Propositions. Function and its types, Composition of function and relations, Cardinality and inverse relations. Functions & Pigeon hole principles.

UNIT-II

Propositional Calculus: Basic operations: AND (^), OR (v), NOT ( ). Truth-value of a component statement, propositions, tautologies, contradictions. Counting Techniques Rules of Sum of products Permutations with and without repetition, Combination.

UNIT-III

Recursion and Recurrence Relation: Polynomials and their evaluation, Sequences, Introduction to AP, GP and AG series, partial fractions, linear recurrence relation with constant coefficients, Homogeneous solutions, Particular solutions, Total solution of a recurrence relation using generating functions.

UNIT-IV

Introduction to Algebraic Structure Definition, elementary properties of algebraic structures, examples of a Monoid, Submonoid, Semigroup, Groups and rings, Homomorphism, Isomorphism and Automorphism, Subgroups and Normal subgroups, Cyclic groups, Intergral domain and fields, Cosets, Lagrange's theorem, Rings, Division Ring.

UNIT-V

Graphs and Trees: Introduction to graphs, Directed and Undirected graphs, Homomorphic and Isomorphic graphs, Subgraphs, Cut points and Bridges, Multigraph and Weighted graph, Paths and circuits, Shortest path in weighted graphs, Eurelian path and circuits, Hamilton paths and circuits. Planar graphs Luler's formula, Trees, Rooted Trees, Spanning Trees & cut-sets, Binary trees and its traversals.

Suggested Readings:

1. Elements of Discrete Mathematics C.L. Liu, 1985, McGraw Hill.
2. Schaum's Outline series: Theory and problems of Probability by S. Lipshutz, 1982, McGraw Hill Singapore.
3. Concrete Mathematics: A Foundation for Computer Science, Ronald Graham, Donald Knuth and Oren Patashik, 1989, Addision-Wesley.
4. Mathematical Structures for Computer Science, Judith L. Gersting, 1993, computer Science Press
5. Appliled Discrete Structures for Computer Science, Doerr and Levasseur, (Chicago: 1985, SRA)
6. Discrete mathematics by A. Chtewynd and P. Diggle (Modular Mathematics series), 1995, Edward Arnold, London.

Rjamwal 11-7-20

7. Discrete Mathematical Structures, B. Koman and R.C. Busby, 1996, PHI 8 Discrete Mathematical Structures with Applications to Computers by Trembley & Manohar, 1995, McGraw Hill.

Department of Computer Science

MCA First Semester

Course Title: Operating System **Course No:MCACC103**
Duration of Exam: 3 hrs **Lectures : 4 Hours per Week**
Internal Assessment : 20 **End-Semester Assessment:80** **Total Marks: 100**

Learning Outcomes:

1. Describe the important computer system resources and the role of operating system in their management policies and algorithms.
2. To understand various functions, structures and history of operating systems and should be able to specify objectives of modern operating systems and describe how operating systems have evolved over time.
3. Understanding of design issues associated with operating systems.
4. Understand various process management concepts including scheduling, synchronization, and deadlocks.
5. To have a basic knowledge about multithreading.
6. To understand concepts of memory management including virtual memory.
7. To understand issues related to file system interface and implementation, disk management.
8. To understand and identify potential threats to operating systems and the security features design to guard against them.
9. To have sound knowledge of various types of operating systems including Unix and Android.
10. Describe the functions of a contemporary operating system with respect to convenience, efficiency, and the ability to evolve.

UNIT-I

Introduction: Definition, Functions, Types of operating system, Computer System Structure- operation, I/O structure, storage structure, hardware protection, Operating System Services.

Process Management: Process Concept, Process Scheduling, Operation On Processes, Cooperating Processes, Threads, Inter-Process Communication.

Process Synchronization: The Critical Section Problem, Synchronization Hardware, Semaphores Classical Problems of Synchronozation, Critical Regions.

UNIT-II

CPU Scheduling: scheduling criteria, scheduling algorithms: FCFS, SJF, priority scheduling, round robin scheduling, multilevel queue scheduling, multilevel feedback queue scheduling, multiple processor scheduling, real time scheduling.

Rjamwal 11.7.20

Memory Management: Logical & Physical Address Space, Swapping, Continuous Allocation (single partition, multiple partitions), Internal, External fragmentation, Paging, Segmentation, Segmentation with Paging.

