GOVERNMENT ARTS COLLEGE (AUTONOMOUS)
COIMBATORE-641 018

Learning outcomes-based Curriculum Framework (LOCF) for

M.Sc. COMPUTER SCIENCE

(Effective from the Academic year 2021-2022)

POSTGRADUATE AND RESEARCH DEPARTMENT OF COMPUTER SCIENCE

MAY-2021

Preamble

Education is the key to development of any society. Role of higher education is crucial for securing right kind of employment and also to pursue further studies in best available world class institutes elsewhere within and outside India. Quality education in general and higher education in particular deserves high priority to enable the young and future generation of students to acquire skill, training and knowledge in order to enhance their thinking, creativity, comprehension and application abilities and prepare them to compete, succeed and excel globally. Sustained initiatives are required to reform the present higher education system for improving and upgrading the academic resources and learning environments by raising the quality of teaching and standards of achievements in learning outcomes across all Postgraduate programs in science, humanities, commerce and professional streams of higher education including computer science.

Over the past decades the higher education system of our country has undergone substantial structural and functional changes resulting in both quantitative and qualitative development of the beneficiaries. Such changes have gained momentum with the introduction of Choice Based Credit System (CBCS) which further expects learning outcome-based curriculum in order to maximize the benefits of the newly designed curriculum. The learning outcome-based curriculum will definitely help the teachers of the discipline to visualize the curriculum more specifically in terms of the learning outcomes expected from the students at the end of the instructional process. It is pertinent to mention here that the purpose of education is to develop an integrated personality of the individual and the educational system provides all knowledge and skills to the learner for this.

Tamil Nadu State Council for Higher Education (TANSCHE) has formed the State Integrated Boards of Studies, which, with great diligence and expertise has devised the mandatory areas that have to be covered for three-year under graduation and two-year post-graduation courses to realize the facilitation of the mobility of faculty and students from one university to another and to easily solve the problem of equivalence among courses. Great care has been taken so that these areas would take 75% of the course content and the remaining 25% can be decided by the individual institutions. The areas that must be covered by the student that are mandatory for earning the degree to have due value has been worked out so that the student will gain enough depth of knowledge in the subject concerned. 25% percent of the syllabus should be designed by the institutions, and the areas covered under this also must have a weightage of 25%. This gives the autonomous institution seamless liberty on every Board of Studies (BOS) to innovate and experiment, and more importantly, it is here that the institution devises appropriate strategies by which (i) to make creative and critical applications of what has been learnt in the mandatory components, and (ii) to meaningfully connect the learners to the career demands and expectations. It is essential that the theoretical subject knowledge of the students must be translated into practical hands-on experience.

One of the significant reforms in the Postgraduate education is to introduce the Learning Outcomes-based Curriculum Framework (LOCF) which makes it student-centric, interactive and outcome-oriented with well-defined aims, objectives and goals to achieve. LOCF also aims at ensuring uniform education standard and content delivery across the country which will help the students to ensure similar quality of education irrespective of the institute and location. With initiatives of University Grants Commission (UGC) for nation-wide adoption and implementation of the LOCF for bachelor's programmes in colleges, universities and HEIs in general. A Core Expert Committee (CEC) was constituted to formulate the modalities for developing the LOCF in various subjects being taught in the Postgraduate courses in sciences, humanities, commerce and professional courses. The CEC also constituted the Subject Expert Committees (SEC) in various subjects to prepare detailed guidelines for the LOCF in subjects concerned.

The key components of the planning and development of LOCF are given in terms of clear and unambiguous description of the Graduate Attributes (GA), Qualification Descriptors (QD), Program Learning Outcomes (PLO) and Course Learning Outcomes (CLO) to be achieved at the end of the successful completion of each Postgraduate program to be offered by HEIs. In Postgraduate education in Information Technology, the programme of study leading to the degree of M.Sc. in Information Technology is discussed herewith.

The Qualification Descriptors (QD), Program Learning Outcomes (PLO) and the Course Learning Outcomes (CLO) were also finalized keeping the broad requirement of the programme in view. The LOCF also gives general guidelines for the Teaching Learning Process (TLP) corresponding to each component of theory, experiment, tutorials, projects and industrial / field visits to be followed in order to achieve the stated outcomes for each component. Finally, some suggestions for using various methods in the assessment and evaluation of learning levels of students are also made. It is a student centric framework where they are expected to learn fundamentals of Information Technology along with the latest trends and techniques like Artificial Intelligence, Internet of Things, and Machine Intelligence along with advanced skillsets that include Mobile Application Development, Object Oriented Programming among many other courses.

The Learning Outcomes (LO) specified by the CEC are the guidelines to determine the structure of the Postgraduate programs to be offered by the Higher Educational Institutions (HEI) of our country. The key components of the planning and development of LOCF are given in terms of clear and unambiguous description of the Graduate Attributes (GA), Qualification Descriptors (QD), Program Learning Outcomes (PLO) and Course Learning Outcomes (CLO) to be achieved at the end of the successful completion of each Postgraduate program to be offered by HEIs.

1. Introduction

Computer Science (CS) has been evolving as an important branch of science and engineering throughout the world in last couple of decades and it has carved out a space for itself like any other disciplines of basic science and engineering. Computer science is a discipline that spans theory and practice and it requires thinking both in abstract terms and in concrete terms. Nowadays, practically everyone is a computer user, and many people are even computer programmers. Computer Science can be seen on a higher level, as a science of problem solving and problem solving requires precision, creativity, and careful reasoning. The ever -evolving discipline of computer science also has strong connections to other disciplines. Many problems in science, engineering, health care, business, and other areas can be solved effectively with computers, but finding a solution requires both computer science expertise and knowledge of the particular application domain.

Computer science has a wide range of specialties. These include Computer Architecture, Software Systems, Graphics, Artificial Intelligence, Computational Science, and Software Engineering. Drawing from a common core of computer science knowledge, each specialty area focuses on specific challenges. Computer Science is practised by mathematicians, scientists and engineers. Mathematics, the origins of Computer Science, provides reason and logic. Science provides the methodology for learning and refinement. Engineering provides the techniques for building hardware and software.

Universities and other HEIs introduced programmes of studies in computer science as this discipline evolved itself to a multidisciplinary discipline. Information Technology is growing rapidly. Increasing applications of computers in almost all areas of human endeavour has led to vibrant industries with concurrent rapid change in technology. Unlike other basic disciplines, developing core competency in this discipline that can be reasonably stable becomes a challenge. In India, it 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 for trained engineering manpower in IT industries. Parallelly, BSc and MSc programmes with specialisation in Computer Science were introduced to train manpower in this highly demanding area. M.Sc. in Computer Science is being planned and introduced in different colleges and institutions.

Computer Science education at Postgraduate level will result in earning a Master of Science in CS. M.Sc. in CS is aimed at Postgraduate 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. There are several employment opportunities and after successful completion of a Postgraduate 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 solving problems, 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 Master'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.

1.1 Types of courses and Course structure

Each program may have three types of courses namely Core courses, Elective courses and Self-study/Skill-based courses

1.1.1 Core Courses

The Core courses are those courses whose knowledge is deemed essential for the students registered for a particular Master's degree program. Where feasible and necessary two or more programs may prescribe one or more common core courses.

- The core courses shall be mandatory for all the students registered for the master's degree program.
- The core courses shall be spread all the semesters of the program.

1.2.1 Elective courses

The elective courses can be chosen from a pool of papers. These courses are intended to

- allow the student to specialise in one or more branches of the broad subject area;
- help the student to acquire knowledge and skills in a related area that may have applications in the broad subject area;
- help the student to bridge any gap in the curriculum and enable acquisition of essential skills, for example, statistical, computational, language, communication skills etc.
- help the student to pursue area of interest
- The student may also choose additional elective courses offered by the college to enable him /her to acquire extra credits from the discipline or across the discipline

1.3.1 Project work

A course (core/elective/self-study/skill based) may take the form of a project work.

2. Curriculum Planning- Learning Outcomes-based Approach

2.1 Nature and Extent of the M.Sc. (CS) Programme

The Postgraduate programs in Computer Science build on science-based education at B.Sc level. This prepares a young mind to launch a rigorous investigation of exciting world of computer science.

Framing and implementation of curricula and syllabi is envisaged to provide an understanding of the basic connection between theory and experiment and its importance in understanding the foundation of computing. This is very critical in developing a scientific temperament and to venture a career which a wide spectrum of applications as well as theoretical investigations. The Postgraduate curriculum provides students with theoretical foundations and practical experience in both hardware and software aspects of computers. The curriculum in computer science is integrated with courses in the sciences and the humanities to offer an education that is broad, yet of enough depth and relevance to enhance student employment opportunities upon graduation. As a Master's degree program, the curriculum is based on the criterion that graduates are expected to function successfully in a professional employment environment immediately upon graduation.

The Postgraduate program in Computer Science is presently being offered though the courses designed for granting the following degrees by various colleges and universities in India. All the courses are of 2-year duration spread over four semesters.

i. M.Sc. with Computer Science

M.Sc. or Master of Science with Computer Science is a general multidiscipline Master programme. The programme has a balanced emphasis on three science subjects, one of which is computer science. A student studying M.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 Mathematics, Physics and Electronics, Mathematics and Electronics, but there are also combinations like Statistics and Economics or Commerce and Economics along with Computer Science.

M.Sc.in Computer Science

M.Sc. (CS) in India is generally a two-year degree program which develops advanced theoretical and research skills in subject in which Honours is opted. It is a specialized programme offering specialization in one science subject and another auxiliary science subject. This programme helps in building an advanced professional or academic career. It is an appropriate course for students who wish to pursue a Master of Science (M.Sc) or Doctor of Philosophy (PhD) and a research or academic career. This program facilitates students who wish to pursue an independent research project in an area of interest under the supervision of an academic. Thus M.Sc (CS) has more CS courses than that of MSc programme.

M.Sc. in CS follow CBCS structure as mandated by UGC. In accordance with CBCS guidelines the courses are categorized into compulsory courses, elective courses, ability enhancement courses. These categories of courses are discussed below keeping the present context of Postgraduate education in CS in mind.

2.2 Types of Courses

2.2.1 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.Sc. The courses (papers, as referred popularly) under this category are going to be taught uniformly across all universities with 30% deviation proposed in the draft. 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.

2.2.2 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)

2.2.3 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).

2.2.4 Generic Elective (GE)

An elective course chosen from another discipline/subject, with an intention to seek exposure beyond discipline/s of choice is called a Generic Elective. The purpose of this category of papers is to offer the students the option to explore disciplines of interest beyond the choices they make in Core and Discipline Specific Elective papers. The list provided under this category are suggestive in nature and HEI can design its own papers under this category based on available expertise, specialization, and contextual requirements, scope and need.

2.2.5 Dissertation/Project

An elective course designed to acquire special/advanced knowledge, such as supplement study/support study to a project work, and a candidate studies such a course on his/her own with an advisory support by a teacher/faculty member is called dissertation/project.

2.2.6 Ability Enhancement Courses (AEC)

The Ability Enhancement Courses may be of two kinds:

A. 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

B. 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 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.

2.2.7 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.

2.3 Aims of Master of Science Programmes in Computer Science

The Master 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 BS programs in computer science are twofold: (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 Master of Science program with Computer Science as one subject (MSc with CS) 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.

3. Graduate Attributes

Graduate Attributes (GA) are the qualities, skills and understandings that students should develop during their time with the HEI. These are qualities that also prepare graduates as agents of social good in future. Graduate Attributes can be viewed as qualities in following subcategories.

- Knowledge of the discipline
- Creativity
- Intellectual Rigour
- Problem Solving and Design
- Ethical Practices
- Lifelong Learning
- Communication and Social Skills

Among these attributes, categories attributes under Knowledge of the Discipline are specific to a programme of study.

3.1.a. Knowledge of Discipline of CS

Knowledge of a discipline is defined as "command of a discipline to enable a smooth transition and contribution to professional and community settings. This Graduate Attribute describes the capability of demonstrating comprehensive and considered knowledge of a discipline. It enables students to evaluate and utilise information and apply their disciplinary knowledge and their professional skills in the workplace.

3.1.b. Creativity

Creativity is a skill that underpins most activities, although this may be less obvious in some disciplines. Students are required to apply imaginative and reflective thinking to their studies. Students are encouraged to look at the design or issue through differing and novel perspectives. Creativity allows the possibility of a powerful shift in outlook and enables students to be open to thinking about different concepts and ideas.

3.1.c. Intellectual Rigour

Intellectual Rigour is the commitment to excellence in all scholarly and intellectual activities, including critical judgement. The students are expected in having clarity in thinking. This capability involves engaging constructively and methodically when exploring ideas, theories and philosophies. It also relates to the ability to analyse and construct knowledge with depth, insight and intellectual maturity.

3.1.d. Problem Solving and Design

Problem solving skills empower students not only within the context of their programmes, but also in their personal and professional lives. Many employers cite good problem solving skills as a desired attribute that they would like graduates to bring to the workplace. With an ability to seek out and identify problems, effective problem solvers are able to actively engage with a situation, think creatively, to consider different perspectives to address identified challenge, to try out possible solutions and subsequently evaluate results as a way to make decisions. Through this process they can consolidate new and emergent knowledge and develop a deeper understanding of their subject discipline.

3.1.e. Ethical Practices

Ethical practice is a key component of professionalism and needs to be instilled in curricula across courses. When operating ethically, graduates are aware that we live in a diverse society with many competing points of view. Ethical behaviour involves tolerance and responsibility. It includes being open-minded about cultural diversity, linguistic difference, and the complex nature of our world. It also means behaving appropriately towards colleagues and the community and being sensitive to local and global social justice issues.

3.1.f. Life-Long Learning

The skill of being a lifelong learner means a graduate is open, curious, willing to investigate, and consider new knowledge and ways of thinking. This flexibility of mind means they are always amenable to new ideas and actively seek out new ways of learning or understanding the world.

3.1.g. Communication and Social Skills

The ability to communicate clearly and to work well in a team setting is critical to sustained and successful employment. Good communication and social skills involve the ability to listen to, as well as clearly express, information back to others in a variety of ways - oral, written, and visual - using a range of technologies.

3.1.h. Self-Management

Graduates must have capabilities for self-organisation, self-review, personal development and life-long learning.

3.2 LIST OF GRADUATE ATTRIBUTES for M.Sc. (CS)

Afore-mentioned GAs can be summarized in the following manner.

GA 1. A commitment to excellence in all scholarly and intellectual activities, including critical judgement

GA 2. Ability to think carefully, deeply and with rigour when faced with new knowledge and arguments.

GA 3. Ability to engage constructively and methodically when exploring ideas, theories and philosophies

GA 4. Ability to consider other points of view and make a thoughtful argument

GA 5. Ability to develop creative and effective responses to intellectual, professional and social challenges

GA 6. Ability to apply imaginative and reflective thinking to their studies

GA 7. Commitment to sustainability and high ethical standards in social and professional practices.

GA 8. To be open-minded about cultural diversity, linguistic difference, and the complex nature of our world

GA 9. Ability to be responsive to change, to be inquiring and reflective in practice, through information literacy and autonomous, self-managed learning.

GA 10. Ability to communicate and collaborate with individuals, and within teams, in professional and community settings.

GA 11. Ability to communicate effectively, comprehending and writing effective reports and design documentation, Summarizing information, making effective oral presentations and giving and receiving clear oral instructions.

GA 12. Ability to demonstrates competence in the practical art of computing in by showing in design an Understanding of the practical methods and using modern design tools competently for complex real-life IT Problems.

GA 13. Ability to use a range of programming languages and tools to develop computer programs and systems that are effective solutions to problems.

GA 14. Ability to understand, designs, and analyse precise specifications of algorithms, procedures, and interaction Behaviour.

GA 15. Ability to apply mathematics, logic, and statistics to the design, development, and analysis of software Systems.

GA 16. Ability to be equipped with a range of fundamental principles of Computer Science that will provide the basis for future learning and enable them to adapt to the constant rapid development of the field.

GA 17. Ability of working in teams to build software systems.

GA 18. Ability to identify and to apply relevant problem-solving methodologies.

GA 19. Ability to design components, systems and/or processes to meet required specifications.

GA 20. Ability to synthesise alternative/innovative solutions, concepts and procedures.

GA 21. Ability to apply decision making methodologies to evaluate solutions for efficiency, effectiveness and sustainability.

GA 22. A capacity for self-reflection and a willingness to engage in self-appraisal.

GA 23. Open to objective and constructive feedback from supervisors and peers.

GA 24. Able to negotiate difficult social situations, defuse conflict and engage positively in purposeful debate.

4. Qualification Descriptors

Qualification descriptors are generic statements of the outcomes of study. Qualification descriptors are in two parts. The first part is a statement of outcomes, achievement of which a student should be able to demonstrate for the award of the qualification. This part will be of interest to those designing, approving and reviewing academic programmes. They will need to be satisfied that, for any programme, the curriculum and assessments provide all students with the opportunity to achieve, and to demonstrate achievement of, the outcomes. The second part is a statement of the wider abilities that the typical student could be expected to have developed. It will be of assistance to employers and others with an interest in the general capabilities of holders of the qualification. The framework has the flexibility to accommodate diversity and innovation, and to accommodate new qualifications as the need for them arises. It should be regarded as a framework, not as a straitjacket.

