VIGNAN'S
Foundation for Science, Technology & Research
(Deemed to be UNIVERSITY)
-Estd. u/s 3 of UGC Act 1956

R22 Academic Regulations

In Compliance with NEP 2020

Prof. P. NAGABHUSHAN
B.E., M.Tech., Ph.D., FIE, FIETE, FIAPS.
Vice Chancellor, VFSTR
Life time Professor, IIIT – Allahabad, Prayagraj
(Formerly: Director, IIIT-Allahabad 2017-22)

VIGNAN'S
Foundation for Science, Technology & Research
(Deemed to be UNIVERSITY)
-Estd. u/s 3 of UGC Act 1956

PREFACE

'You are born to Blossom' – What an inspiring title the book authored by APJ Abdul Kalam and Arun K Tiwari carries. The journey to blossom has got to be heralded by education. The purpose of education is to ensure that the 'Life Blossoms'. Earning a degree and getting a placement should be the just happening things, and should not become the only celebrated goals for education. In the book cited above, Honourable Kalam, Former President of India, underscores that "The scheme of civil society depends on Educating young people to become enlightened citizens and adults who are responsible, thoughtful and enterprising"

VIGNAN aims to seed these concepts in every learner who transits through this temple of learning. The doctrine of VIGNAN entitled R-22 contains the principles of policies laid down by the University, to realize the spirit of "Blossoming the lives" providing a foundation-strong professional education on the ethos of 'Creative learning for Critical thinking and Critically analysing for Creative decision making'. Certainly, our University is one of the earliest Universities, in fact the University is a trend setting one in completely internalising the concepts of the policies brought out in National Education Policy (New Educational Policy) NEP-2020, and inculcating the spirit in R-22. The R-22 document articulates the Academic Regulations of the University, which is being presented now and shall be inforce with immediate effect from the academic year 2022-23, not only for those who have joined in 2022, also the aspirants of 2021-22 are enabled into the navigation.

R-22 presents a novel design for the academic pursuit, making an exploratory cross disciplinary traversal for a learner who should find learning both holistic and experiential. The learner is ensured to enjoy the continuity in learning and the learner is supported to align and realign, enroute utilising the benefits of constructive feedbacks that s/he receives because of continuous assessment. S/he will be empowered to enjoy the opportunities to explore, experiment and experience.

R-22 eliminates the melancholy of examinations. The expected severity of breakdown due to the anxiety of examination system is replaced by an affectionate assessment system, increasing the effectiveness in accomplishing the outcomes.

In brief, NEP-2020 compliant revised academic regulation of the University – the R-22, is VIGNAN's commitment to alleviate the acuteness in the present educational practices. It intends to provide a strategic solution to the critical observation made by Bharat ratna awardee, Professor. CNR Rao – "India has exam system, not education system. When will young people stop taking exams and do something worthwhile?" (Thought for the Day, Times of India 13.08.2022)

Here is R-22, which assures that the learners at VIGNAN are bound to do something worthwhile – very much worthwhile.

***

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

REGULATIONS - TABLE OF CONTENTS

R22 MCA

2 YEAR

POST GRADUATE PROGRAMME

MCA R22 - Academic Regulations, Curriculum and Course Contents

(Applicable for the students admitted into first year from the academic year 2022-23 onwards)

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

EXECUTIVE ABSTRACT

The Document R22 Academic regulations, Curriculum and course contents, is an articulation of the VFSTR deemed to be University's commitment towards NEP-2020, with a view that it enables student(s) to maintain the spirit of continuous learning and continuous assessment to replace the normal tendency of preparing just before a test or an examination. The proposed framework accomplishes multi-disciplinary holistic education, continuous assessment along with multiple honorable exit options if a student falls short to complete the requirements to earn the degree within the stipulated period including the permissible spill over period.

R22 is in sync with NEP-2020, with higher weight given to continuous / formative assessment, in an Integrated learning frame work which comprises of Learning – Thinking – Understanding – Skilling – Applying – Creating. Emphasis on continuous formative assessment with a creative summative assessment will facilitate the candidate to "Move away from high stake examinations – towards more continuous and comprehensive evaluation".

The MCA degree offered will be of two years' (4 semesters) duration with an exit option within this period, with suitable certification that will enable the candidate to have a professional career and as well as serve as a reminder to return and update his / her qualification in the future. Following one year of study and the completion of the required credits, a Post graduate diploma will be conferred. However, the intention of the learners is not to join for the award of the Post graduate diploma with lateral exit, but to acquire a MCA degree.