UNIT-III

Virtual Memory, Demand Paging, Performance of Demand Paging, Page Replacement. Page Replacement Algorithms- FIFO, optimanl, LRU, LRU approximation algorithms, counting algorithms Trashing, Demand Segmentation.

Deadlocks: Characterization, Methods For Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery From Deadlock.

UNIT-IV

I/O Management: I/O system, I/O strategies, buffering.

File System Interface: File Concept, Access Methods-sequential, direct, index, Directory Structure single-level, two-level, tree-structured, acyclic-graph, general graph.

File System Implementation: File System Structure, allocation, Methods-contiguous allocation, linked allocation, indexed allocation, Free Space management, Directory Management, Directory Implementation, Efficiency and Performance.

UNIT-V

Secondary Storage Structure: Disk Structure, Disk Scheduling, FCFS, SSTF, SCAN, C-SCAN, Look Scheduling, Selection of A Scheduling Algorithm, Disk Management-disk formatting, boot block, bad blocks.

LINUX/UNIX: Features of LINUX operating system, Components of LINUX, Scheduling, Process and memory management, Basic Linux commands, Overview of Shell script programming.

Suggested Readings

1. Silberschatz, Galvin, "Operating System Concepts", Addison Wesley Publishing company, 1989
2. William Stallings, "Operating System", Macmillan Publishing Company.
3. Deitel H.M., "An Introduction To Operating Syste", Addision Wesley Publishing Company, 1984.
4. Tanenbaum, A.S., "Modern Operating System", Prentice Hall of India.
5. Milenkovic M, "Operating system-concepts and design", McGraw Hill, International edition.

Department of Computer Science
MCA First Semester

Course Title: Operating Research **Course No:MCAEC104**
Duration of Exam: 3 hrs **Lectures : 4 Hours per Week**
Internal Assessment : 20 **End-Semester Exam: = 80** **Total Marks: 100**

Learning Outcome:

1. To learn good principles of algorithm design;
2. To learn how to analyse algorithms and estimate their worst-case and average- case behaviour (in easy cases);
3. To become familiar with fundamental data structures and with the manner in
4. which these data structures can best be implemented; become accustomed to the description of algorithms in both functional and procedural styles;
5. To learn how to apply their theoretical knowledge in practice (via the practical

Rjamwal 11.7.20

6. component of the course).

UNIT-I

Overview: Introduction to Operation Reserch, techniques, tools, phases, limitations and applications in OR Linear programming-I: (Graphical method) Introduction, Formulation of a linear programming problem with different types of constraints, requirements, assumption,s merits and demerits, applications of LP, Graphical analysis, Graphical solution, Multiple, unbounded solution and infeasible problems and its applications

Linear programming-II: (Simplex method (SM)) Introduction, SM with several decision variables. Two phase simplex method, M-method, multiple, unbounded solution, infeasible problems, Sensitivitiy and duality analysis in LP, Dual Simplex Problems.

UNIT-II

Transportation Problem (TP): Structure and formulation of TP, Procedure for TP, Methods for finding initial feasible and optimal solution, Unbalanced TP, maximization TP, degeneracy problems in TP.

Assignment Problem (AP): Approach, procedure and maximization, unblalanced assignment problems, Hungarian Method.

Project Scheduling: Network analysis concept, CPM/PERT methods for scheduling of projects.

UNIT-III

Sequencing problems: Processing n-jobs through two, three, M machines, Processing of n-jobs through M machines.

Replacement decisions: Replacement of items that deteriorate with time (with and without change in money value), Staff replacement problem.

UNIT-IV

Integer and dynamic programming: Integer programming, formulation techniques, unimodularity, cutting plane method, branch and bound method.

UNIT-V

Dynamic programming: Methodology and its programming applications.

Game Theory: Basic Semesterinology, solution methods of pure and mixed strategy games, principle of dominance, limitations.