4.1. Qualification Descriptor for M.Sc. with CS

On completion of M.Sc. with Computer Science, the expected learning outcomes that a student should be able to demonstrate are the following.

QD-1. Fundamental understanding of the principles of Computer Science and its connections with other disciplines

QD-2. Procedural knowledge that creates different types of professionals related to Computer Science, including research and development, teaching and industry, government and public service;

QD-3. Skills and tools in areas related to computer science and current developments in the academic field of study.

QD-4. Use knowledge, understanding and skills required for identifying problems and issues, collection of relevant quantitative and/or qualitative data drawing on a wide range of sources, and their application, analysis and evaluation using methodologies as appropriate to Computer Science for formulating solutions.

QD-5. Communicate the results of studies undertaken in Computer Science accurately in a range of different contexts using the main concepts, constructs and techniques.

QD-6. Meet one’s own learning needs, drawing on a range of current research and development work and professional materials.

QD-7. Apply Computer Science knowledge and transferable skills to new/unfamiliar contexts,

QD-8. Demonstrate subject-related and transferable skills that are relevant to industry and employment opportunities.

5. Programme Learning Outcomes

These outcomes describe what students are expected to know and be able to do by the time of graduation. They relate to the skills, knowledge, and behaviours that students acquire in their graduation through the program.

5.1. Programme Learning Outcomes for MSc with CS

The Master of Science with Computer Science (MSc with CS) program enables students to attain, by the time of graduation:

PLO-A. Demonstrate the aptitude of Computer Programming and Computer based problem solving skills.

PLO-B. Display the knowledge of appropriate theory, practices and tools for the specification, design, and implementation

PLO-C. Ability to learn and acquire knowledge through online courses available at different
MOOC Providers.

PLO-D. Ability to link knowledge of Computer Science with other two chosen auxiliary disciplines of study.

PLO-E. Display ethical code of conduct in usage of Internet and Cyber systems.

PLO-F. Ability to pursue higher studies of specialization and to take up technical employment.

PLO-G. Ability to formulate, to model, to design solutions, procedure and to use software tools to solve real world problems and evaluate.

PLO-H. Ability to operate, manage, deploy, configure computer network, hardware, software operation of an organization.

PLO-I. Ability to present result using different presentation tools.

PLO-J. Ability to appreciate emerging technologies and tools.

Programme Learning Outcomes for M.Sc. Computer Science

These outcomes describe what students are expected to know and be able to do by the time of graduation. They relate to the skills, knowledge, and behaviors that students acquire in their graduation through the program.

The Master of Science with Computer Science (M.Sc. with CS) program enables students to attain, by the time of graduation:

PLOs1: Fundamental knowledge in problem solving, general computing, and in depth knowledge in Computer Science (**Disciplinary Knowledge**).

PLOs 2: An ability to identify, analyse, design, optimize and implement system solutions using suitable computing techniques leading to propulsion towards employability (**Research-related skills).**

PLOs 3: An ability to understand and provide solutions to real life problems in Internet of Things with thrust in lifelong learning **(Problem Solving).**

PLOs 4: Fundamental knowledge in computational methods and tools for solving realtime problems **(Communication Skills).**

PLOs 5: An ability to act as a leader, or as a part of a team to create multifunctional software products **(Leadership readiness/qualities).**

PLOs 6: An Ability to demonstrate individual practical experiences in a variety of programming languages and situations **(Analytical reasoning).**

PLOs 7: An understanding of professional, ethical, legal, security and social issues and responsibilities **(Moral and ethical awareness/reasoning).**

PLOs 8: An ability to use current techniques, skills, and tools necessary for computing practice **(Scientific reasoning).**

PLOs 9: An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modelling and design of computer-based systems in a way that demonstrates comprehension of the trade-offs involved in design choices **(Critical thinking**).

6. Course Structure of M.Sc. Computer Science

PG - SCHEME OF EXAMINATIONS: CBCS PATTERN

(For the Students admitted during the Academic Year 2021-2022 and onwards)

Sub Code	Part	Title of the Paper	Hrs (wk	Internal (CA)	Externa l	Total Mark	Ext– Min.	Total Pass Mark	Credits
Semester – I									
21MCS11C	**A**	**Core I:** Computer System Architecture	**5**	**50**	**50**	**100**	**25**	**50**	**4**
21MCS12C	**A**	**Core II:** Data Structures and Algorithms	**5**	**50**	**50**	**100**	**25**	**50**	**4**
21MCS13C	**A**	**Core III:** Advanced Java Programming	**5**	**50**	**50**	**100**	**25**	**50**	**4**
21MCS14C	**A**	**Core IV:** Data Communication and Computer Networks	**5**	**50**	**50**	**100**	**25**	**50**	**4**
21MCS15C	**A**	**Core V:** Database Management Systems	**5**	**50**	**50**	**100**	**25**	**50**	**5**
21MCS16P	**A**	**Core Practical 1 :**Advanced Java Programming Lab	**3**	**50**	**50**	**100**	**25**	**50**	**2**
21MCS17P	**A**	**Core Practical 2 :**DBMS Lab	**2**	**50**	**50**	**100**	**25**	**50**	**2**
			30			**700**			**25**
Semester – II									
21MCS21C	**A**	**Core VI:** Data mining with R Programming	**5**	**50**	**50**	**100**	**25**	**50**	**4**
21MCS22C	**A**	**Core VII:** Operating Systems	**5**	**50**	**50**	**100**	**25**	**50**	**4**
21MCS23C	**A**	**Core VIII:** Data Science using Python	**5**	**50**	**50**	**100**	**25**	**50**	**4**
21MCS24C	**A**	**Core IX:** Software Engineering Concepts	**5**	**50**	**50**	**100**	**25**	**50**	**4**
21MCS25E	**B**	**Elective – I:**	**5**	**50**	**50**	**100**	**25**	**50**	**5**
21MCS26P	**A**	**Core Practical 3 :** Data mining With R Programming Lab	**3**	**50**	**50**	**100**	**25**	**50**	**2**
21MCS27P	**A**	**Core Practical 4 :** Python Programming Lab	**2**	**50**	**50**	**100**	**25**	**50**	**2**
			30			**700**			**25**

Sub Code	Paper	Title of the Paper	Hrs (wk)	Internal (CA) Marks	External Marks	Total Marks	Ext– Min.	Total Pass Mark	Credits
Semester – III									
21MCS31C	**A**	**Core X :**Digital Image Processing	**5**	**50**	**50**	**100**	**25**	**50**	**4**
21MCS32C	**A**	**Core XI:** Cloud Computing	**5**	**50**	**50**	**100**	**25**	**50**	**4**
21MCS33C	**A**	**Core XII:** Web Programming Essentials	**5**	**50**	**50**	**100**	**25**	**50**	**4**
21MCS34C	**A**	**Core XIII:** Mobile Applications Development	**5**	**50**	**50**	**100**	**25**	**50**	**4**
21MCS35E	**B**	**Elective – II:**	**5**	**50**	**50**	**100**	**25**	**50**	**5**
21MCS36P	**A**	**Core Practical 5:**Digital Image Processing Lab using Python	**3**	**50**	**50**	**100**	**25**	**50**	**2**
21MCS37P	**A**	**Core Practical 6:** Mobile Applications Development Lab	**2**	**50**	**50**	**100**	**25**	**50**	**2**
			30			**700**			**25**
Semester – IV									
21MCS41C	**A**	**Core XIV:** Open Source Technology (PHP/MySQL)	**5**	**50**	**50**	**100**	**25**	**50**	**3**
21MCS42P	**A**	**Core Practical 7:** Open Source Technology (PHP/MySQL) Lab	**5**	**50**	**50**	**100**	**25**	**50**	**2**
21MCS43V	**A**	**Project Viva Voce**	**20**	**100**	**100**	**200**	**50**	**100**	**10**
			30			**400**			**15**
		Total /Credits				**2500**			**90**

Core -Includes core theory, practical and electives Includes 50 continuous Internal Assessment Marks for Theory and Practical papers respectively Project evaluation done by both Internal and External examiner for 100 Marks.

ELECTIVES FOR SECOND SEMESTER

ELECTIVE – I

Discrete Structures and optimization

Machine Learning Techniques

Artificial Intelligence

Theory of computation and Compilers

Programming Languages and Computer Graphics

ELECTIVES FOR THIRD SEMESTER

ELECTIVE – II

Computer Optimization Techniques

Reactive Programming

Enterprise Resource Planning

Embedded Systems

IOT Architecture and Protocol

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	**Computer System Architecture**	I	21MCS11C

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Describe the basic concepts of data representations, register transfer logic and arithmetic operations.
2. Develop the Instruction execution stages and different types of addressing modes and Memory Organization.
3. Explain the architecture and functionality of central processing unit.
4. Discuss different types of peripheral devices of computer.
5. Distinguish types of serial communication techniques.
6. Infer the concept of memory hierarchy.
7. Discuss the concept of memory organization.
8. Explain the use of cache memory and virtual memory.
9. Justify the concept of memory management hardware.

UNIT I:

Data Representation: Data Types, Number Systems and Conversion, Complements, Fixed Point Representation, Floating Point Representation, Error Detection Codes, Computer Arithmetic - Addition, Subtraction, Multiplication and Division Algorithms.

Register Transfer and Microoperations: Register Transfer Language, Bus and Memory Transfers, Arithmetic, Logic and Shift Microoperations.

UNIT II:

Basic Computer Organization and Design: Stored Program Organization and Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory-Reference Instructions, Input-Output, Interrupt.

Microprogrammed Control: Control Memory, Address Sequencing, Design of Control Unit.

UNIT III:

Central Processing Unit: General Register Organization, Stack Organization, Instruction Formats, Addressing Modes, RISC Computer, CISC Computer.

UNIT IV:

Input-Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous Data Transfer, Modes of Transfer, Priority Interrupt, DMA, Serial Communication.

UNIT V:

Memory Hierarchy: Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory, Memory Management Hardware.

Multiprocessors: Characteristics of Multiprocessors, Interconnection Structures.

TEXT BOOK

1. M.Morris mano, **“Computer System Architecture**”, 3ed, Pearson Education India, 2007.

REFERENCES

1. Hayes J.P, “**Computer Architecture and Organization**”, 3^{rd}Edition,TataMcgraw Hill,2009.

FURTHER READINGS

1. John L Hennessy & David A. Patterson, **“Computer Architecture: A Quantitative Approach**”, 5Th Edn Elsevier India, 2014.

2. **https://nptel.ac.in/courses/106103068**

Semester I		Course Code 21MCS11C		Title of the Paper COMPUTER SYSTEM ARCHITECTURE					Hours 5	Credits 4
Course Outcome(CO)		CO1	CO2	CO3	CO4	CO5	CO6	CO7	CO8	CO9
Programme Learning Outcome (PLO)	DISPLINARY KNOWLEDGE	✓	✓		✓		✓		✓	
	RESEARCH-RELATED SKILLS		✓			✓				✓
	PROBLEM SOLVING			✓			✓			
	COMMUNICATION SKILLS	✓				✓				✓
	LEADERSHIP READINESS/ QUALITIES			✓			✓			✓
	ANALYTICAL REASONING				✓			✓		✓
	MORAL AND ETHICAL AWARENESS		✓				✓			✓
	SCIENTIFIC REASONING		✓		✓		✓			✓
	CRITICAL THINKING		✓			✓		✓		✓

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	DATA STRUCTURES AND ALGORITHMS	I	21MCS12C

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Outline basic and advanced data structures.
2. Discuss Graphs and algorithm design and analysis and to increase the knowledge of usage of data structures in algorithmic perspective.
3. Point out searching and sorting techniques.
4. Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it.
5. Describe the dynamic-programming paradigm and Synthesize dynamic-programming algorithms, and analyse them.
6. Understand P, NP, polynomial reduction, NP-hardness, and NP-Completeness.

UNIT I: LINEAR DATA STRUCTURES

Introduction - Abstract Data Types (ADT) – Stack – Queue – Circular Queue - Double Ended Queue - Applications of stack – Evaluating Arithmetic Expressions - Other Applications - Applications of Queue - Linked Lists - Singly Linked List - Circularly Linked List - Doubly Linked lists – Applications of linked list – Polynomial Manipulation.

UNIT II: NON-LINEAR TREE STRUCTURES

Binary Tree – expression trees – Binary tree traversals – applications of trees – Huffman Algorithm - Binary search tree - Balanced Trees - AVL Tree - B-Tree – Heap- Heap operations- Hash set.

UNIT III: GRAPHS

Representation of graph - Graph Traversals - Depth-first and breadth-first traversal - Applications of graphs - Topological sort – shortest-path algorithms - Dijkstra‟s algorithm – Bellman-Ford algorithm – Floyd's Algorithm - minimum spanning tree – Prim's and Kruskal's algorithms.

UNIT IV: ALGORITHM DESIGN AND ANALYSIS

Algorithm Analysis – Asymptotic Notations - Divide and Conquer – Merge Sort – Quick Sort - Binary Search - Greedy Algorithms – Knapsack Problem – Dynamic Programming – Optimal Binary Search Tree - Warshall‟s Algorithm for Finding Transitive Closure.

UNIT V: ADVANCED ALGORITHM DESIGN AND ANALYSIS

Backtracking – N-Queen's Problem - Branch and Bound – Assignment Problem - P & NP problems – NP-complete problems – Approximation algorithms for NP-hard problems – Traveling salesman problem.

TEXT BOOKS

1. Anany Levitin,"**Introduction to the Design and Analysis of Algorithms**",2nd Editon, Pearson Education,2015.
2. E. Horowitz, S.Sahni and Dinesh Mehta, "**Fundamentals of Data structures in C++",**2nd Edition,

University Press,2007.

3. E. Horowitz, S. Sahni and S. Rajasekaran, "**Computer Algorithms/C++**", Second Edition, University Press,2007.
4. M. A. Weiss, "**Data Structures and Algorithm Analysis in Java**", 3rd Edition, Pearson Education Asia, 2013.
5. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "**Introduction to algorithms**", 3rd Edition, PHI Learning Private Ltd,2012.
6. Tenenbaum,A.S., Langram Y. Augestein M.J, "**Data Structures using C**"2nd Edition, Pearson Education,2004.
7. V. Aho, J. E. Hopcroft, and J. D. Ullman, "**Data Structures and Algorithms**",1st Edition, Pearson Education , 2002.

REFERENCES

1. George T.Heineman**, " Algorithms in a Nutshell",** 2nd Edition, O'Reilly Publication,2008.

FURTHER READINGS

1. Michael T. Goodrich and Roberto Tamassia, "**Data Structures and Algorithms in Java**", Wiley Publication, 1997.
2. **https://nptel.ac.in/courses/106106127**

Semester **I**		**Course Code** **21MCS12C**	**Title of the Paper** **DATA STRUCTURES AND ALGORITHMS**			**Hours** **5**	**Credits** **4**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**	✓					✓
	RESEARCH-RELATED SKILLS			✓			✓
	PROBLEM SOLVING					✓	
	COMMUNICATION SKILLS	✓		✓			✓
	LEADERSHIP READINESS/QUALITIES				✓		
	ANALYTICAL REASONING		✓		✓		
	MORAL AND ETHICAL AWARENESS	✓		✓			✓
	SCIENTIFIC REASONING	✓			✓		
	CRITICAL THINKING			✓			✓

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	ADVANCED JAVA PROGRAMMING	I	18MCS13C

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Explain Java Programming language and to design and implement object-oriented solutions.
2. Describe the principles of inheritance, packages and Interface.
3. Design applications with threads in Java.
4. Appraise the concepts of Applets, JDBC and JavaScripts.
5. Create a graphical user interface (GUI) .
6. Identify Java APIs for program development.
7. Justify the JavaScript language & the Document Object Model.

UNIT - I

An Overview of Java – Data types – Variable – Operators – Expressions – Arrays – Control Statements. **Java Classes**: Fundamentals – Methods – Constructors – this Keyword – Method Overloading, Nested and inner classes.
Inheritance: Basics, using super – Method Overriding – Abstract Classes.

UNIT – II

Packages and Interfaces: Packages and Member Access – Importing packages.
Interface: Defining and Implementing Interface – Applying interface – Variables in Interface. **Exception Handling**: Fundamentals – Exception types – Uncaught exceptions – Using try and catch – Multiple catch clauses – Nested try statements – Java Built in Exception – Creating your own sub classes.

UNIT – III

Multi threaded Programming: Java thread model – Thread Class and runnable interface – Main thread – Creating the thread – Creating multiple threads – Synchronization – Inter thread communication.
I/O basics: Stream Classes – Predefined streams – Reading/Writing console input/output. **Java Library:** String handling – Utility Classes- Input/output.

UNIT – IV

Applet fundamentals: – Introduction of AWT - AWT Classes – Window fundamentals – Working with frame windows – Introducing graphics – Working with Colour – Working with Fonts.
Accessing database with JDBC: Relationship databases – creating a database in MySQL – Manipulating databases with JDBC – Transaction processing.