Suggested Readings

1. V.K. Kaproor, Operations Research, Techniques for Management, Edition 7, Publishers: Sultan Chand and sons, 2004.
2. S.S. Rao optimization theory and applications, Viley Eastern Ltd., New Delhi.
3. S.D. Sharma: Operations research, Kedar Nath, Ram Nath & Co.
4. H.A. Taha, Operations Research-An introduction, Macmillan Publishing co.inc.New York.
5. Kanti Swarup, P K Gupta and Man mohan, Operations Research, Sultan Chand and sons, New
6. Prem Kumar Gupta and D.S., Hira, Operations Research-An introduction, S.Chand and Company Ltd, New Delhi.

Department of Computer Science

MCA First Semester

Course Title: Numerical Computing **Course No:MCAEC105**
Duration of Exam: 3 hrs **Lectures : 4 Hours per Week**
Internal Assessment : 20 **End-Semester Exam. = 80** **Total Marks: 100**

Rjamwal 11.7.20

UNIT-I

Floating point Arithmetic: Representation of floating point numbers, Operations, Normalization, Pitfalls of floating point representation, Errors in numerical computation Iterative methods: Zeros of a single transcendental equation and zeroes of polynomial using Bisection Methods, Iteration Method, Regula-Falsi method, Newton Raphson method, Secant method, Rate of convergence of iterative methods.

UNIT-II

Simultaneous Linear Equations: Solutions of system of Linear equations, Gauss Elimination direct method and pivoting, III Conditioned system of equations, Refinement of solution. Gauss Seidal iterative method, Rate of convergence.

UNIT-III

Interpolation and approximation: Finite Difference, Difference tables Polynomial Interpolation: Newton's forward and backward formula Central Difference Formulae: Gauss forward and backward formula, Stirling's, Bessel's, Everett's formula. Interpolation with unequal intervals: language's Interpolation, Newton Divided difference formula, Hermite's Interpolation Approximation of function by Taylor's series and Chebyshev polynomial.

UNIT-IV

Numerical Differentiation and integration: Introduction, Numerical Differentiation, Numerical Integration, Trapezoidal rule, Simpson's rules, Boole's Rule, Weddle's Rule Euler-Maclaurin
Formula solution of differential equations: Picard's Method, Euler's Method, Taylor's Method, Runge-Kutta methods.

UNIT-V

Curve fitting, Cubic Spline and Approximation: Method of least squares, fitting of straight lines, polynomials, exponential curves etc Frequency Chart: Different frequency chart like Histogram, Frequency curve, Pi-chart. Regression analysis: Linear and Non-linear regression, Multiple regression.

Suggested Readings:

1. Rajaraman V., "computer Oriented Numerical methods", PHI
2. Gerald & Wheltley, "Applied Numerical Analysis", AW
3. Jain, Iyengar and Jain, "Numerical Methods for Scientific and Engineering Computations", New Age Int.
4. Grewal B.S., "Numerical methods in Engineering and Science", Khanna Publishers, Delhi.
5. T. Veerarajan, T Ramachandran, "Theory and Problems in Numerical Methods", TMH
6. Pradip Niyogi, "Numerical Analysis and Algorithms", TMH
7. Francis Scheld, "Numerical Analysis", TMH

Department of Computer Science

MCA First Semester

Course Title: E-Commerce **Course No:MCAEC106**
Duration of Exam: 3 hrs **Lectures : 4 Hours per Week**
Internal Assessment : 20 **End-Semester Exam. = 80** **Total Marks: 100**

Learning Outcomes:

1. To understand the basics of electronic commerce.

Rjamwal 11-7-20

2. To understand the EDI
3. To learn about transactions on e-portals and models operating on them.
4. To learn the ethics of management

UNIT-I
Introduction Electronic Commerce with Traditional System, Applications of Electronic Commerce, Advantages & Disadvantages of Electronic Commerce, The Mechanisms of Electronic Commerce.

UNIT-II
EDI: Definition of EDI, Electronic Data Interchange (EDI), EDI would Benefit Business Relationships between organizations, Network Enabled business practices, Applications of EDI, EDI Advantages, EDI Disadvantages, EDI Model, Protocol, Encryption, Data Standards Used in EDI.

UNIT-III
E-Commerce Models: Business to consumer, Business to Business, Consumer to consumer, other models – Brokerage Model, Aggregator Model, Info-mediary Model, Community Model and value chain Model.

UNIT-IV
E-payments Systems: Types of Electronic payment System, Types of E-payment systems, E-cash, E-cheque, credit card, Smart Card, Electronic Purses, types of Receipts, Traditional & modern Payments System, Steps for Electronic Payment, payment Security, Problems with traditional Payment methods, Net Banking, The Shopping Process & Advantages of Pay seal.