UNIT – V

JavaScript: Introduction – Syntax – Types – JavaScript Variables, Operators, Control and Looping Structures – Strings and Numbers – Display methods – Popup boxes – Date & Time – Function – Event Handlers – Document formatting – Math strings.

TEXT BOOKS

1. Herbert Schildt, "**The Complete Reference Java**", 9th Edition, Tata McGraw, 2014.
2. Deitel and Deitel, "**Java How to Program**", Seventh Edition, Pearson Education Asia, 2007.
3. Steven Bright, "**JavaScript Fundamentals**", 1stEdition, Monday Sadiku Publication, 2016.

REFERENCES

1. E Ramaraj, P Geetha, S Muthukumaran, " **Advanced Java Programming**", Yes Dee publishing,2020.

FURTHER READINGS

1. E Balagurusamy , **"Programming with Java",** Sixth edition ,Tata McGraw-Hill,2019.
2. https://nptel.ac.in/courses/106105191

Semester **I**		**Course Code** **21MCS13C**		**Title of the Paper** **ADVANCED JAVA PROGRAMMING**			**Hours** **5**	**Credits** **4**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**	**CO7**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**		✓		✓		✓	
	RESEARCH-RELATED SKILLS	✓		✓		✓		✓
	PROBLEM SOLVING		✓		✓			✓
	COMMUNICATION SKILLS	✓				✓	✓	
	LEADERSHIP READINESS/QUALITIES		✓		✓			
	ANALYTICAL REASONING		✓					✓
	MORAL AND ETHICAL AWARENESS	✓			✓			
	SCIENTIFIC REASONING		✓			✓		
	CRITICAL THINKING		✓		✓		✓	

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	DATA COMMUNICATION AND COMPUTER NETWORKS	I	21MCS14C

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Distinguish the rudiments of data communication and computer networks.
2. Infer the architecture and topologies of a number of different networks.
3. Explain the functional principles of TCP/IP layering.
4. Illustrate modern communications.
5. Relate the algorithmic approach of cryptography and Steganography**.**

UNIT – I

Data Communication: Components of a Data Communication System, Simplex, Half- Duplex and Duplex Modes of Communication; Analog and Digital Signals; Noiseless and Noisy Channels; Bandwidth, Throughput and Latency; Digital and Analog Transmission; Data Encoding and Modulation Techniques; Broadband and Baseband Transmission; Multiplexing, Transmission Media, Transmission Errors, Error Handling Mechanisms.

UNIT – II

Computer Networks: Network Topologies, Local Area Networks, Metropolitan Area Networks, Wide Area Network, Wireless Networks, Internet.

Network Models: Layered Architecture, OSI Reference Model and its Protocols; TCP/IP Protocol Suite, Physical, Logical, Port and Specific Addresses, Switching Techniques.

UNIT – III

Functions of OSI and TCP/IP Layers: Framing, Error Detection and Correction; Flow and Error Control; Sliding Window Protocol, HDLC, Multiple Access – CSMA/CD, CSMA/CA, Reservation, Polling, Token Passing, FDMA, CDMA, TDMA, Network Devices, Backbone Networks, Virtual LANs.

IPv4 Structure and Address Space; Classful and Classless Addressing; Datagram, Fragmentation and Checksum; IPv6 Packet Format, Mapping Logical to Physical Address (ARP), Direct and Indirect Network Layer Delivery; Routing Algorithms, TCP, UDP and SCTP Protocols; Flow Control, Error Control and Congestion Control in TCP and SCTP.

UNIT – IV

World Wide Web (WWW): Uniform Resource Locator (URL), Domain Name Service (DNS), Resolution - Mapping Names to Addresses and Addresses to Names; Electronic Mail Architecture, SMTP, POP and IMAP; TELNET and FTP.

UNIT – V

Network Security: Malwares, Cryptography and Steganography; Secret-Key Algorithms, Public-Key Algorithms, Digital Signature, Virtual Private Networks, Firewalls.

TEXT BOOKS

1. Behrouz A Forouzan,AchyutGodbole, “**Data communications and Networking**”, Fourth Edition, Tata Mc-Graw Hill, 2017.
2. A. S. Tanenbaum ,“**Computer Networks**”, 4th edition, Pearson Education/ PHI,2003.
3. Behrouz A. Forouzan , “**TCP / IP**”, 4th Edition, Tata Mc Graw-Hill, 2006.
4. W. Stallings,“**Cryptography and Network Security Principles and Practices**”, Fourth Edition, Prentice Hall, 2005.

REFERENCES

1. Dr. SidnieFeit, “**TCP/IP**”, Second Edition, Tata Mc Graw-Hill, 2008.
2. W. Stallings, “**Data and Computer Communications**”, 8thEdition, Pearson Education, 2007.

FURTHER READINGS

1. https://nptel.ac.in/courses/106108098
2. https://nptel.ac.in/courses/106105080

Semester I		Course Code 21MCS14C	Title of the Paper DATA COMMUNICATION AND COMPUTER NETWORKS		Hours 5	Credits 4
Course Outcome(CO)		CO1	CO2	CO3	CO4	CO5
Programme Learning Outcome (PLO)	DISPLINARY KNOWLEDGE		✓			✓
	RESEARCH-RELATED SKILLS	✓			✓	✓
	PROBLEM SOLVING	✓		✓		✓
	COMMUNICATION SKILLS	✓			✓	
	LEADERSHIP READINESS/QUALITIES		✓			✓
	ANALYTICAL REASONING	✓				✓
	MORAL AND ETHICAL AWARENESS		✓		✓	✓
	SCIENTIFIC REASONING	✓		✓		
	CRITICAL THINKING	✓				✓

	Subject Title	Sem.	Sub Code
2020 -21 Onwards	DATABASE MANAGEMENT SYSTEMS	I	21MCS15C

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Explain Data Models and Schemas to database management system.
2. Discuss the relational model and relational algebra operations.
3. Create, manipulate, and query databases with SQL commands.
4. Apply proper techniques, such as normalization, in designing a database.
5. Justify the concept of PL/SQL.
6. Relate the parallel and distributed data base systems.
7. Distinguish the different types of NoSQL databases.
8. Explain the impact of the cluster on database design.
9. Use the data control, definition, and manipulation languages of the NoSQL databases covered in the course.
10. Design Hadoop Distributed File System (HDFS) as a foundation for NoSQL technologies.

Unit I:

Database System Concepts and Architecture: Data Models, Schemas, and Instances; Three-Schema Architecture and Data Independence; Database Languages and Interfaces; Centralized and Client/Server Architectures for DBMS.

Data Modeling: Entity-Relationship Diagram, Relational Model - Constraints, Languages, Design, and Programming, Relational Database Schemas, Update Operations and Dealing with Constraint Violations; Relational Algebra and Relational Calculus, Codd Rules.

Unit II:

Normalization for Relational Databases: Functional Dependencies and Normalization; Algorithms for Query Processing and Optimization; Transaction Processing, Concurrency Control Techniques, Database Recovery Techniques, Object and Object-Relational Databases; Database Security and Authorization.

Unit III:

SQL: Data Definition and Data Types; Constraints, Queries, Insert, Delete, and Update Statements; Views, Stored Procedures and Functions; Database Triggers, SQL Injection.

Unit IV:

Enhanced Data Models: Temporal Database Concepts, Multimedia Databases, Deductive Databases, XML and Internet Databases; Mobile Databases, Geographic Information Systems, Genome Data Management, Distributed Databases and Client-Server Architectures.

Unit V:

Big Data Systems: Big Data Characteristics, Types of Big Data, Big Data Architecture, Introduction to Map-Reduce and Hadoop; Distributed File System, HDFS.

NOSQL: NOSQL and Query Optimization; Different NOSQL Products, Querying and Managing NOSQL; Indexing and Ordering Data Sets; NOSQL in Cloud.

TEXTBOOKS

1. Abraham Silberschatz, korth and Sudarsan “**Database System Concepts**” 7thedition,McGraw Hill International,2019.
2. S K Singh, “**Database Systems Concepts, Design and Applications**”, Pearson Education, 2006.
3. Alan beaulieu “**Learning SQL**” 2nd Edition, O’Reilly Publication,2009.
4. David Dietrich , Barry Heller, beibeiYang,**”Data Science and Big Data Analytics :Discovering, Analyzing, Visualizing and Presenting Data**, John Wiley & Sons, Inc , EMC education Service, 2014.

5 C.Balakrishnan, **“Fundamentals of NoSQL”**1stEdition,LAP LAMBERT Academic Publishing, 2014.

REFERENCES

1. CJ Date, “**An Introduction to Database Systems**”, 6th Edition, Addison Wesley Publishing Company New York, 1995.

FURTHER READINGS

1. Abraham Silberschatz, Henry F. Korth and S.Sudarashan , **“Database System Concepts”** Tata McGraw Hill, 7th edition ,2021.
2. https://nptel.ac.in/courses/106106220
3. https://nptel.ac.in/courses/106106095

Semester **I**		**Course** **Code** **21MCS35E**		**Title of the Paper** **DATABASE MANAGEMENT SYSTEMS**						**Hours** **5**	**Credits** **5**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**	**CO7**	**CO8**	**CO9**	**CO10**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**	✓	✓	✓		✓	✓	✓		✓	✓
	RESEARCH-RELATED SKILS	✓		✓	✓		✓		✓		
	PROBLEM SOLVING		✓		✓	✓		✓		✓	✓
	COMMUNICATION SKILLS	✓		✓		✓	✓		✓		✓
	LEADERSHIP READINESS/QUALITIES						✓			✓	
	ANALYTICAL REASONING		✓		✓		✓		✓		✓
	MORAL AND ETHICAL AWARENESS	✓								✓	✓
	SCIENTIFIC REASONING	✓		✓	✓		✓	✓		✓	✓
	CRITICAL THINKING	✓	✓		✓		✓	✓	✓		

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	Core Practical 1: ADVANCED JAVA PROGRAMMING LAB	I	21MCS16P

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Test various programming Techniques in Java practically.
2. Create Internet Programming, using Java Applets .
3. Apply event handling on AWT and Swing components.
4. Design Java programs to access database, using Java Data Base Connectivity (JDBC) .
5. Create dynamic web pages, using Servlets and JSP.

LIST OF PRACTICALS

1. Write a java program to create a class, objects using constructor.
2. Write a java program to implement inheritance.
3. Write a java program to illustrate run time exception and I/O exception.
4. Write a java program to create a package.
5. Create an Employee package to maintain the information about the employee. Use the Constructors to initialize the Employee number and use Overloading method to get the Basic pay of the employee. By using package create a java program.
6. Write a java program to implement multithreading.
7. Write a java program to create GUI components.
8. Write a java program to draw images and animate them.
9. Write a java program that connects to a database using JDBC and performs addition, deletion, modification and retrieval operation.
10. Write a java program to design a web page using applet and HTML.
11. Write a java script to perform all arithmetic operations.
12. Write a java Script to search and element in an array.

<table>
<tr><td colspan="2">Semester
I</td><td>Course
Code
21MCS16P</td><td colspan="3">Title of the Paper
ADVANCED JAVA
PROGRAMMING
LAB</td><td>Hours
3</td><td>Credits
2</td></tr>
<tr><td colspan="2">Course Outcome(CO)</td><td>CO1</td><td>CO2</td><td>CO3</td><td>CO4</td><td colspan="2">CO5</td></tr>
<tr><td rowspan="9">Programme Learning Outcome (PLO)</td><td>DISPLINARY KNOWLEDGE</td><td></td><td>✓</td><td></td><td></td><td colspan="2">✓</td></tr>
<tr><td>RESEARCH-RELATED SKILLS</td><td>✓</td><td></td><td>✓</td><td></td><td colspan="2">✓</td></tr>
<tr><td>PROBLEM SOLVING</td><td></td><td>✓</td><td></td><td>✓</td><td colspan="2"></td></tr>
<tr><td>COMMUNICATION SKILLS</td><td>✓</td><td></td><td>✓</td><td></td><td colspan="2">✓</td></tr>
<tr><td>LEADERSHIP READINESS/QUALITIES</td><td></td><td></td><td>✓</td><td></td><td colspan="2"></td></tr>
<tr><td>ANALYTICAL REASONING</td><td>✓</td><td></td><td></td><td></td><td colspan="2">✓</td></tr>
<tr><td>MORAL AND ETHICAL AWARENESS</td><td></td><td></td><td>✓</td><td></td><td colspan="2">✓</td></tr>
<tr><td>SCIENTIFIC REASONING</td><td>✓</td><td></td><td>✓</td><td></td><td colspan="2">✓</td></tr>
<tr><td>CRITICAL THINKING</td><td>✓</td><td></td><td></td><td></td><td colspan="2">✓</td></tr>
</table>

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	Core Practical 2: DBMS LAB	I	21MCS17P

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Plan good formal foundation on the relational model of data.
2. Recommend SQL and procedural interfaces to SQL comprehensively.
3. Order techniques relating to query processing by SQL engines.
4. Organize techniques relating to ODBC and its implementations.
5. Design Record Management using Cursors.

LIST OF PRACTICALS

1. DDL & DML – data types, create, alter, drop table, integrity constraints
2. Insert, delete and update commands
3. DCL & TCL – grant, revoke, rollback and commit
4. Select command with operators like arithmetic, comparison, logical, order by, group by etc.
5. SQL Functions – date, numeric, character, conversion, avg, max, min, sum,count
6. Set operations – union, intersect and minus
7. Join query concept – simple, equi, non–equi, self, outerjoin
8. Complex queries and subqueries
9. Database objects – view, synonym, index, sequence – create, alter and drop
10. Report writer using SQL
11. PL/SQL – Introduction – character set, data types –execution
12. PL/SQL attributes %type, %rowtype, function comparison, if condition, loop, for, while and goto etc.
13. Record management using cursors
14. Function – definition and implementation
15. Database triggers – syntax, parts and types of triggers

Semester I		Course Code 21MCS17P	Title of the Paper DBMS LAB		Hours 2	Credits 2
Course Outcome(CO)		CO1	CO2	CO3	CO4	CO5
Programme Learning Outcome (PLO)	DISPLINARY KNOWLEDGE	✓			✓	
	RESEARCH-RELATED SKILLS		✓			✓
	PROBLEM SOLVING			✓		✓
	COMMUNICATION SKILLS	✓			✓	✓
	LEADERSHIP READINESS/QUALITIES	✓		✓		✓
	ANALYTICAL REASONING		✓		✓	
	MORAL AND ETHICAL AWARENESS	✓			✓	
	SCIENTIFIC REASONING	✓		✓		✓
	CRITICAL THINKING		✓		✓	

	Subject Title	Sem.	Sub Code
2021 -22 Onwards	DATA MINING WITH R PROGRAMMING	II	21MCS21C

COURSE OUTCOMES

On successful completion of the course the students will be able to:

1. Discuss the basics in R programming in terms of constructs, control statements, string functions.
2. Priorities the use of R for Big Data analytics.
3. Identify the characteristics of Big Data, recognize the challenges of Big Data and be acquainted with the application domains for Big Data and know how to set up a Big Data environment.
4. Design R programming for Text processing.
5. Apply the R programming from a statistical perspective.

UNIT-I: Introduction – Data – Types of Data – Data Mining Functionalities – Interestingness of Patterns – Classification of Data Mining Systems – Data Mining Task Primitives –Association rule mining: Mining Frequent Patterns, Associations and Correlations – Mining Methods – Mining various kinds of association rules.

UNIT-II: Classification and Clustering: Classification and Prediction - Basic Concepts- Decision Tree Induction - Bayesian Classification – Rule Based Classification – Classification by Back Propagation Cluster Analysis - Types of Data – Categorization of Major Clustering Methods–K-means-Partitioning Methods – Hierarchical Methods – Clustering High Dimensional Data- Constraint Based Cluster Analysis – Outlier Analysis – Data Mining Applications.

UNIT-III: Big Data Analytics: Introduction to the Big Data Era – Description of Big Data – Industry Examples of Big Data – Descriptive power and predictive Pattern Matching – The Value of Data – Big Data Analytics – Architectures, Frameworks, and Tools – Big Data Analytics Methodology – Challenges – Big Data Analytics in Healthcare.

UNIT-IV: Getting Started with R- R Nuts and Bolts - Getting Data in and Out of R - Using Textual and Binary Formats for Storing Data- Interfaces to the Outside World- Sub setting R Objects - Vectorized Operations - Managing Data Frames with the dplyr package.

UNIT-V: Control Structures -Functions- Scoping Rules of R - Loop Functions- Debugging Tool in R-Profiling R Code- Simulation.

TEXT BOOKS

1. Jiawei Han and Micheline Kamber, "**Data Mining Concepts and Techniques**",SecondEdition, Elsevier, 2007. (Unit I andII)
2. Stephan Kudyba Foreword by Thomas H.Davenport, "**Big Data, Mining, and Analytics**", CRC Press, 2015. (UnitIII)
3. Roger D. Peng, "**R Programming for Data Science**" Lean Publishing, 2014. (Unit IV &V).