UNIT-V
E-Marketing, E-Customer Relationship Management, E-Supply chain Management. Security Issues in E-Commerce: Security risk of E-commerce, Types of threats, Security tools and risk management approach. Cyber laws, Business Ethics, IT Acts.

Suggested Readings:

1. Bharat Bhaskar, Electronic commerece – Framework Technologies and Applications. Tata McGraw Hill.
2. Ravi Kalakota & A.B. Whinston, Frontiers of Electronic commerce, Pearson Education.
3. Ravi Kalakota & A.B. Whinston, Electronic Commerce – A Manager's Guide, Pearson Education.
4. Agarwala kamlesh, N and Agarwala Deeksha, Business on the Net_Introduction to the E-Com. Macmillan India.
5. P. T. Joseph, E-commerce: A Managerial Perspective, PHI, 2002.

Department of Computer Science

MCA First Semester

Course Title: Statistical Methods for Computer Science **Course No:MCAFC105**
Duration of Exam: 2 hrs **Lectures : 2 Hours per Week**
Internal Assessment : 10 **End-Semester Exam. = 40** **Total Marks: 50**

Learning Outcomes:

Rjamwal 11.7.20

1. To make students understand the concept of probability.
2. To familiarize the students with statistics.
3. To understand sampling

UNIT-I

Probability Theory: concept of Probability, Random experiment: Sample spaces; classical probability and frequency, subjective probability, probability of an event, conditional prob ability, mutually exclusive events; Random variable; classification of random variables; mean and variance of discrete random variable; mathematical expections; variance and standard deviation; mode and median, moments of random variable, moment generating functions.

UNIT-II

Discrete Probability Distributions: Binomial (Derivation, mean and variance and fitting of Binomial distribution), Possion (Possion as a limiting case of Binomial distribution, meant and variance and fitting of Possion distribution).

Continuous Probability Distributions: Standard variables and normal distribution, mean and variance of normal distribution, computing normal probabilities, fitting of normal distribution in a given set of data, Student's T test and F-Static test.

UNIT-III

Basic Statistics: Measures of central tendencies:- Mean, Median, Mode; Mesures of dispersion: Range variance and standard deviation; Frequency distribution and cumulative frequency distributions;

Linear correlation coefficient; Linear regression; Non-linear regression; Multiple correlation and multiple-regression;

Concept of Population, Sample, Importance of Sampling and its advantages, Sampling distributions, mean and standard deviation of the sampling distribution of means, Sampling distributionof proportions, mean and standard deviation of sampling distribution of propotions, Sample Variance, Sampling distribution of variances.

Suggested Readings:

1. AFFI, A.A.: Statistical Analysis: A Computer Oriented Approach, Academic Press, Inc. 1979.
2. MORRIS, C., ROLPH, J,: Introduction to Data Analysis and Statistical inference, Prentice-Hall, 1981.
3. SCALZO, F.: Elementary Computer Assisted Statistics, Van Nostrand Reinherd Co. Ltd., 1978.
4. JOHNSTON, J.: Econometric Methods, McGrah-Hill.
5. HOGG, R.V., CRAIG, A.L.: Introduction to Mathematical Statistics, American Publishing Co. Pvt. Ltd.
6. YULE, U.G., KENDALL, M.G. An Introduction to the Theory of Statics, Charles Griffinand Co. Ltd.
7. DRAPER, N.A., SMITH, H.: Applied Regresion Analysis John-Wiley and Sons, Inc.
8. ANDERSON, T.W.: An Introduction to Multivariate Statistical Analysis, John-WileyandSons. Inc.
9. MORRISON. D.F.: Multivariate Statistical Methods, McGraw-Hill.

Department of Computer Science

MCA Second Semester

Course Title: Data Structures **Course No:MCACC201**

Duration of Exam: 3 hrs **Lectures : 4 Hours per Week**

Internal Assessment : 20 **End-Semester Exam. = 80** **Total Marks: 100**

Learning Outcomes:

1. To be familiar with fundamental data structures and with the manner in which these data structures can best be implemented; become accustomed to the description of algorithms in both functional and procedural styles
2. To have a knowledge of complexity of basic operations like insert, delete, search on these data structures.
3. Ability to choose a data structure to suitably model any data used in computer applications.
4. Design programs using various data structures including hash tables, Binary and general search trees, heaps, graphs etc.
5. Ability to assess efficiency tradeoffs among different data structure implementations.
6. Implement and know the applications of algorithms for sorting, pattern matching etc.