REFERENCES

1. K.P. Soman, ShyamDiwakar and V. Ajay, "**Insight into Data mining Theoryand Practice**", Easter Economy Edition, Prentice Hall of India,2006.

2. G. K. Gupta, “**Introduction to Data Mining with Case Studies**”, Easter Economy Edition, Prentice Hall of India,2006.
3. Alain F. Zuur, Elena N. Ieno, Erik H.W.G. Meesters, “**A Beginner’s Guide to R**”, Springer,2009.
4. Graham Williams, “**Data Mining with Rattle and R**”, Springer Science,2011.

FURTHER READING

1. Luis Torgo , **“Data Mining with R: Learning with Case Studies”,** Second Edition, Chapman and Hall/CRC publication,2010.
2. **https://nptel.ac.in/courses/106105174**
3. **https://nptel.ac.in/courses/110107092**

Semester **II**		**Course Code** **21MCS21C**	**Title of the Paper** **DATA MINING WITH R PROGRAMMING**		**Hours** **5**	**Credits** **4**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**	✓		✓		✓
	RESEARCH-RELATED SKILLS	✓			✓	
	PROBLEM SOLVING		✓		✓	
	COMMUNICATION SKILLS	✓		✓		✓
	LEADERSHIP READINESS/QUALITIES		✓	✓		
	ANALYTICAL REASONING	✓			✓	✓
	MORAL AND ETHICAL AWARENESS		✓	✓		
	SCIENTIFIC REASONING	✓		✓		✓
	CRITICAL THINKING		✓		✓	✓

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	OPERATING SYSTEMS	II	21MCS22C

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Outline the advanced concepts in Operating System.
2. Interpret the Operating file system and programming for inter process communication and Classical IPC Problems.
3. Justify different process scheduling algorithm and synchronization techniques to avoid deadlock.
4. Infer different memory management techniques like paging, segmentation and demand paging etc.
5. Differentiate architecture of UNIX operating system, system calls, memory management policies and its working principles.

UNIT – I

Introduction – Evolution of Operating systems – Serial, Simple Batch, Multi programmed Batch, Timesharing, Distributed and Real time operating systems – System calls- Structure of Operating System – Interrupts- Processes Model – Creation – Termination – Process Hierarchy – Process States – Implementation of Processes – PROCESS AND THREADS: Relationship between process and threads – Thread State – Thread Synchronization – Types of Thread – Multithreading model.

UNIT – II

Inter Process Communication – Concurrent Process – Deadlock - Race condition – Critical Region – Mutual Exclusion – Sleep and wakeup – Semaphores – Mutexes –Peterson's Solution- Message Passing. Classical IPC Problems: The Dining Philosophers Problem – The Readers and Writers Problem – The Sleeping Barber Problem – Producer Consumer problem - Deadlock Prevention, Avoidance and Detection and recovery.

UNIT – III

MEMORY MANAGEMENT: Memory hierarchy – Linking and Loading the process – Memory Management requirement - Fixed partitioning - Dynamic partitioning – Buddy Systems – Simple paging – Multilevel paging – Simple Segmentation – segmentation and paging.

VIRTUAL MEMORY MANAGEMENT: Need for Virtual Memory management – Demand Paging – Copy on write - Page Fault handling – Demand Segmentation – Combined demand segmentation and paging - Thrashing- working set model.

UNIT – IV

FILE SYSTEM MANAGEMENT: Files – Access methods - File System Architecture – Functions of File Management –Directory and disk structure – file sharing – File system implementation – directory implementation - File Allocation – free space management.

I/O MANAGEMENT AND DISK SCHEDULING: Organization of I/O function – Evolution of I/O function – Types of I/O devices – Logical Structure of I/O functions – I/O Buffering – Disk I/O – Disk Scheduling algorithms – Disk Cache.

UNIT – V

LINUX: Design Priciples-Kernel Modules-Process Management- Scheduling-memory Management- File Systems-input and Output-Inter Process Communication- Network Structure.

TEXT BOOKS

1. William Stallings, "**Operating Systems**", Second Edition ,Prentice Hall of India, 2000. [Units I, II, III,IV].
2. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, "**Operating System Concepts**", 8th Edition, Wiley,2013.

REFERENCES

1. Andrew S.Tanenbaum, "**Modern Operating Systems**",Second Edition, PHI/Pearson Education Asia, 2001.

FURTHER READINGS

1. Vikram Sindhu S and Dr.Mohanmmed Rafi , **"Operating System"** , Rigi Publication 2020.
2. **https://nptel.ac.in/courses/106106144**

Semester **II**		**Course Code** **21MCS22C**	**Title of the Paper** **OPERATING SYSTEMS**		**Hours** **5**	**Credits** **4**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**		✓	✓		✓
	RESEARCH-RELATED SKILLS	✓		✓		✓
	PROBLEM SOLVING	✓	✓		✓	
	COMMUNICATION SKILLS			✓	✓	✓
	LEADERSHIP READINESS/QUALITIES	✓	✓		✓	
	ANALYTICAL REASONING	✓				✓
	MORAL AND ETHICAL AWARENESS		✓	✓		
	SCIENTIFIC REASONING	✓			✓	✓
	CRITICAL THINKING		✓	✓		

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	DATA SCIENCE USING PYTHON	II	21MCS23C

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Describe the basic concepts of Python.
2. Prepare and pre-process data.
3. Judge the results of analytics effectively.
4. Differentiate Numpy and Pandas.
5. Use conditional loops and list by python.
6. Appraise Visualization through Matplotlib.

Unit-I:

Why Python for Data Analysis? - Essential Python Libraries – Installation and setup python basics: The python Interpreter- Ipython Basics- Data Structure and Sequences: Tuple – list.

Unit-II

NumPy Basics: Arrays and Vectorized Computation -The Numpynd array: A Multidimensional Array Object - Universal Functions: Fast Element-wise Array Functions - File Input and Output with Arrays - Linear Algebra -Random Number Generation.

Unit-III

Getting started with pandas: Introduction to pandas Data Structures -Essential Functionality - Summarizing and Computing Descriptive Statistics- Handling Missing Data -Hierarchical Indexing - Other pandas Topics.

Unit-IV

Data Loading, Storage, and File Formats: Reading and Writing Data in Text Format - Binary Data Formats - Interacting with HTML and Web APIs -Interacting with Databases - Data Wrangling: Clean, Transform, Merge, Reshape.

Unit-V

Plotting and Visualization: A Brief matplotlib API Primer - Plotting Functionsin pandas -Python Visualization Tool Ecosystem - Time Series.

TEXT BOOK

1. Wes McKinney, "**Python for Data Analysis**",2nd Edition,O'ReillyMedia, 2012.

REFERENCES

1. Jake VanderPlas, “**Python Data Science Handbook: Essential Tools for Working with Data**” , O’ReillyMedia , 2016.

FURTHER READING

1. https://nptel.ac.in/courses/106106179
2. https://nptel.ac.in/courses/106106212

Semester II		Course Code 21MCS23C	Title of the Paper DATA SCIECE USING PYTHON			Hours 5	Credits 4
Course Outcome(CO)		CO1	CO2	CO3	CO4	CO5	
Programme Learning Outcome (PLO)	DISPLINARY KNOWLEDGE		✓		✓		✓
	RESEARCH-RELATED SKILLS	✓				✓	
	PROBLEM SOLVING		✓		✓		✓
	COMMUNICATION SKILLS		✓	✓			
	LEADERSHIP READINESS/QUALITIES	✓		✓	✓		✓
	ANALYTICAL REASONING		✓			✓	
	MORAL AND ETHICAL AWARENESS	✓		✓			✓
	SCIENTIFIC REASONING		✓	✓		✓	
	CRITICAL THINKING		✓		✓		

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	SOFTWARE ENGINEERING CONCEPTS	II	21MCS24C

COURSE LEVEL OUTCOME

On successful completion of the course the students will be able to:

1. Describe Software Engineering, Various models, Software Design, Software Development and Various Testing Strategies.
2. Discuss engineering layered technology and Process frame work.
3. Justify the role of project management including planning, scheduling, risk management, etc.
4. Analyze approaches to verification and validation including static analysis, and reviews.
5. Assess software testing approaches such as unit testing and integration testing.
6. Recommend quality control and how to ensure good quality software.

UNITI

SOFTWARE ENGINEERING: Software Engineering – A Layered Technology – A Process Framework - CMMI – **PROCESS MODELS:** Prescriptive Models – The Waterfall Model – Incremental Process Model – Evolutionary Process Model - Specialized Process Model. **SYSTEM ENGINEERING:** The System Engineering Hierarchy. **REQUIREMENTS ENGINEERING:** Requirements Engineering Tasks – Initiating the Requirements Engineering Process.

UNIT II

BUILDING THE ANALYSIS MODEL: Requirements Analysis - Data Modeling Concepts – Flow Oriented Modeling.**DESIGN ENGINEERING:** Design Process – Design Concepts – Design Model. **ARCHITECTURAL DESIGN:** Software Architecture – Architectural Styles and Patterns - Architectural Design. **COMPONENT- LEVEL DESIGN:** Component – Designing Class Based Components. **UI DESIGN:** The Golden Rules - UI Analysis and Design.

UNIT III

METRICS FOR PROCESS AND PROJECTS: Metrics in the Process and Project Domains – Software Measurement – Metrics for Software Quality. **ESTIMATION FOR SOFTWARE PROJECT:** Resources – Decomposition Techniques. **PROJECT SCHEDULING:** Project Scheduling - Defining a Task Set for the Software Project.

UNIT IV

RISK MANAGEMENT: Software Risks – Risk Identification – Risk Projection.

QUALITY MANGEMENT: Quality Concepts – Software Quality Assurance – Formal Technical Reviews – Software Reliability.

CHANGE MANAGEMENT: Software Configuration Management – The SCM Process.

UNIT V

SOFTWARE TESTING: A Strategic Approach to Software Testing – Test, Test Case and

Test Suite – Verification and Validation – Alpha, Beta and Acceptance Testing – Functional Testing – Structural Testing – Levels of Testing – Validation Testing – The Art of debugging – Testing Tools.

TEXT BOOKS

1. Roger S. Pressman, "**Software Engineering – A Practitioners Approach**", 6th Edition , Tata McGraw Hill International, 2005. (UNITS: I, II, III &IV).
2. K.K. Aggarwal, Yogesh Singh, "**Software Engineering**", 3rd Edition, New Age International Publishers, 2008. (UNIT:V).

REFERENCES

1. Ian Sommerville, "**Software Engineering**", Eighth Edition, Pearson Education, 2009.
2. Srinivasan Desikan and Gopalasamy Ramesh, "**Software Testing for Principles and Practices**", Pearson Education , 2007.

FURTHER READING

1. Richard Fairley, **"Software Engineering Concepts"** Tata McGraw Hill Education ,2017.
2. https://nptel.ac.in/courses/106101061

Semester **II**		**Course Code** **21MCS24C**	**Title of the Paper** **SOFTWARE ENGINEERING CONCEPTS**			**Hours** **5**	**Credits** **4**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**		✓		✓		✓
	RESEARCH-RELATED SKILLS		✓			✓	
	PROBLEM SOLVING	✓		✓	✓		
	COMMUNICATION SKILLS			✓		✓	
	LEADERSHIP READINESS/QUALITIES	✓			✓		✓
	ANALYTICAL REASONING		✓	✓		✓	
	MORAL AND ETHICAL AWARENESS	✓			✓		✓
	SCIENTIFIC REASONING		✓	✓			✓
	CRITICAL THINKING		✓		✓	✓	

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	Core Practical 3: DATA MINING WITH R PROGRAMMING LAB	II	21MCS26P

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Outline the concepts of Data mining techniques practically.
2. Discuss the R Programming environment and basics of R's syntax used to implement the Data mining techniques.
3. Infer the functionality of R by using add-on packages.
4. Extract data from files and other sources and perform various data manipulation tasks on them.
5. Code statistical functions in R.
6. Use R Graphics and Tables to visualize results of various statistical operations on data.
7. Apply the knowledge of R gained to data Analytics for real life applications.

LIST OF PRACTICALS

1. Write an R program for classification using decision tree using PARTY PACKAGE.
2. Write an R program for clustering using K-Means.
3. Write an R program for Partitioning around Medoid (PAM)Clustering.
4. Write an R program for Hierarchical clustering.
5. Write an R program for Association Rule Mining.
6. Write an R program for outlier detection.
7. Write an R program to visualize the data using histogram.
8. Write an R Program to visualize the data using Box plot.
9. Write an R program to visualize the data using Scatterplot.
10. Write an R program to implement pre processing concept.

Semester II		Course Code 21MCS26P	Title of the Paper DATA MINING WITH R PROGRAMMING LAB				Hours 3	Credits 2
Course Outcome(CO)		CO1	CO2	CO3	CO4	CO5	CO6	CO7
Programme Learning Outcome (PLO)	DISPLINARY KNOWLEDGE	✓			✓			✓
	RESEARCH-RELATED SKILLS		✓		✓	✓		
	PROBLEM SOLVING			✓			✓	
	COMMUNICATION SKILLS	✓			✓			✓
	LEADERSHIP READINESS/QUALITIES		✓				✓	
	ANALYTICAL REASONING	✓		✓		✓		
	MORAL AND ETHICAL AWARENESS		✓		✓			✓
	SCIENTIFIC REASONING	✓					✓	
	CRITICAL THINKING		✓		✓		✓	

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	Core Practical 4: PYTHON PROGRAMMING LAB	II	21MCS27P

COURSE LEVELOUTCOMES

On successful completion of the course the students will be able to:

1. Design forms using various functions.
2. Apply rich controls and conditional statement logic in Python.
3. Demonstrate the functionality of stack and regular expressions throughPython.
4. Create and manipulate array functions using NumPy.
5. Create indexing scripts using Pandas.
6. Build applications using Pandas.

List of Practical's:

1. Write Python applications using variables, data types.

2. Write Python application using strings and functions.

3. Write Python applications using loops, arrays, sorting.

4. Write Python applications using dictionaries, lists and tuples.

5. Write Python applications using matrices.

6. Create Calculator Program.

7. Array Function using NumPy.

8. Aggregation function using NumPy.

9. Data Operation using Scipy Basics.

10. Pandas Basics.

11. Twitter API Integration for tweet Analysis.

Semester II		Course Code 21MCS27P	Title of the Paper PYTHON PROGRAMMING LAB			Hours 2	Credits 2
Course Outcome(CO)		CO1	CO2	CO3	CO4	CO5	CO6
Programme Learning Outcome (PLO)	DISPLINARY KNOWLEDGE	✓		✓		✓	
	RESEARCH-RELATED SKILLS				✓		✓
	PROBLEM SOLVING		✓				✓
	COMMUNICATION SKILLS	✓		✓		✓	
	LEADERSHIP READINESS/QUALITIES		✓		✓		✓
	ANALYTICAL REASONING	✓		✓		✓	
	MORAL AND ETHICAL AWARENESS		✓				
	SCIENTIFIC REASONING			✓		✓	
	CRITICAL THINKING	✓			✓		✓

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	DIGITAL IMAGE PROCESSING	III	21MCS31C

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Recall the need for image transforms different types of image transforms and their properties.
2. Compare different techniques employed for the enhancement of images.
3. Distinguish different causes for image degradation and overview of image restoration techniques.
4. Infer the need for image compression and to learn the spatial and freque ncy domain techniques of image compression.
5. Design different feature extraction techniques for image analysis and recognition.

UNIT – I

Introduction: What is Digital image processing – the origin of DIP – Examples of fields that use DIP – Fundamentals steps in DIP – Components of an image processing system. Digital Image Fundamentals: Elements of Visual perception – Light and the electromagnetic spectrum – Image sensing and acquisition – Image sampling and Quantization – Some Basic relationship between Pixels.

UNIT – II

Image Enhancement in the Spatial Domain: Background – some basic Gray level Transformations – Histogram Processing – Enhancement using Arithmetic / Logic operations – Basics of spatial filtering – Smoothing spatial filters – Sharpening spatial filters.

UNIT – III

Color Image Processing: Color Fundamentals-Color Models-Pseudo color Image Processing-Color transformations-Smoothing and Sharpening-Color Segmentation-Noise in Color Images.

UNIT – IV

Morphological Image processing: Preliminaries-Dilation and Erosion-Opening and Closing-The Hit-or-Miss Transformation-Some Basic Morphological Algorithms. Image Segmentation: Detection and Discontinuities – Edge Linking and Boundary detection – Thresholding – Region-Based segmentation – Segmentation by Morphological watersheds.

UNIT – V

Image Processing using Python : Introduction to Python-Python Libraries-numpy, matplotlib, pillow(PIL) – Image Open- Image Read –Image Save- Image Storage Formats- Image mode – Changing Color Spaces-Image Arithmetic operations -Image Negative - Image Histogram–Image

Smoothing & Sharpening functions - Geometric Transformation of Images-Computing Image features(Statistical)–Edge detection functions.

TEXT BOOKS

1. Rafael C. Gonzalez, Richard E. Woods, "**Digital Image Processing**", 4th Edition, PHI/PearsonEducation,2018.
2. Ashwin Pajankar, **"Python 3 Image Processing"**, BPB Publications, 2019.