UNIT-I

Preliminaries: concept & notation, common operation on data structures, algorithm complexity, time-space trade off between algorithm, physical & logical representation of different data structures. Arrays: Arrays defined, representing arrays in memory, various operation (traversal, insertion, deletion), Multidimensional arrays, Sequential allocation, Address calculation, sparse arrays.
Sorting & Searching: Selection sort, Bubble sort, Merge sort, Radix sort, Quick sort, Swquantial search, linear search and their complexity.

UNIT-II

Linked List: Definition, type (Linear, circular, doubly linked, inverted), representing linked lists in memory, advantages of using linked list over arrays, various operations on Linked list (traversal, insertion, deletion).

UNIT-III

Trees Structures: Tree, Binary Trees, Tree Traversal Algorithms (Pre-Order, In-Order, Post-Order), Threaded Trees, Trees in various sorting & Search Algorithms & their Complexity (Heap Sort, Binary Search Trees), AVL trees, imbalances and rotations.

UNIT-IV

Graphs: Description of graph structure, Implementing graphs in memory, Graphs traversals (Depth First Searching, Breadth first searching).

UNIT-V

File Structures, Concepts of fields & records, Classification of files, File operations, File organizations variable length records and text files. Indexing structures like B, B+ trees, ISAM. Hashing techniques for Direct Files.

Practicum:8 Credits

Given the problem statement, students are required to formulate problem, develop flowchart/algorithm, write code, execute and test it. Students should be given assignments on following :

Ramva 11.7.20

1. Write program that uses functions to perform the following:
2. Creation of list of elements where the size of the list, elements to be inserted and deleted are dynamically given as input.
3. Implement the operations, insertion, deletion at a given position in the list and search for an element in the list
4. To display the elements in forward / reverse order
5. Write a program that demonstrates the application of stack operations (Eg: infix expression to postfix conversion)
6. Write a program to implement queue data structure and basic operations on it (Insertion, deletion, find length) and code atleast one application using queues.
7. Write a program that uses well defined functions to Create a binary tree of elements and Traverse the a Binary tree in preorder, inorder and postorder,
8. Write program that implements linear and binary search methods of searching for an elements in a list
9. Write and trace programs to understand the various phases of sorting elements using the methods
   a.Insertion Sort
   b.Quicksort
   c.Bubble sort
10. Write and trace programs to Create a Binary search tree and insert and delete from the tree.
11. Represent suitably a graph data structure and demonstrate operations of travesrals on it.

Suggested Readings:

1. Symour Lipschutz, "Theory and problems of Data Structure", St. Schaum's Outline series in Computers, Tata McGraw – Hill.
2. Horowitz, E., and Sahni, S., "Fundamentals of data structures", Computer Science Press.
3. Tanhenbaum, A.M., and Augenstein, M.J., "Data Structures with C", Prentice – Hall.
4. "Tremblay & Sorenson, An introduction to Data Structures with Applications:, Tata McGraw-Hill.
5. Aho, A.V., Hopcraft, and Ullman, J.E., "Data structure and Algorithms", Addison Wesley.
6. Thomas Coremen, Introduction to Algorithms, Second edition, Prentice Hall of India(2007)2nd Ed.
7. Mark Allen Weiss, Data Structures & Data Structures & Algorithm analysis in C, Dorling Kingsley (2002) 3rd ed.

Department of Computer Science

MCA Second Semester

Course Title: Computer Organization and Architecture **Course No:MCACC202**
Duration of Exam: 3 hrs **Lectures : 4 Hours per Week**
Internal Assessment : 20 **End-Semester Exam. = 80** **Total Marks: 100**

Learning Outcomes:

Rjamwal 11.7.20

1. To make students understand the basic structure, operation and characteristics of digital computer.
2. To familiarize the students with arithmetic and logic unit as well as the concept of the concept of pipelining.
3. To familiarize the students with hierarchical memory system including cache memories and virtual memory.
4. To make students know the different ways of communicating with I/O devices and
5. standard I/Ointerfaces.