REFERENCES

1. B. Chanda, D. Dutta Majumder, "**Digital Image Processing and Analysis**", 2nd Edition, PHI,2003.
2. Nick Efford, "**Digital Image Processing a practical introducing using Java**", Pearson Education,2004.
3. SandipanDey, **"Hands-On Image Processing with Python**", Packt Publication, 2018.

FURTHER READING

1. https://nptel.ac.in/courses/117105079

Semester **III**		**Course Code** **21MCS31C**	**Title of the Paper** **DIGITAL IMAGE PROCESSING**		**Hours** **5**	**Credits** **4**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**	✓		✓		
	RESEARCH-RELATED SKILLS		✓		✓	
	PROBLEM SOLVING		✓			✓
	COMMUNICATION SKILLS	✓		✓		
	LEADERSHIP READINESS/QUALITIES		✓		✓	✓
	ANALYTICAL REASONING	✓		✓		
	MORAL AND ETHICAL AWARENESS			✓	✓	
	SCIENTIFIC REASONING	✓		✓		✓
	CRITICAL THINKING	✓		✓	✓	

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	CLOUD COMPUTING	III	21MCS32C

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Reproduce the Concepts & Technologies Associated with Cloud Computing.
2. Analyze the trade-offs between deploying applications in the cloud and over the local infrastructure.
3. Compare the advantages and disadvantages of various cloud computing platforms.
4. Deploy applications over commercial cloud computing infrastructures such as Amazon Web Services, Aneka frame work, and IBM blue.
5. Program data intensive parallel applications in the cloud.
6. Analyze the performance, scalability, and availability of the underlying cloud technologies and software.
7. Identify security and privacy issues in cloud computing.

UNIT – I

Introduction to Parallel and Distributed Computing: Introduction, Architecture and Distributed computing models and technologies, SOA, Web Services.

Grid, Cluster and Utility Computing: Introduction, Architecture, Pros & Cons, Real time applications.

UNIT – II

Introduction to Cloud Computing: Definition, History, Comparison of Cloud Computing with Grid, Cluster and Utility Computing, Deployment models – Private, Public, Hybrid and Community - Pros and Cons of Cloud Computing. SaaS, PaaS, IaaS etc.

UNIT – III

Virtualization: Types of Virtualization, Tools for Virtualization, Architecture of VMM, Virtualization for Cloud.

Advanced Web Technologies: AJAX and Mashup – Programming examples using applications.

UNIT – IV

Map Reduce Paradigms: Introduction, GFS Architecture, HDFS Architecture, Hbase, Google big Table, Amazon's (key value) pair storage and Microsoft's Azure infrastructure, Map reduce programming examples

UNIT – V

Cloud Computing Framework: Amazon EC2, S3 storage revises, Aneka frame work, IBM blue Cloud.

Applications: Distributed search engine and distributed data mining in the cloud.

TEXT BOOKS

1. Anthony T Velte, Toby J Velte and Robert Elsenpeter, "**Cloud Computing: A Practical Approach**", Tata McGraw Hill, New Delhi,2010.
2. Liu M L, "**Distributed Computing Principles and Applications**",4th Edition, Pearson Education, New Delhi,2009.

REFERENCES

1. Author Erl, "**Cloud Computing: Concepts, Technology & Architecture**" 1st edition ,Pearson Education India, 2014.

FURTHER READING

1. Author Erl "**Cloud Computing Design Patterns**" 1st edition, Pearson Education India, 2015.
2. **https://nptel.ac.in/courses/106105223**

Semester **III**		**Course Code** **21MCS32C**		**Title of the Paper** **CLOUD COMPUTING**			**Hours** **5**	**Credits** **4**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**	**CO7**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**	✓				✓		✓
	RESEARCH-RELATED SKILLS		✓		✓		✓	
	PROBLEM SOLVING		✓		✓			✓
	COMMUNICATION SKILLS	✓		✓			✓	
	LEADERSHIP READINESS/QUALITIES		✓			✓		✓
	ANALYTICAL REASONING	✓		✓			✓	
	MORAL AND ETHICAL AWARENESS		✓			✓		✓
	SCIENTIFIC REASONING	✓			✓			
	CRITICAL THINKING				✓			✓

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	WEB PROGRAMMING ESSENTIALS	III	21MCS33C

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Outline Internet, HTML5, CSS3, JavaScript, PHP with a view to developing professional software development skills.
2. Explain functionalities of World Wide Web.
3. Explore mark-up languages features and create interactive web pages using them.
4. Design Client side validation using scripting languages.
5. Illustrate Open source JavaScript libraries.
6. Design front end web page and connect to the back end databases.

UNIT I:

INTRODUCTION TO WWW

Internet Standards – Introduction to WWW – WWW Architecture – SMTP – POP3 – File Transfer Protocol - Overview of HTTP, HTTP request – response — Generation of dynamic web pages.

UNIT II

Markup Language (HTML5): Basics of Html -Syntax and tags of Html- Introduction to HTML5 -Semantic/Structural Elements -HTML5 style Guide and Coding Convention– Html Svg and Canvas – Html API's - Audio & Video - Drag/Drop - Local Storage - Web socket API– Debugging and validating Html.

Cascading Style Sheet (CSS3): The need for CSS – Basic syntax and structure Inline Styles – Embedding Style Sheets - Linking External Style Sheets - Introduction to CSS3 – Backgrounds - Manipulating text - Margins and Padding - Positioning using CSS -Responsive Web Design - Introduction to LESS/SASS

UNIT III

OVERVIEW OF JAVASCRIPT

Introduction - Core features - Data types and Variables - Operators, Expressions, and Statements Functions - Objects - Array, Date and Math Related Objects - Document Object Model - Event Handling - Controlling Windows & Frames and Documents - Form validations.

UNIT IV JavaScript ES6

The let Keyword- the const keyword-Javascript Arrow Functions- Javascript For/of-Javascript Classes- Javascript promises-Javascript Symbol- default Parameters-Fuction Rest parameter-Array.find() and Array.findindex().

UNIT V Node.js

Node.js Introduction-Node.js Modules-Node.js File System-Node.js URLModule-Node.js NPM-Node.js Events-Node.js Upload Files-Node.js Email –Connecting with DB(Node.js with Mongo DB).

TEXT BOOKS:

1. David Flanagan, "**JavaScript: The Definitive Guide**", Sixth Edition, O'Reilly Media,2011.

2. Harvey & Paul Deitel& Associates, Harvey Deitel and Abbey Deitel, "**InternetandWorldWide Web - How To Program**", Fifth Edition, PearsoEducation,2011.
3. James Lee, BrentWare, "**Open Source Development with LAMP: Using Linux, Apache, MySQL, Perl, and PHP**" , 4th Edition, Addison Wesley, Pearson 2009.
4. Thomas A. Powell, "**HTML & CSS: The Complete Reference**", Fifth Edition,2010.
5. Thomas A Powell, Fritz Schneider, "**JavaScript: The Complete Reference**",Third Edition, Tata McGraw Hill,2013.
6. **Unit IV:https://www.w3schools.com/js/js_es6.asp**
7. **Unit V: https://www.w3schools.com/nodejs/nodejs_intro.asp**

REFERENCES

1. Shriram K. Vasudevan, SunandhiniMuralidharan, Meenakshi Sundaram, Chandni Suresh, **"Essentials of Internet Programming"**,I K International Publishing House Pvt. Ltd, 2015.

FURTHER READING

1. David Sawyer McFarland, **"JavaScript and JQuery: Interactive Front-End Web Development"** O'Reilly Media ,2008.

Semester **III**		**Course Code** **21MCS33C**	**Title of the Paper** **WEB PROGRAMMING ESSENTIALS**			**Hours** **5**	**Credits** **4**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**		✓		✓		✓
	RESEARCH-RELATED SKILLS	✓		✓			✓
	PROBLEM SOLVING	✓		✓			✓
	COMMUNICATION SKILLS		✓			✓	
	LEADERSHIP READINESS/QUALITIES	✓		✓	✓		✓
	ANALYTICAL REASONING		✓			✓	
	MORAL AND ETHICAL AWARENESS			✓		✓	✓
	SCIENTIFIC REASONING	✓			✓		
	CRITICAL THINKING		✓			✓	✓

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	MOBILE APPLICATIONS DEVELOPMENT	III	21MCS34C

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Recognize the components and structure of mobile application development frameworks for Android.
2. Discuss the SQLite on Android.
3. Recognizes mobile computing platforms and mobile computing.
4. Distinguish the basic concepts of Android phone features and capabilities.
5. Design Android application project.
6. Recognizes Android application's life cycle states and uses life cycle methods.
7. Select and use various layouts such as Linear Layout, Relative Layout, and Slide Drawer in constructing the view of Android applications.
8. Discriminate padding and margins to manage layout in device independent pixels (DIP).
9. Discuss how fragments should be used in newer versions of Android.
10. Recognize and use Android Selection Widgets.

UNIT-I

INTRODUCTION: Introduction to mobile applications - Importance of mobile applications – Strategies and challenges – Software and hardware requirements for developing mobile applications – Types of mobile applications – Benefits of creating mobile applications – Marketing and advertising mobile applications

UNIT-II

MOBILE USER INTERFACE DESIGN: Mobile application users – Social aspect of mobile interfaces - Accessibility – Design patterns – Designing for the platforms.

UNIT-III

MOBILE APPLICATIONS ARCHITECTURE: Smart Client – Smart Client Architecture – Messaging Architecture – The Model-View-Controller Model - Delegate Pattern- Building Smart Client Applications-Design, Development, implementation, testing and deployment phase- MVVM mobile architecture design.

UNIT - IV

MOBILE APPLICATION DEVELOPMENT: Introduction to Android Platform – Android architecture overview - Application life cycle - UI design for Android - UI fragments - Different types of layouts – Widgets – List view – View pager - Dialogs.

UNIT-V

DATABASE: Files and database – SQLite on Android – Loading asynchronous data - Map API.

TEXT BOOKS

1. Jeff McWherter and Scott Gowell , "**Professional Mobile Application Development**",

John Wiley & Sons,2012.

2. Bill Philips, Kristin Marsicano and Chris Stewart, "**Android Programming: The big Nerd Ranch guide**", 2ndEdition,O'Reilly,2017.
3. Martyn Mallick, "**Mobile and Wireless Design Essentials**", Wiley,2003.

REFERENCES

1. Ronan Schwarz, Phil Dutson, James Steele and Nelson To, "**The AndroidDeveloper's Cookbook -Building Applications with the Android SDK**", Addison Wesley,2013.
2. Mark Murphy, "**The Busy Coder's Guide to Android Development**", Commons Ware,2009.

FURTHER READING

1. Anubhav Pradhan and Anil V Deshpande, "**Composing Mobile App, Learn, Explore, Apply**" Wiley ,2014.
2. https://nptel.ac.in/courses/106106147

Semester III		Course Code 21MCS34C		Title of the Paper MOBILE APPLICATIONS DEVELOPMENT					Hours 5	Credits 4
Course Outcome(CO)		CO1	CO2	CO3	CO4	CO5	CO6	CO7	CO8	CO9
Programme Learning Outcome (PLO)	DISPLINARY KNOWLEDGE	✓		✓		✓				✓
	RESEARCH-RELATED SKILLS		✓				✓	✓		✓
	PROBLEM SOLVING	✓				✓		✓		
	COMMUNICATION SKILLS		✓		✓		✓			✓
	LEADERSHIP READINESS/QUALITIES	✓				✓			✓	
	ANALYTICAL REASONING		✓			✓		✓		✓
	MORAL AND ETHICAL AWARENESS	✓		✓		✓		✓		
	SCIENTIFIC REASONING		✓		✓		✓		✓	
	CRITICAL THINKING	✓				✓		✓		

Year	Subject Title	Sem.	Sub Code

2021 -22 Onwards	**Core Practical 5: DIGITAL IMAGE PROCESSING USING PYTHON LAB**	**III**	**21MCS36P**

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Explain various Image Processing Techniques using Python.
2. Justify the image formation and the role of human visual system play in perception of gray and color image data.
3. Execute image arithmetic operations.
4. Rewrite geometric transformations to images like scaling, rotation, translation, etc.
5. Create the edge algorithm and gradient filters and image segmentation using watershed algorithm.

LIST OF PRACTICALS

1) Write a python program to read an image file and write into different image formats.
2) Write a python program to find the neighbor pixels N_4,N_8& N_d at particular pixel (i,j).
3) Write a python program to perform the following basic gray level transformations,
    a) Image negatives
    b) Log transformation
4) Write a python program to display the histogram of an image and manipulate using histogram equalization.
5) Write a python program to perform the following basic image operations.
    a) Access pixel values and modify
    b) Access image properties
    c) Setting Region of Image(ROI)
    d) Splitting and Merging images
    e) Making image Borders
6) Write a python program to show the output following image arithmetic operations.
    a) Image Addition
    b) Image Subtraction
    c) Bit-wise operations
7) Write a python program to convert the images from one color space to another and plot the result.
8) Write a python program to apply the simple and adaptive thresholding on any input gray scale image and show the result.
9) Write a python program to apply different geometric transformations to images like scaling, rotation, translation, etc.
10) Write a python program implement following filters.
    a) Smoothening filters
    b) Sharpening filters
11) Write a python program to perform the following morphological operations.
    a) Erosion

    b) Dilation
    c) Opening
    d) Closing
12) Write a python program to apply any two image gradient filters and plot the input and output images.
13) Write a python program apply the canny edge detection algorithm.
14) Write a python program to perform the image segmentation using watershed algorithm.
15) Write a python program to calculate the any five features of different images.

Semester **III**		**Course Code** **21MCS36P**	**Title of the Paper** **DIGITAL IMAGE PROCESSING USING PYTHON LAB**		**Hours** **3**	**Credits** **2**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**	✓		✓		✓
	RESEARCH-RELATED SKILLS		✓		✓	
	PROBLEM SOLVING			✓		✓
	COMMUNICATION SKILLS	✓		✓		
	LEADERSHIP READINESS/QUALITIES	✓	✓			✓
	ANALYTICAL REASONING		✓	✓		
	MORAL AND ETHICAL AWARENESS	✓			✓	✓
	SCIENTIFIC REASONING	✓				
	CRITICAL THINKING				✓	✓

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	**Core Practical 6: MOBILE APPLICATIONS DEVELOPMENT LAB**	III	21MCS37P

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Describe the components and structure of mobile application development frameworks for Android practically.
2. Discriminate the capabilities and limitations of mobile devices.
3. Operate the tools for Android application project.
4. Use padding and margins to manage layout in device independent pixels (DIP).
5. Justify the use of fragments in newer versions of Android.
6. Reorganize and use Android Selection Widgets.

LIST OF PRACTICALS

1. Android SDK installation and study.
2. Defining Layouts.
3. Single Activity Application, Application with multiple activities, using intents to Launch Activities.
4. Application using GUI Widgets.
5. Application with Notifications.
6. Creating and Saving Shared Preferences and Retrieving Shared Preferences.
7. Usage of SQLite Databases for storage.
8. Case Study: Developing mobile applications for problems in the field like e-learning, banking, insurance, sales and health services.

Semester III		Course Code 21MCS37P	Title of the Paper MOBILE APPLICATIONS DEVELOPMENT LAB			Hours 2	Credits 2
Course Outcome(CO)		CO1	CO2	CO3	CO4	CO5	CO6
Programme Learning Outcome (PLO)	DISPLINARY KNOWLEDGE	✓		✓		✓	
	RESEARCH-RELATED SKILLS		✓				✓
	PROBLEM SOLVING			✓		✓	
	COMMUNICATION SKILLS	✓		✓			✓
	LEADERSHIP READINESS/QUALITIES		✓			✓	
	ANALYTICAL REASONING			✓		✓	
	MORAL AND ETHICAL AWARENESS	✓		✓			✓
	SCIENTIFIC REASONING			✓			
	CRITICAL THINKING		✓		✓	✓	

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	Open Source Technology (PHP/MySQL)	IV	21MCS41C

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Describe major elements of the PHP & MySQL work and justify why PHP is good for web development.
2. Describe how a static website be turned into a dynamic website to run from a database using PHP and MySQL.
3. Analyze the basic structure of a PHP web application and be able to install and maintain the web server, compile, and run a simple web application.
4. Design databases and use PHP MyAdmin to work with MySQL.
5. Propose the ways of connecting to MySQL through PHP, and how to manipulate tables thento Connect to SQL Server and other data sources.

UNIT - I

Basic Syntax, Defining variable and constant, Php Data type, Operator and Expression.- Decisions and loop Making Decisions, Doing Repetitive task with looping, Mixing Decisions and looping with Html.-. Function What is a function, Define a function, Call by value and Call by reference, Recursive function, String Creating and accessing, String Searching & Replacing String, Formatting String, String Related Library function.

UNIT-II

Array Anatomy of an Array, Creating index based and Associative array Accessing array, Element Looping with Index based array, Looping with associative array using each () and for each(), Some useful Library function-Handling Html Form with Php Capturing Form, Data Dealing with Multi-value filed, and Generating File uploaded form, redirecting a form after submission.

UNIT-III

Working with file and Directories Understanding file& directory, Opening and closing, a file, Coping, renaming and deleting a file, working with directories, Creating and deleting folder, File Uploading & Downloading- Session and Cookie Introduction to Session Control, Session Functionality.

UNIT-IV

What is a Cookie, Setting Cookies with PHP.Using Cookies with Sessions, Deleting Cookies, Registering Session variables, Destroying the variables and Session.

UNIT-V

Database Connectivity with MySql Introduction to RDBMS, Connection with MySql Database, Performing basic database operation(DML) (Insert, Delete, Update, Select), Setting query parameter, Executing query Join (Cross joins, Inner joins, Outer Joins, Self joins.) - Exception Handling Understanding Exception and error, Try, catch, throw. Error tracking and debugging.

TEXT BOOKS

1. W Jason Gilmore “**Beginning PHP and MySQl From Novice to professional**” 4th EditionApress,2010.

REFERENCES

1. Robin Nixon, “**Learning PHP, MySQL, and java Script: A Step-By-Step Guide to Creating Dynamic Websites**”o'reillyPress,2021.

FURTHER READING

1. Luke Welling and Laura Thmsan, **“ PHP and MySQL Web Development”** Fifth edition , Pearson Education,2016.

2. Lokesh Gupta , **“PHP and MySQL Web Development By Lokesh Gupta”** 1st edition, Notion Press,2020.

Semester **IV**		**Course Code** **21MCS41C**	**Title of the Paper** **Open Source Technology (PHP/MySQL)**		**Hours** **5**	**Credits** **3**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**		✓		✓	
	RESEARCH-RELATED SKILLS	✓		✓		✓
	PROBLEM SOLVING		✓		✓	
	COMMUNICATION SKILLS	✓				✓
	LEADERSHIP READINESS/QUALITIES			✓		
	ANALYTICAL REASONING			✓		✓
	MORAL AND ETHICAL AWARENESS	✓			✓	
	SCIENTIFIC REASONING			✓		✓
	CRITICAL THINKING	✓			✓	

Year	Subject Title	Sem.	Sub Code
2021-2022 **Onwards**	**Open Source Technology (PHP/MySQL)**	**VI**	**21MCS42P**

COURSELEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Design software using open source tools like PHP and MySql.
2. Create basic programs using PHP Concepts.
3. Apply In-built and create user defined function in PHP programming.
4. Design and develop a web site using form controls for presenting web based content.
5. Use Cookies to store and receive Data.
6. Create Session Variable and handle session.

List of Experiments:

1. Write a PHP program to validate the Textbox.
2. Write a PHP program to draw different shapes.
3. Write a PHP program to perform the string manipulation.
4. Write a PHP program using MySQL table.
5. Write a PHP program to perform user registration form using HTML tags.
6. Write a PHP program to display date and time.
7. Write a PHP program to check user login.
8. Write a PHP program to create a college website.
9. Write a PHP program for cookies and session scripts.
10. Write a PHP program to perform file read, write, open and append operation.
11. Write a PHP program to check student grade based on marks.
12. Write a PHP program to show day of the week using switch.
13. Write a PHP factorial program using for loop in php.
14. Write a PHP program to create Chess board in PHP using for loop.
15. Write a PHP program to calculate Electricity bill in PHP.
16. Write a simple calculator program in PHP using switch case.
17. Write a PHP program to check if a person is eligible to vote.
18. Write a PHP Program to Select, Update, Delete data from MySQL Data Base.

Semester VI		Course Code 21MCS42P	Title of the Paper Open Source Technology (PHP/MySQL)			Hours 5	Credits 2
Course Outcome(CO)		CO1	CO2	CO3	CO4	CO5	CO6
Programme Learning Outcome (PLO)	DISPLINARY KNOWLEDGE	✓	✓		✓	✓	
	RESEARCH-RELATED SKILLS		✓	✓			✓
	PROBLEM SOLVING	✓		✓	✓	✓	
	COMMUNICATION SKILLS		✓		✓		✓
	LEADERSHIP READINESS/QUALITIES	✓		✓		✓	✓
	ANALYTICAL REASONING		✓		✓	✓	
	MORAL AND ETHICAL AWARENESS	✓		✓		✓	✓
	SCIENTIFIC REASONING		✓		✓	✓	✓
	CRITICAL THINKING	✓	✓		✓		✓

ear	Subject Title	Sem.	Sub Code
2021 -22 Onwards	PROJECT VIVA VOCE	IV	21MCS43V

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Develop plans with relevant people to achieve the project's goals.
2. Break work down into tasks and determine handover procedures.
3. Identify links and dependencies, and schedule to achieve deliverables.
4. Estimate and cost the human and physical resources required, and make plans to obtain the necessary resources.
5. Allocate roles with clear lines of responsibility and accountability.

Total Marks: 100

Guidelines for Internal Assessment

1) Title and Abstract submission (10 Marks)

- The Abstract should have at least 2Pages.
- No Marks will be given if submitted after due date

2) Review:1 (25 Marks)

* The Student should submit the following to the Respective Guides during this Review meeting

  1) System Study /Analysis

  - Existing System
  - Proposed System
  - Feasibility Analysis

  2) System design (Which ever applicable to your Project)

  - Data Flow Diagram
  - System Flow Design
  - ERD
  - Table Design
  - Input Design
  - Modules Design

3) Review:2 (25 Marks)

* The Students should submit the following to the respective Guides without fail

  1) System Implementation

  - About the Software used
  - Implementation of modules

4) Review:3 **(25 Marks)**

* The Students should submit the following to the respective Guides without fail

1) System Testing

- Testing Strategies
- Test Plan
- Testing Report of your Software

5) Rough Project Report Submission for correction **(15Marks)**

6) Project Report Submission **Total Marks 100***

***<u>Note:</u>**

1. If the Student fails to attend Review Meeting, respective marks will not be awarded and treated as **"Absent"**.
2. The Student cannot submit project report, if they are absent for all Reviews.

Semester IV		**Course Code 21MCS43V**	**Title of the Paper PROJECT VIVA VOCE**		**Hours 20**	**Credits 10**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**	✓				✓
	RESEARCH-RELATED SKILLS		✓			✓
	PROBLEM SOLVING			✓		✓
	COMMUNICATION SKILLS	✓			✓	
	LEADERSHIP READINESS/QUALITIES		✓	✓		✓
	ANALYTICAL REASONING	✓			✓	
	MORAL AND ETHICAL AWARENESS	✓		✓		
	SCIENTIFIC REASONING	✓		✓		✓
	CRITICAL THINKING	✓		✓		

ELECTIVE: I

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	DISCRETE STRUCTURES AND OPTIMIZATION	II	21MCS25E

Course Level Outcomes

On successful completion of the course the students will be able to:

1. Construct mathematical arguments using logical connectives and quantifiers.
2. Verify the correctness of an argument using propositional and predicate logic and truth tables.
3. Demonstrate the ability to solve problems using counting techniques and combinatory in the context of discrete probability.
4. Use graphs and trees as tools to visualize and simplify situations.
5. Perform operations on discrete structures such as sets, functions, relations, and sequences.

UNIT-1:

Mathematical Logic: Propositional and Predicate Logic, Propositional Equivalences, Normal Forms, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference.

Sets and Relations: Set Operations, Representation and Properties of Relations, Equivalence Relations, Partially Ordering.

UNIT-1I:

Counting, Mathematical Induction and Discrete Probability: Basics of Counting, Pigeonhole Principle, Permutations and Combinations, Inclusion- Exclusion Principle, Mathematical Induction, Probability, Bayes' Theorem.

UNIT-1II:

Group Theory: Groups, Subgroups, Semi Groups, Product and Quotients of Algebraic Structures, Isomorphism, Homomorphism, Automorphism, Rings, Integral Domains, Fields, Applications of Group Theory.

Graph Theory: Simple Graph, Multigraphs, Weighted Graph, Paths and Circuits, Shortest Paths in Weighted Graphs, Eulerian Paths and Circuits, Hamiltonian Paths and Circuits, Planner graph, Graph Coloring, Bipartite Graphs, Trees and Rooted Trees, Prefix Codes, Tree Traversals, Spanning Trees andCut-Sets.

UNIT-1V:

Boolean Algebra: Boolean Functions and its Representation, Simplifications of Boolean Functions.

UNIT-V:

Optimization: Linear Programming - Mathematical Model, Graphical Solution, Simplex and Dual Simplex Method, Sensitive Analysis; Integer Programming, Transportation and Assignment Models, PERT-CPM: Diagram Representation, Critical Path Calculations, Resource Levelling, Cost Consideration in Project Scheduling.

TEXT BOOK

1. Richard Kohar“ **Basic Mathematics (Logic, Set Theory & Probability)”** World Scientific publisher ,2016 , (Unit-1).
2. Eric Gossett “**Discrete Mathematics with Proof**”2ndEdition,Wiley 2009. (Unit II & III).
3. J.Eldonwhitesitt “**Boolean Algebra & its Applications**”Dover Publishing ISBN: 9780486684833,(Unit IV).
4. Metei A. J., Veena Jain “**Optimization Using Linear Programming**” Mercury Learning & Information,ISBN:9781683923466, 16839234642019.,(Unit IV).

REFERENCES

1. V.kbalakrishnan “**Introductory Discrete Mathematics (Dover books on Computer Science)”** Dover Publicastion, New Edition, Feb 2000.

FURTHER READING

1. **https://nptel.ac.in/courses/106106094**

Semester II		Course Code 21MCS25E	Title of the Paper Discrete Structures and Optimization		Hours 5	Credits 5
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**		✓			✓
	RESEARCH-RELATED SKILLS	✓			✓	
	PROBLEM SOLVING	✓		✓		
	COMMUNICATION SKILLS		✓		✓	
	LEADERSHIP READINESS/QUALITIES			✓		
	ANALYTICAL REASONING	✓		✓		
	MORAL AND ETHICAL AWARENESS	✓		✓		✓
	SCIENTIFIC REASONING		✓			✓
	CRITICAL THINKING	✓		✓	✓	

	Subject Title	Sem.	Sub Code
2021 -22 Onwards	**MACHINE LEARNING TECHNIQUES**	**II**	**21MCS25E**

COURSE LEVEL OUTCOMES
On successful completion of the course the students will be able to:

1. Recall the basic concepts and techniques of Machine Learning.
2. Discriminate the Supervised and Unsupervised learning techniques.
3. Explain various probability based learning techniques.
4. Interpret graphical models of machine learning algorithms.
5. Distinguish between, supervised, unsupervised and semi-supervised learning.
6. Apply the apt machine learning strategy for any given problem.
7. Judge supervised, unsupervised or semi-supervised learning algorithms for any given problem.
8. Design systems that uses the appropriate graph models of machine learning.
9. Modify existing machine learning algorithms to improve classification efficiency.

UNIT I :
INTRODUCTION: Learning – Types of Machine Learning – Supervised Learning – The Brain and the Neuron – Design a Learning System – Perspectives and Issues in Machine Learning – Concept Learning Task – Concept Learning as Search – Finding a Maximally Specific Hypothesis – Version Spaces and the Candidate Elimination Algorithm – Linear Discriminates – Perceptron – Linear Separability – Linear Regression.
UNIT II
LINEAR MODELS: Multi-layer Perceptron – Going Forwards – Going Backwards: Back Propagation Error – Multi-layer Perceptron in Practice – Examples of using the MLP – Overview – Deriving Back-Propagation – Radial Basis Functions and Splines – Concepts – RBF Network – Curse of Dimensionality – Interpolations and Basis Functions – Support Vector Machines .
UNIT III
TREE AND PROBABILISTIC MODELS: Learning with Trees – Decision Trees – Constructing Decision Trees – Classification and Regression Trees – Ensemble Learning – Boosting – Bagging – Different ways to Combine Classifiers – Probability and Learning – Data into Probabilities – Basic Statistics – Gaussian Mixture Models – Nearest Neighbor Methods – Unsupervised Learning – K means Algorithms – Vector Quantization – Self Organizing Feature Map.
UNIT IV
DIMENSIONALITY REDUCTION AND EVOLUTIONARY MODELS: - Dimensionality Reduction – Linear Discriminant Analysis – Principal Component Analysis – Factor Analysis – Independent Component Analysis – Locally Linear Embedding – Isomap – Least Squares Optimization – Evolutionary Learning – Genetic algorithms – Genetic Offspring: - Genetic Operators – Using Genetic Algorithms – Reinforcement Learning – Overview – Getting Lost Example – Markov Decision Process.
UNIT V
GRAPHICAL MODELS: 9 Markov Chain Monte Carlo Methods – Sampling – Proposal Distribution – Markov Chain Monte Carlo – Graphical Models – Bayesian Networks – Markov Random Fields – Hidden Markov Models – Tracking Methods.

TEXT BOOKS

1. Stephen Marsland, “**Machine Learning – An Algorithmic Perspective”**, Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
2. Tom M Mitchell, “**Machine Learning”**, First Edition, McGraw Hill Education, 2013.

REFERENCES

1. Peter Flach, “**Machine Learning: The Art and Science of Algorithms that Make Sense of Data”**, FirstEdition, Cambridge University Press, 2012.
2. Jason Bell, “**Machine learning – Hands on for Developers and Technical Professionals”**, First Edition,Wiley, 2014 .

FURTHER READING

1. EthemAlpaydin, “**Introduction to Machine Learning 3e (Adaptive Computation and Machine Learning Series)**, Third Edition, MIT Press, 2014.
2. **https://nptel.ac.in/courses/106106139**

Semester **II**		**Course Code** **21MCS24C**		**Title of the Paper** **MACHINE LEARNING TECHNIQUES**					**Hours** **5**	**Credits** **5**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**	**CO7**	**CO8**	**CO9**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**	✓		✓		✓		✓		
	RESEARCH-RELATED SKILLS		✓		✓			✓		✓
	PROBLEM SOLVING	✓		✓			✓		✓	
	COMMUNICATION SKILLS	✓				✓			✓	
	LEADERSHIP READINESS/QUALITIES	✓		✓		✓		✓		✓
	ANALYTICAL REASONING			✓			✓			✓
	MORAL AND ETHICAL AWARENESS	✓			✓		✓		✓	
	SCIENTIFIC REASONING		✓				✓		✓	
	CRITICAL THINKING	✓		✓		✓	✓			✓

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	**Artificial Intelligence (AI)**	II	21MCS25E

COURSE LEVEL OUTCOME

On successful completion of the course the students will be able to:

1.Describe searching techniques, constraint satisfaction problem and game playing techniques.
2. Apply these techniques in applications which involve perception, reasoning and learning.
3. Explain the role of agents and how it is related to environment and the way of evaluating it and how agents can act by establishing goals.
4. Examine real world Knowledge representation.
5. Analyze and design a real world problem for implementation and understand the dynamic behavior of a system.
6. Use different Parsing techniques to design AI machine and enveloping applications for real world problems.

UNIT: I

Approaches to AI: Turing Test and Rational Agent Approaches; State Space Representation of Problems, Heuristic Search Techniques, Game Playing, Min-Max Search, Alpha Beta Cutoff Procedures.

Knowledge Representation: Logic, Semantic Networks, Frames, Rules, Scripts, Conceptual Dependency and Ontologies; Expert Systems, Handling Uncertainty in Knowledge.

UNIT: II

Planning: Components of a Planning System, Linear and Non Linear Planning; Goal Stack Planning, Hierarchical Planning, STRIPS, Partial Order Planning.

Natural Language Processing: Grammar and Language; Parsing Techniques, Semantic Analysis and Pragmatics.

UNIT: III

Multi Agent Systems: Agents and Objects; Agents and Expert Systems; Generic Structure of Multiagent System, Semantic Web, Agent Communication, Knowledge Sharing using Ontologies, Agent Development Tools.

UNIT: IV

Fuzzy Sets: Notion of Fuzziness, Membership Functions, Fuzzification and Defuzzification; Operations on Fuzzy Sets, Fuzzy Functions and Linguistic Variables; Fuzzy Relations, Fuzzy Rules and Fuzzy Inference; Fuzzy Control System and Fuzzy Rule Based Systems.

UNIT:V

Genetic Algorithms (GA): Encoding Strategies, Genetic Operators, Fitness Functions and GA Cycle; Problem Solving using GA.

Artificial Neural Networks (ANN): Supervised, Unsupervised and Reinforcement Learning; Single Perceptron, Multi Layer Perceptron, Self Organizing Maps, Hopfield Network

TEXT BOOK:

1. David L. Poole and Alan K. Mackworth "**Artificial Intelligence foundation of computational Agents**", 2ndEdition,published by Cambridge University Press,2010.
2. Michael Wooldridge"**An Introduction to MultiAgent Systems**", 2nd Edition,Wiley Publication, June 2009.
3. Guanrongchen, Trung Tat Pham,"**Introdction to Fuzzy Sets , Fuzzy Logicand Fuzzy Control Systems**",1st Edition, CRC Press publication ,2000.
4. K.F Man, K.S Tang, S. Kwong" **Genetic Algorithms Concepts & Design**" Springer London , 2012.