UNIT-I

Computer Architecture Vs. Computer organization, Number Systems and Boolean Algebra, Binary, octal and Hexadecimal number systems and conversion, binary arithmetic, Binary codes, Boolean algebra, Basic operations, Basic Theorems, Boolean Functions-Canonical forms, Simplification of Boolean functions, Karnaugh's maps,

UNIT-II

Processor organization: RISC Vs CISC processors, Register based CPU, Accumulator based CPU, and Stack based CPU, Instruction formats, Types of instructions, Instruction execution, instruction cycle, addressing modes, Control unit design : micro programmed Vs hardwired control unit

UNIT-III

ALU Design: Basic function of arithmetic and logic unit, Design of Accumulator logic, Addition, Subtraction, Multiplication, Division algorithms, floating point representation IEE786, floating point arithmetic operations, Decimal arithmetic operations.

UNIT-IV

Memory Organization: Memory hierarchy, main memory, auxiliary memory, Assiciate memory, Cache memory, Virtual memory.

Input-Output Organization: Input-output interface – asynchronous data transfer, modes of transfer, priority interrupt, DMA, Input output processor.

Suggested Readings:

1. M. Morris Mano, "Computer System Architecture", Prentice-Hall of India, Pvt. Ltd., 3rd Ed.
2. William Stalling "Computer Organization and Architecture", Prentice-Hall of India, Pvt. Ltd., 7th Edition, 2005.
3. Carl Hamacher, Zvonko G. Vranesic and Safwat G. Zaky, "Computer Organization", McGraw-Hill, Fifth Edition, 2001.
4. John P. Hayes, "computer Architecture and Organisation", McGraw Hill, 1998.
5. M. Morris Mano, "Digital Design", Prentice-Hall of India, Pvt. Ltd., Third Edition, 2004.
6. Thomas L. Floyd and R.P, Jain, "Digital Fundamentals", Pearson Education, Tenth edition, 2008.
7. Leach Malvino, "Digital Principles and Applications", Tata McGraw Hill, Fifth edition, 2005.

Department of Computer Science

MCA Second Semester

Course Title: Java Programming **Course No:MCACC203**

Rjamwal 11.7.20

Duration of Exam: 3 hrs **Lectures : 4 Hours per Week**
Internal Assessment : 20 **End-Semester Exam. = 80** **Total Marks: 100**

Learning Outcomes:

1. Knowledge of the structure and model of the Java programming language,
2. Use the Java programming language for various programming technologies
3. Develop software in the Java programming language,
4. Evaluate user requirements for software functionality required to decide whether the Java programming language can meet user requirements

UNIT-I

Introduction: Evolution of OO Methodology, Basic Concepts of OO Approach, Comparison of Object oriented and Procedure Oriented Approaches, Advantages of OOPs, Applications of OOPs. OO concepts: Abstraction, Encapsulation, Inheritance, Polymorphism,
Core Java: Introduction, Operator, Data type, Variable, Arrays, control Statements.

UNIT-II

Methods and Classes, Inheritance, Package and Interface, Exception Handling, Multithread programming, I/O Java Applet, String handling, Networking, Event handling, Introduction to AWT, AWT controls, Layout managers, Menus, Images, Graphics.

Unit-III

Java Swing: Creating a Swing Applet and Application, Programming using Panes, Pluggable Look and feel, Labels, Texts fields, Buttons, Toggle buttons, Checkboxes, Radio Buttons, View Ports, Scroll Panes, Scroll Bars, Lists, combo box, Progress Bar, Menus and Toolbars, Layered Panes, Tabbed Panes, Split Panes, Layout, Windows, Dialog Boxes, Inner frame. JDBC: The connectivity Model, JDBC/ODBC Bridge, java.sql package, connectivity to remote database, navigating through multiple rows retrieved from a database.

UNIT-IV

Java beans: Application Builder tools, The bean developer kit(BDK), JAR files, Introspection, Developing a simple bean, using Bound properties, The Java Beans API, Session Beans, Entity Beans, Introduction to Enterprise Java beans (EJB), Introduction to RMI (Remote Methods Invocation): A simple client-server application using RMI.

UNIT-V

Java Servlets: Servlet basics, Servlet API basic, life cycle of a Servlet, Running Servlet, Debugging Servlets.
Thread-safe Servlets, HTTP Redirects, Cookies, introduction to Java Server pages (JSP)

Practicum:8 credits

1. Students are required to implement object-oriented paradigm using JAVA. Below are the list of some of the experiments.
2. Program on strings: Check the equality of two strings, Reverse a string.

Rjamwal 11.7.20

3. Program using loops: to find the sum of digits of a given number, display a multiplication table, display all prime numbers between 1 to 1000.
4. Program to demonstrate all math class functions.
5. Program on files : to copy a file to another file using Java to package classes.
6. Program to demonstrate method over-riding and overloading
7. Programs on inheritances.
8. Multi-threaded programming.

REFERENCE BOOKS:

1. Margaret Levine Young, "The Complete Reference Internet", TMH.
2. Naughton, Schildt, "The Complete Reference JAVA2", TMH.
3. Balagurusamy E, "Programming in JAVA", TMH.
4. Dustin R. Callway, "Inside Servlets", Addison Wesley
5. Mark Wutica, "Java Enterprise Edition", QUE.
6. Steven Holzner, "Java2 Black book", dreamtech.

Department of Computer Science
MCA Second Semester

Course Title: R-Programming **Course No:MCAFC205**
Duration of Exam: 2 hrs **Lectures : 2 Hours per Week**
Internal Assessment : 10 **End-Semester Exam. = 40** **Total Marks: 50**

Learning Outcomes:

1. To understand the basics of R programming.
2. To implement statistics in R-Programming.
3. To understand the concept of Free software

UNIT-I

Features, how to run R, commets, identifiers, constants, variables, operators, strings, data types: basic data types, vectors, lists, matrices, array, factors, data frames, Connecting R to external interfaces – csv file, Microsoft Excel, MySQL.

UNIT-II

Functions and Packages: function definition, function calling, mathematical function, character functions, statistical functions, packages: installing a package, loading a package.

Charts and Graphs: bar charts histogram, line graph, pie charts, box plots, scatter plots and strip charts.

UNIT-III

Test of Hypotheses: population with known variance, population with unknown variance, population proportion. Non-parametric test: Two samples test, K-samples test. ANOVA- Latin Square Design, Correlation Analysis – Karl Pearson, Spearman, Kendall correlation coefficient. Simple Linear Regression, Multiple Linear Regression, Decision Trees.

Ramwa 11.7.20

Suggested Readings:

1. Beginner's Guide for Data Analysis using R Programming, Jeeva Jose, Khanna Publishers, New Dehi, 2018.

Department of Computer Science

MCA Second Semester

Course Title: Computer Hardware & Troubleshooting **Course No:MCAFC206**
Duration of Exam: 2 hrs **Lectures : 2 Hours per Week**
Internal Assessment : 10 **End-Semester Exam. = 40** **Total Marks: 50**

Learning Outcomes:

1. To understand the concept of hardware inside a CPU.
2. To make the students familiarize with motherboard and its components.
3. To understand troubleshooting.

UNIT-I

Memory chips & Modules, memory types, advanced memory technologies. Troubleshooting memory Power Supply: Power supply function and operation, SMPS, power protection and back up. Backup power system: UPS

UNIT-II

PC Motherboard family tree, Components of motherboard, Input-output ports, I/O bus system, Chipset, Interfaces and I/O Ports, IDE interface: ATA standards, master-slave configuration, SCSI interface, serial port and parallel ports, USB, keyboards, mice, printer and monitors.

UNIT-III

Formatting/Partitioning of Hard Disk, Installation of Operating System, Troubleshooting with PC POST (Power on Self Test), BIOS Errors, Replacement of components etc. Maintenance: Windows file repairing, Use of system tools like Disk defragmentation, Disk clean up, Scan disk etc. Use of open source data recovery tools.

Suggested Readings:

1. Craig Zacker & John Rourtrc: PC Hardware- The complete reference, Tata McGraw Hills.
2. S.K. chauhan: PC Upgrading, Maintenance and troubleshooting guide.
3. Mark Minosi: The complete PC Upgrade and Maintenance Guide publications.

Rjamwal 11.7.20