REFERENCES

1. Jon Krohn, Grant Beyaleveld and aglaebassens , **" Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence",** First edition ,Pearson Education, 2020.
2.

FURTHER READING

1. Russell, **"Artificial Intelligence**", 3rd edition, Pearson Education India, 2015.
2. **https://nptel.ac.in/courses/106105077**

Semester II		Course Code 21MCS25E	Title of the Paper Artificial Intelligence (AI)			Hours 5	Credits 5
Course Outcome(CO)		CO1	CO2	CO3	CO4	CO5	CO6
Programme Learning Outcome (PLO)	DISPLINARY KNOWLEDGE	✓			✓		✓
	RESEARCH-RELATED SKILLS	✓		✓		✓	
	PROBLEM SOLVING		✓		✓		
	COMMUNICATION SKILLS		✓				✓
	LEADERSHIP READINESS/QUALITIES	✓		✓			
	ANALYTICAL REASONING		✓		✓		✓
	MORAL AND ETHICAL AWARENESS	✓		✓		✓	
	SCIENTIFIC REASONING		✓				✓
	CRITICAL THINKING	✓		✓			

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	**Theory of Computation and Compilers**	II	21MCS25E

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Explain the concepts and different phases of compilation with compile time error handling.
2. Interpret language tokens using regular expressions, context free grammar and finite automata and design lexical analyzer for a language.
3. Develop the parsers and experiment the knowledge of different parsers design without automated tools.
4. Construct the intermediate code representations and generation.
5. Convert source code for a novel language into machine code for a novel computer.

UNIT: I

Theory of Computation: Structure of a Compiler - Deterministic Finite Automaton (DFA), Non-Deterministic Finite Automaton (NDFA), Equivalence of DFA and NDFA, Regular Languages, Regular Grammars, Regular Expressions, Regular expressions to Finite automata- – NFA to DFA – Minimising the number of states of a Deterministic Finite Automata - Lexical Analysis.

UNIT – II

Context Free Language: Pushdown Automaton (PDA), Context Free Grammar, Chomsky Normal Form, Greibach Normal Form – Derivations and Parse trees – Ambiguity – Capabilities of Context free grammars – Top down and Bottom up parsing – Handles – Shift Reduce parsing – Operator precedence parsing – Recursive descent parsing.

UNIT – III

Automatic parsing techniques – LR parsers – Canonical collection of LR(0) items – Construction of SLR parsing table – LR(1) sets of items Construction – Construction of canonical LR Parsing Tables

UNIT – IV

Syntax Direction Translation – Semantic actions - Inherited and Synthesized Attributes – Implementation of Syntax Directed Translators – Intermediate Code: Postfix notation, Quadruples, Triples, Indirect triples – Methods of translation of Assignment statements, Boolean expressions.

UNIT – V

Symbol tables and Code generation: Representing information in a Symbol Table – Data Structures for Symbol table – Introduction to Code Optimization: Local Optimization, Global Optimization, Loop Optimization, Peep-Hole Optimization - Basic blocks – Introduction to Code Generation.

TEXT BOOK:

1. Alfred v.Aho, Monica S. Lam, Ravi Sethi, Jeffrey D.Ullman " **Compiler: Principles, Techniques and Tools"**, 2ndEdition,Pearson Education,2007.

REFERENCES

1. John E.Hopcroft,Rajeev Motwani,Jeffrey D.Ullman , **"Introduction to Automata Theory, Languages and Computation",** 2nd Edition,Pearson Education,2002.
2. Dhamdhere D. M., Compilers construction Principles and Practice, Macmillan India Ltd

FURTHER READINGS

1. **https://nptel.ac.in/courses/106104072**
2. **https://nptel.ac.in/courses/106108113**
3. **https://nptel.ac.in/courses/106108052**

Semester II		Course Code 21MCS25E	Title of the Paper Theory of Computation and Compilers				Hours 5	Credits 5
Course Outcome(CO)		CO1	CO2	CO3	CO4	CO5	CO6	CO7
Programme Learning Outcome (PLO)	DISPLINARY KNOWLEDGE		✓			✓		✓
	RESEARCH-RELATED SKILLS	✓			✓		✓	
	PROBLEM SOLVING	✓		✓		✓		✓
	COMMUNICATION SKILLS		✓				✓	
	LEADERSHIP READINESS/QUALITIES	✓		✓			✓	✓
	ANALYTICAL REASONING			✓	✓		✓	
	MORAL AND ETHICAL AWARENESS	✓					✓	✓
	SCIENTIFIC REASONING		✓		✓			✓
	CRITICAL THINKING			✓	✓		✓	

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	PROGRAMMING LANGUAGES AND COMPUTER GRAPHICS	II	21MCS25E

COURSE LEVEL OUTCOMES
On successful completion of the course the students will be able to:

1. Develop programs using the fundamentals and basics of C Language.
2. Create programs using the basic elements like control statements, Arrays and Strings.
3. Describe the procedural and object oriented paradigm with concepts of streams, classes, functions, data and objects.
4. Discuss the concept of function overloading, operator overloading, virtual functions and polymorphism.
5. Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming.
6. Justify the fundamental concept of , HTML,XML, JavaScript, to developing professional Programs.
7. Interpret the concepts of different type of geometric transformation of objects in 2D and 3D.

UNIT: I

Language Design and Translation Issues: Programming Language Concepts, Paradigms and Models, Programming Environments, Virtual Computers and Binding Times, Programming Language Syntax, Stages in Translation, Formal Transition Models.
Elementary Data Types: Properties of Types and Objects; Scalar and Composite Data Types.

UNIT: II

Programming in C: Tokens, Identifiers, Data Types, Sequence Control, Subprogram Control, Arrays, Structures, Union, String, Pointers, Functions, File Handling, Command Line Arguments,Preprocessors.

Object Oriented Programming: Class, Object, Instantiation, Inheritance, Encapsulation, Abstract Class, Polymorphism.

UNIT: III

Programming in C++: Tokens, Identifiers, Variables and Constants; Data types, Operators, Control statements, Functions Parameter Passing, Virtual Functions, Class and Objects; Constructors and Destructors; Overloading, Inheritance, Templates, Exception and Event Handling; Streams and Files; Multifile Programs.
Web Programming: HTML, DHTML, XML, Scripting, Java, Servlets, Applets.

UNIT: IV

Computer Graphics: Video-Display Devices, Raster-Scan and Random-Scan Systems; Graphics Monitors, Input Devices, Points and Lines; Line Drawing Algorithms, Mid-Point Circle and Ellipse Algorithms; Scan Line Polygon Fill Algorithm, Boundary-Fill and Flood-Fill.

2-D Geometrical Transforms and Viewing: Translation, Scaling, Rotation, Reflection and Shear Transformations; Matrix Representations and Homogeneous Coordinates; Composite Transforms, Transformations Between Coordinate Systems, Viewing Pipeline, Viewing Coordinate Reference Frame, Window to View-Port Coordinate Transformation, Viewing Functions, Line and Polygon Clipping Algorithms.

UNIT: V

3-D Object Representation, Geometric Transformations and Viewing: Polygon Surfaces, Quadric Surfaces, Spline Representation, Bezier and B-Spline Curves; Bezier and B-Spline Surfaces; Illumination Models, Polygon Rendering Methods, Viewing pipeline and coordinates; general projection transforms and clipping.

TEXT BOOK:

1. E**.**Balagurusamy,"**Programming in Ansi C**" , 7th Edition , Tata McGrawHill Publication,2016.
2. E**.**Balagurusamy,"**object Oriented Programming in C++**" , 7th Edition, Tata McGrawHill Publication, 2017.
3. Thomas A. Powell, "**HTML & CSS: The Complete Reference**", Fifth Edition, Tata McGraw Hill,2010.
4. Thomas A Powell, Fritz Schneider, "**JavaScript: The Complete Reference**",Third Edition, Tata McGraw Hill,2013.
5. D. Hearn and M.P. Baker, "**Computer Graphics**", 2ndEdition, Pearson Education, Prentice Hall,2005.

REFERENCES

1. S. Harrington, "**Computer Graphics"** , 2nd Edition , Tata McGraw-Hill Book Co,1987.

FURTHER READING

1. Foley, Van Dam, Feiner and Hughes "**Computer Graphics Principles & practice**",3rdEdition , Pearson Education,2013.
2. **https://nptel.ac.in/courses/106102067**
3. **https://nptel.ac.in/courses/106105171**
4. **https://nptel.ac.in/courses/106105151**
5. **https://nptel.ac.in/courses/106102063**

Semester II		Course Code 21MCS25E	Title of the Paper Programming Languages and Computer Graphics				Hours 5	Credits 5
Course Outcome(CO)		CO1	CO2	CO3	CO4	CO5	CO6	CO7
Programme Learning Outcome (PLO)	DISPLINARY KNOWLEDGE		✓			✓		✓
	RESEARCH-RELATED SKILLS	✓			✓		✓	
	PROBLEM SOLVING		✓			✓		✓
	COMMUNICATION SKILLS	✓		✓		✓	✓	
	LEADERSHIP READINESS/QUALITIES		✓		✓			✓
	ANALYTICAL REASONING	✓			✓	✓		
	MORAL AND ETHICAL AWARENESS		✓				✓	✓
	SCIENTIFIC REASONING	✓		✓			✓	
	CRITICAL THINKING		✓	✓		✓		

ELECTIVE-II

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	COMPUTER OPTIMIZATION TECHNIQUES	III	21MCS35E

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Describe the mathematical concepts and numerical methods required for Information Technology.
2. Solve real life problems in Business and Management.
3. Apply the theory of optimization methods and algorithms to develop and for solving various types of optimization problems.
4. Apply optimization techniques in problems of Engineering and Technology.
5. Solve the mathematical results and numerical techniques of optimization theory to concrete Engineering problems by using computer software.

UNIT I

OPERATION RESEARCH: Basics of OR – OR & Decision Making -Linear Programming- Mathematical Formulation - Graphical Solution – Canonical & Standard Forms of LPP.

UNIT II

SIMPLEX METHOD: Simplex Method for <, =, > constraints – Charne's Method of Penalties– Transportation Problem and its Solution.

UNIT III

MATHEMATICAL LOGIC: Propositions - Precedence Rules for Operators - Tautologies- Laws of Equivalence –Substitution Rules - Natural Deduction System - Developing Natural Deduction System Proofs.

UNIT IV

INTERPOLATION: Lagrange's and Newton Interpolation - Interpolating Polynomials using Finite Difference **NUMERICAL INTEGRATION:** Trapezoidal, Simpson's rules and Romberg Integration.

UNIT V

NUMERICAL METHODS: Polynomial Equation: Brige - Vieta, Graeffe's Root Squaring Methods. **INTEGRATION:** Gauss Lagrange – Gauss Chebyshev- Gauss Hermite Methods.

TEXT BOOKS

1) Manmohan & Gupta, "**Operations Research**", Sultan Chand Publishers, New Delhi, 2002. (UNIT I).

2) KantiSwarup-Gupta-ManMohan, “**Operations Research**” Seventh edition, (UNIT II).

3) David Gries, “**The Science of Programming**”, Narosa Publishers, House, New Delhi, Chapters1, 2, 3.1 to 3.3 (UNIT III).

4) M.K.Jain, S.R.K. Lyengar, R.K.Jain,“**Numerical Methods for Scientific and EngineeringComputation**”, 3rd ED., New Age Publication. New Delhi, Chapters: 2.8, 5.8 (UNITS IV &V).

REFERENCES

1. Shubham Agarwal, **“Computer Based Optimization Techniques”**, Alpha science International Ltd,2015.

FURTHER READING

1. Christos H. Papadimitriou, **“ Combinatorial Optimization: Algorithms and Complexity (Dover Books on Computer Science)”,**New edition, Dover Publications Inc., 2000.
2. **https://nptel.ac.in/courses/112101298**

Semester **III**		**Course Code** **21MCS35E**	**Title of the Paper** **COMPUTER OPTIMIZATION TECHNIQUES**		**Hours** **5**	**Credits** **5**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**		✓	✓		
	RESEARCH-RELATED SKILLS	✓			✓	✓
	PROBLEM SOLVING		✓			
	COMMUNICATION SKILLS	✓		✓		✓
	LEADERSHIP READINESS/QUALITIES		✓		✓	
	ANALYTICAL REASONING	✓		✓		
	MORAL AND ETHICAL AWARENESS	✓				✓
	SCIENTIFIC REASONING		✓		✓	
	CRITICAL THINKING	✓		✓		✓

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	**Reactive Programming**	III	21MCS35E

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Discuss Reactive programming to handle applications events and data flow.
2. Recognize reactive programming to design the components and other pieces of software in order to react to those changes instead of asking for changes.
3. Discuss Angular expose observables.
4. Use RxJS with Angular Forms.
5. Explain how to chain common RxJS operators together.
6. Interpret the requirement of observables in Angular and how to implement the same solution without an observable chain.

UNIT – I

Getting Started with the Echoes Player Lite App: Browser and Development Environment: Augury Dev Tools Extension- Version Control and Deployment- Setting the Development Environment-Terminal/Command Line- Editor (IDE or Code Editor)- Application Structure- **RxJS and ngrx**: Reactive Extensions for Angular: RxJS- ngrx: Reactive Extensions for Angular: ngrx/core- ngrx/store- ngrx/effects-ngrx/store-log-monitor-Case Study: Sample Application.

UNIT - II

Getting Familiar with Boilerplate for Development: Boilerplate Overview-The “config” directory-Customizing Karma with Configuration-Karma-Supported Frameworks-Karma’s Test Process Modes-Webpack-Third-Party Libraries- The App Directory.

UNIT – III Adding State Management with ngrx/store: **Benefits of Store as State Management- Redux Primer- ngrx/store- Adding ngrx/store.**Creating Reactive Components: Presentational and Container: **Introduction to HTML Components- Container / Smart Component- Presentational / Dumb Component-grx/store and Change-Detection Strategy- Creating now-playlist-filter Presentational Component- View Components with Augury.**

UNIT – IV

Understanding Services with ReactiveProgramming:Diving into the Search Service- Adding a Player- BDD for Services**- Managing Side Effects with ngrx/effects:**Introduction to ngrx/effects- Constructing Side Effects to Actions- Retrieving Data from Store inside an Effect- Testing Effects.

UNIT – V

Reactive Forms and Common Solutions: Looking Closely at Reactive Forms: Switching to Reactive Forms- Creating a Reactive Search Form- Using Form Builder to Create a Reactive Form- Enhancing the Form with New Features: Adding Presets Form Control- React to Form Changes- Updating Side Effects- Trigger Search When Preset Is Changed - Common Solutions.

TEXT BOOKS

1. Oren Farhi “**Reactive Programming with Angular and ngrx: Learn to Harness the Power of Reactive Programming with RxJS and ngrxExtensions**”, Apress,2017.

REFERENCES

1. Shama Hoque , **“Full-Stack React Projects: Learn MERN stack development by building modern web apps using MongoDB, Express, React, and Node.js”** , 2nd edition ,Packt Publishing Limited,2020.

FURTHER READINGS

1. Mehul Mohan , **“ Advanced Web Development With React, Concepts: SSR and PWA with Next.js using React with advanced concepts” ,** First edition, BPB Publications, 2020.

Semester **III**		**Course Code** **21MCS35E**	**Title of the Paper** Reactive Programming			**Hours** **5**	**Credits** **5**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**		✓		✓		✓
	RESEARCH-RELATED SKILLS	✓			✓	✓	
	PROBLEM SOLVING	✓		✓	✓		✓
	COMMUNICATION SKILLS		✓			✓	
	LEADERSHIP READINESS/QUALITIES	✓		✓			✓
	ANALYTICAL REASONING		✓		✓		
	MORAL AND ETHICAL AWARENESS	✓		✓			
	SCIENTIFIC REASONING		✓			✓	
	CRITICAL THINKING	✓		✓		✓	✓

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	ENTERPRISE RESOURCE PLANNING	III	21MCS35E

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Describe the need for ERP in business.
2. Discuss the role of information technology in business.
3. Use Enterprise software and infer its role in integrating business functions.
4. Analyze the strategic options for ERP identification and adoption.
5. Design the ERP implementation strategies.
6. Create reengineered business processes for successful ERP implementation.

UNIT - I

Introduction to ERP: Introduction – Evolution of ERP – What Is ERP – Reasons for The Growth of The ERP Market – The Advantages of ERP – Why do Many ERP Implementations Fail – Why are ERP Packages Being used now. Enterprise – An Overview: Introduction – Integrated Management Information – Business Modeling – Integrated Data Model.

UNIT - II

ERP and Related Technologies: Introduction – Business Process Reengineering – Management Information System – Decision Support System – Executive Information Systems – Data Warehousing – Data Mining – Online Analytical Processing – Supply Chain Management. ERP A Manufacturing Perspective: Introduction – ERP CAD / CAM – Materials Requirement Planning – Bill of Material – Closed Loop – Manufacturing Resource Planning – Distribution Requirements Planning – JIT and Kanban Product Data Management – Data Management – Benefits of PDM – Make to Order and Make to Stock – Assemble to Order – Engineer to Order – Configure to Order.

UNIT - III

ERP Modules: Introduction – Finance – Plant Maintenance – Quality Maintenance – Material Management. Benefits of ERP: Introduction – Reduction of Lead Time – On Time Shipment – Reduction in Cycle Time – Improved Resource Utilization – Better Customer Satisfaction – Improved Supplier Performance – Increased Flexibility – Reduced Quality Costs – Improved Information Accuracy and Decision-Making Capability.

UNIT - IV

ERP Market: Introduction – SAP AG – Baan Company – Oracle Corporation – People Soft – JD Edwards World Solutions Company – System Software Associate QAD. ERP Implementation Life Cycle: Introduction – Pre evaluation Screening – Package Evaluation – Project Planning Phase – Gap Analysis – Reengineering Configuration – Implementation Team Training – Testing-Going Live – End User Training – Post Implementation.

UNIT – V

Vendors, Consultants and Users: Introduction – In House Implementation – Pros and Cons –

Vendors – Consultants – End Users. Future Directions In ERP: Introduction – New Markets – New Channels – Faster Implementation Methodologies – Business Models and Bap is Convergence on Windows NT – Application Platforms – New Business Segments – More Features – Web Enabling – Market Snapshot.

TEXT BOOKS

1. Alexis Leon "**Enterprise Resource Planning**",4th Edition, Tata McGraw Hill,2020.

REFERENCE BOOKS

1. Alexis Leon, **"Enterprise Resource Planning",** Fourth edition, Tata McGraw Hill, 2019.

FURTHER RTEADING

1. Ganesh L , **" Enterprise Resource Planning"**, discovery publishing, 2005.

Semester **III**		**Course Code** **21MCS35E**	**Title of the Paper** **ENTERPRISE RESOURCE PLANNING**			**Hours** **5**	**Credits** **5**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**		✓		✓		✓
	RESEARCH-RELATED SKILLS	✓			✓	✓	
	PROBLEM SOLVING	✓		✓	✓		✓
	COMMUNICATION SKILLS		✓			✓	
	LEADERSHIP READINESS/QUALITIES	✓		✓			✓
	ANALYTICAL REASONING		✓		✓		
	MORAL AND ETHICAL AWARENESS	✓		✓			
	SCIENTIFIC REASONING		✓			✓	
	CRITICAL THINKING	✓		✓		✓	✓

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	**EMBEDDED SYSTEMS**	III	21MCS35E

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Describe the concepts of Architecture, designing of Embedded Systems and it stools.
2. Infer microcontroller, different components of microcontroller and their interactions.
3. Compare the embedded systems such as I/O, timers, interrupts and interaction with peripheral devices.
4. Develop the Instruction execution stages and different types of addressing modes and Memory Organization.
5. Write the programs for microcontroller.
6. Judge the role and Design Concept of embedded systems in industry.

UNIT – I

8051 Microcontroller: Introduction – 8051 Architecture – Input / Output Pins, Ports and Circuits - External Memory – Counters / Timers – Serial Data Input / Output – Interrupts.

UNIT – II

Instruction Set and Programming Moving Data – Addressing Modes – Logical operations –Arithmetic Operation – Jump and Call Instructions – Simple Program. Applications: Keyboard Interface – Display Interface – Pulse Measurements – DIA and AID Conversions – Multiple Interrupts.

UNIT – III

Concepts on RTOS: Introduction to RTOS – Selecting an RTOS – Task and Task states – Tasks and data – Semaphores and shared data. MORE operating systems services: Interrupt Process communication – Message Queues, Mailboxes and pipes – Timer Functions – Events – Memory Management – Interrupt Routines in an RTOS Environment.

UNIT – IV

Basic Design using a RTOS: Principles – Encapsulating semaphores and Queues – Hard real time scheduling considerations – Saving memory space and power – introductions to RTL & QNX.

UNIT – V

Software Tools: Embedded Software Development Tools: Hosts and Target Machines – Linker/Locators for Embedded software-getting Embedded software into the Target systems. Debugging Techniques: Testing on your Host machine – Instruction set simulators – The assert macro – using laboratory tools.

TEXT BOOKS

1. David.E.Simon, "**An Embedded Software Primer**",8th Edition, Pearson Education,2009.
2. Kenneth J Ayala, PenramInternational , "**The 8051 Microcontroller and Architecture Programming and Application**" 2ndEdition,Delmar Cengage Learning Publication,1991.

REFERENCES

1. Raj Kamal, **"Embedded Systems, Architecture, Programming and Design"** 3rd Edition, Tata McGraw Hill Education, 2017.

FURTHER READINGS

1. **https://nptel.ac.in/courses/108105057**

Semester III		**Course Code 21MCS35E**	**Title of the Paper** EMBEDDED SYSTEMS			**Hours 5**	**Credits 5**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**	✓		✓			✓
	RESEARCH-RELATED SKILLS	✓			✓		
	PROBLEM SOLVING		✓	✓		✓	
	COMMUNICATION SKILLS	✓			✓		✓
	LEADERSHIP READINESS/QUALITIES		✓			✓	
	ANALYTICAL REASONING	✓			✓		✓
	MORAL AND ETHICAL AWARENESS			✓		✓	
	SCIENTIFIC REASONING		✓	✓		✓	
	CRITICAL THINKING	✓				✓	

Year	Subject Title	Sem.	Sub Code
2021 -22 Onwards	IoT ARCHITECTURE AND PROTOCOLS	III	21MCS35E

COURSE LEVEL OUTCOMES

On successful completion of the course the students will be able to:

1. Describe the Internet of Things and its Architecture.
2. Recall the IOT protocols.
3. Discuss the architecture, operation, and business benefits of an IoT solution.
4. Examine the potential business opportunities that IoT can uncover.
5. Justify the relationship between IoT, cloud computing, and big data.
6. Identify how IoT differs from traditional data collection systems.
7. Design an IoT device to work with a Cloud Computing infrastructure.

UNIT I – OVERVIEW

IoT-An Architectural Overview– Building an architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations. M2M and IoT Technology Fundamentals- Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, Everything as a Service (XaaS), M2M and IoT Analytics, Knowledge Management.

UNIT II – REFERENCE ARCHITECTURE:

IoT Architecture-State of the Art – Introduction, State of the art, Reference Model and architecture, IoT reference Model - IoT Reference Architecture-Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views. Real-World Design Constraints-Introduction, Technical Design constraints-hardware is popular again, Data representation and visualization, Interaction and remote control.

UNIT III – IOT DATA LINK LAYER & NETWORK LAYER PROTOCOLS:

PHY/MAC Layer (3GPP MTC, IEEE 802.11, IEEE 802.15), Wireless HART, Z-Wave, Bluetooth Low Energy, Zigbee Smart Energy, DASH7 - Network Layer-IPv4, IPv6, 6LoWPAN, 6TiSCH, ND, DHCP, ICMP, RPL, CORPL, CARP.

UNIT IV – TRANSPORT & SESSION LAYER PROTOCOLS:

Transport Layer (TCP, MPTCP, UDP, DCCP, SCTP) - (TLS, DTLS) – Session Layer-HTTP, CoAP, XMPP, AMQP, MQTT.

UNIT V – SERVICE LAYER PROTOCOLS & SECURITY:

Service Layer -oneM2M, ETSI M2M, OMA, BBF – Security in IoT Protocols – MAC 802.15.4, 6LoWPAN, RPL, Application Layer.

TEXT BOOK:

1. Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, StamatisKarnouskos, David Boyle, "**From Machine-to-Machine to**

the Internet of Things: Introduction to a New Age of Intelligence", 1 st Edition, Academic Press,2014.
2. Peter Waher, "**Learning Internet of Things**", PACKT publishing, 2015.

REFERENCES:

1. Bernd Scholz-Reiter, Florian Michahelles, **"Architecting the Internet of Things**", ISBN 978- 3-642-19156-5 e-ISBN 978-3-642-19157-2,Springer.
2. Daniel Minoli, "**Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications**", ISBN: 978-1-118- 47347-4, WileyPublications,2013.
3. Vijay Madisetti and Arshdeep Bahga, "**Internet of Things (A Hands-onApproach)**", 1 st Edition, VPT, 2014.

FURTHER READING

1. Raj Kamal , **"Internet of Things : Architecture and Design Principles"** , First edition, Tata McGraw Hill Education, 2017.
2. https://nptel.ac.in/courses/106105166

Semester **III**		**Course Code** **21MCS35E**	**Title of the Paper** **IoT ARCHITECTURE AND PROTOCOLS**				**Hours** **5**	**Credits** **5**
Course Outcome(CO)		**CO1**	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**	**CO7**
Programme Learning Outcome (PLO)	**DISPLINARY KNOWLEDGE**	✓		✓				✓
	RESEARCH-RELATED SKILLS	✓		✓		✓		✓
	PROBLEM SOLVING		✓			✓	✓	
	COMMUNICATION SKILLS	✓		✓		✓		
	LEADERSHIP READINESS/QUALITIES	✓			✓			✓
	ANALYTICAL REASONING		✓			✓		
	MORAL AND ETHICAL AWARENESS	✓			✓			✓
	SCIENTIFIC REASONING		✓	✓		✓	✓	
	CRITICAL THINKING		✓		✓	✓		✓

7. Teaching-Learning Methodologies

The teaching-learning process should be in-line with the course objective and outcomes. Teaching has to ensure that the suggested outcomes are ensured for each course and overall programme. Teaching-aids should be used wherever required to facilitate proper and impactful learning. Blended learning is recommended with the use of MOOC platforms and classroom teaching.

To meet the set objectives of the course and enable students achieve the expected outcomes of the course the teaching-learning process should be appropriately chosen. Though the teachers are best positioned to create innovative models suitable for teaching the course, certain well accepted and widely tested processes are suggested to achieve the desired outcomes

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 should be made mandatory and 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 should be arranged to help the students understand the practices in the industry and developments in the field.

MOOCS - Teacher should choose 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.

PROJECT - Wherever possible the laboratory assignments can be designed in the form of a mini project. For example, the database course lab assignments can be designed to build a complete system for library management. Similarly, summer/ Semester breaks can be utilized for guiding students to develop live projects with industry orientation/ industry problem. Teamwork work

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

(2) MAJOR PROJECT - The major project should be defined based on the student proposals keeping in mind that opportunity to demonstrate the knowledge and skills gained during the course. One-One mentoring support should be provided.

(3) Simulation - Packages to provide simulated environments to teach various components of networking and hardware working should be used wherever feasible.

8. Assessment Methods

The committee recommends that assessment should be viewed not only merely as a testing by the institution to evaluate the students' progress, but also as a valuable tool for a student to learn what is expected of him/her, where their level of knowledge and skill is lacking, and perhaps most importantly, what he/she could do to improve these levels with the valuable inputs of the lecturers. Assessment methods are the strategies, techniques, tools and instruments for collecting information to determine the extent to which students demonstrate desired learning outcomes. In the Master's programmes leading to degrees such as BSc with Computer Science and MSc in Computer Science, the assessment and evaluation methods focus on testing the conceptual understanding of the basic ideas of computer hardware and software, development of programming skills and experimental techniques, retention and ability to apply the knowledge acquired to real -life applications, and to solve new problems and communicate the results and findings effectively. Based on the Learning Objectives defined for each course as proposed in detail, assessment methods can be designed to monitor the progress in achieving the Learning Objectives during the course and test the level of achievement at the end of the course. Several methods can be used to assess student learning outcomes. Relying on only one method to provide information about the program will only reflect a part of students' achievement.

Modular Assessment

As the courses are broken up into a smaller more cohesive learning outcomes a module will consist of a number of these smaller, finer grained assessments of which the majority can be considered to be formative assessments that aid the learning process rather than assessments aimed at solely being used to evaluate the student.

Continuous Assessment

The continuous assessment occurs on a regular and continuous basis, it is an ongoing formative and summative process, involves the monitoring of students, is integrated with teaching, involves a systematic collection of marks or grades into a final score, may be used to determine the students' final grades.

Direct methods of assessment ask students to demonstrate their learning while indirect methods ask students to reflect on their learning. Tests, essays, presentations, etc. are generally direct methods of assessment, and indirect methods include surveys and interviews. For each Learning Objective, a combination of direct and indirect assessment methods should be used.

Formative Assessment

While *formative assessment* is to gather feedback from formal or informal processes that can be used by the instructor and the students to gather evidence for the purpose of improving learning, *summative assessment* measures the level of success or proficiency that has been obtained at the end of an instructional unit, by comparing it against some standard or benchmark. Nevertheless, the outcome of a *summative assessment* can be used formatively when students or faculty use the results to guide their efforts and activities in subsequent courses. Daily programming assignments or home-assignments is a good way of implementing *formative assessment* and gives an idea of how well the students understood and could apply each programming concept. Another way of *formative assessment* can be that at the end of each class period, a student response system can be used to ask students one or more questions about the topic taught on that day. Regular tutorial Assignment, Term -paper, Seminar Presentation, Surprise Quizzes, Open-book Quizzes should be adopted for formative assessments. It is suggested that 25-30% weightage be given *Formative Assessments* in case of theory components while 30-40% weightage be given to the Programming/Laboratory/Projects/Dissertation components of the various courses.

During the semester, at least three smaller formative assessments shall be given for each course. To pass a course a student had to achieve marks between 70% in two of the assessment opportunities. The philosophy is that the student could fail one opportunity and take the experience gained from that opportunity to pass subsequent assessments.

Summative Assessment

For the traditional summative assessment, it is the semester tests based. The students need to attend two semester tests which consist of half of the content they learned for each test. Students are admitted to an examination for individual courses if they attain the minimum semester mark of 40%. Summative Assessment for the theory papers, can be a combination of Mid -Semester Test, Individual /Team Project report, Oral Presentations of Seminar/Projects, Viva -Voce Examination for dissertation and End Semester closed book examination. Summative Assessment methods shall be different for theory courses and Practical Courses.

It is suggested that the examination questions should be asked keeping the learning outcomes in mind and also covering all the Units. Term papers, problem solving assignments, Lab

projects, Internship experience, group projects are recommended for achieving the expected outcomes. Wherever possible, students need to do minor projects in practical classes to learn the technology and also to apply the technology for problem solving. As this is a technology-oriented programme and new technologies are introduced quite often, care should be taken to familiarize the students with the recent advances through seminars or term papers and case studies. This should be given due weightage during continuous evaluation process. To achieve this objective, the following are suggested

(i) The end examination papers should be covering all units of the syllabus. Questions should be balanced and evaluate the comprehension, analytical and problem -solving skills.

(ii) The students should be evaluated on teamwork in addition to the technical skills through projects.

(iii) Ability to self-learning and solving new problems should be assessed through assignments, Seminars and project work.

(iv) It is recommended that 25-30% weightage of marks shall be devoted for formative assessment.

(v) It is recommended that 40% weightage be given for practical and laboratory work. (vi) Peer evaluation component is recommended for project evaluation and seminar.

(vii) Online course certification should be encouraged and equivalent grade for the same need to be worked to achieve the outcome of self-learning.

GOVERNMENT ARTS COLLEGE (AUTONOMOUS), COIMBATORE-18

PG & RESEARCH DEPARTMENT OF COMPUTER SCIENCE

M.Sc Degree Examination, Nov/Dec 2021

SUBJECT CODE: 21MCS23C **TIME : 3 Hrs**

SUBJECT: ADVANCED JAVA PROGRAMMING **MAX. MARKS: 50**

SECTION-A

I Answer ALL questions (5 *1=5)

1. When one object acquires all the properties and behaviours of parent object known as -____________ .
   a. Polymorphism b. Inheritance c. Abstraction d. Encapsulation
2. _______ in java is a data type that contains fixed set of constants
   a. Variable b. constant c. Enum d. overriding
3. Threads can be created by extending ________ class
   a. Thread b. runnable c. new d. one
4. Graphics object can only be drawn on______
   a. Window b. panel c. frame d. graphics
5. Which is not scripting languages?
   a. JavaScript b. Python, c. Ruby d. Java

II Answer Any THREE questions (3*2 =6 Marks)

6. Discuss the data types in java
7. What is meant by Instance Variable?
8. Write the java thread model?
9. How to work with fonts in AWT?
10. Write the java Script syntax with examples

Section-B

III Answer ALL questions(5*3 =15 Marks)

11. a. How to Declaring the Array Variables with examples? (or)
    b. what is method overriding in Java
12. a. Discuss about the variables in interface (or)
    b. How to creating your own sub classes

13. a. How to create tread in java? Or
    b. Discuss about the utility classes?

14. a. Discuss about how to working with frame windows in Applet with an examples?
    (or)

b. Write a short note on Transaction processing

15. a. Discuss about the JavaScript with examples? (or)

b. Discuss about document formatting with an example?

Section-C

IV Answer Any Three questions only(3*8 =24 Marks)

16. Discuss about the constructor and using this constructor to write a java program and explain
17. How to use try and catch in java programming with an example?
18. Discuss about the states of threads and write a multithread programming in java to find natural numbers and factorial of numbers
19. How to manipulating databases with JDBC with an examples?
20. Explain about event handlers and date time functions with an example of programme?