SALIENT FEATURES OF THE REGULATION

- Continuous learning
- Continuous assessment
- Post graduate Diploma
- Honorable exit options
- Sabbatical Semester option to pursue innovation, incubation, entrepreneurial and advanced
- exploratory activities and subsequent re-entry.
- Onward continuation to M. Tech degree programme

1. INTRODUCTION

This document contains the academic regulations, scheme of assessments, curriculum, detailed syllabi, course contents with text / reference books recommended, course outcomes, skills acquired, and the projects / assignments that are to be performed for each course for the conduct of 2-year MCA degree programme.

1.1 Definition

For the purpose of R22 regulation, definitions as follows shall apply:

- **"Degree"** shall refer to the MCA Degree Program.
- **"Course"** shall refer to such Course(s) for which a student shall earn Credits after due assessment as per the laid provisions. Each Course shall comprise of Lecture (L), Tutorial (T) and Practical (P) Sessions. A Course may have either or all the three Components. Project is also treated as a Course.

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

- **"Academic activities"** shall refer to the activities like Lecture (Physical Lecture Session), Tutorial (Participatory Discussion / Self-Study / Desk Work / Quiz / Seminar Presentation, etc activities that make the student absorb & assimilate, the delivered contents effectively) and Practical / Practice sessions (includes hands-on-Experience / Practice experiments/ Field Studies/ Case Studies etc activities that enable the student to acquire the requisite skill).
- **"Continuous Assessment"** shall refer to the evaluation of the student spread over the entire semester on the various constituent components of the prescribed course.
- **"Semester"** shall refer to a period covering the two assessment periods viz Formative and Summative Assessment Exams period. A semester would generally be spread over twenty weeks.
- **"Course Drop"** shall refer to a student having to undertake a repeat of the Course(s) not being able to complete the Credit requirements of the Course(s), under the conditions stipulated in the regulation.
- **"Supplementary Examinations"** shall refer to the examination(s) conducted to allow the student to appear in the un-cleared / underscored Semester - End summative assessment component, with a view to accord him an additional opportunity to improve upon his previous score.
- **"Blank Semester"** shall refer to a Semester in which a student either does not register for any course at the beginning of the Semester OR chooses to DROP all courses OR is so compelled to DROP all the courses, as the case may be.
- **"Semester Drop"** shall refer to availing a blank semester. However, if drop is availed to pursue a creative extension activity, then it is defined as semester sabbatical.
- **"Spill Over Semester"** shall refer to the additional semester(s) beyond the completion of prescribed normal semesters.
- **"AAA Section"** shall refer to the Academics, Assessment and Award Section of the Institute.
- **"Attendance"** refers to the Physical personal presence in an academic activity session.
- **"Summer Semester"** refers to a Semester that is decided to be held during the intervening period of Even and Odd Semester (i.e. Summer Vacations period).
- **"School"** refer to a division of institute dealing with two or more specific area of discipline / study comprising of the departments related with exclusive emphasis on trans-disciplinary research.
- **"Department"** refer to a division of institute dealing with a specific area of discipline / study.
- **"Center"** refer to a structured unit within the school / department established with the purpose to carry out advanced research
- **"Grade Point"** refers to the quantification of the performance of a candidate in a particular course as defined herein.
- **"Honorable Exit Option"** refers to the Exit Options available to students when they are unable to complete the prescribed two-year MCA Degree program in four successive years.
- **"SGPA"** refers to the Semester Grade Point Average and is calculated as detailed in the regulations subsequently.
- **"CGPA"** refers to the Cumulative Grade Point Average and is calculated as detailed in the regulations subsequently.
- **"Division"** refers to the Division awarded to the student as per the mechanism detailed in the regulations subsequently
- **"HoD"** refers to the Head of the respective Department, where the student is enrolled for his/her Branch of Study
- **"Internship"** refers to onsite Practical Training offered by reputed companies / Institutions, in India or abroad. To be undertaken only upon prior approval of the respective HoD.
- **"Project"** refers to a course executed by a candidate on a specific research problem at VFSTR / any organization of repute. To be undertaken only upon prior approval of the respective HoD.
- **"Credit equivalence and credit transfer committee"** refers to the committee designated to look into for credit equivalence and credit transfer

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

1.2 Academic Administration

The academic programmes of VFSTR are governed by the rules and regulations approved by the Academic Council from time to time. The various academic activities are conducted following a fixed time schedule duly approved by the Academic Council in line with the AICTE / UGC regulations. The academic activities of VFSTR are followed meticulously as specified in the academic calendar as approved by the Academic Council. This academic calendar is shared with all the stake holders well before the beginning of the respective academic year. The curriculum and the course contents of all the programmes are discussed by the respective Board of Studies (BoS), analyzed and recommended for implementation. The Academic Council, being the highest statutory body, chaired by the Vice-Chancellor, meets at least twice or thrice a year and discusses, suggests and approves all the important academic matters related to curriculum and course contents in particular including the recommendations of BoS. The revised regulations (R22) was approved in its 31st Academic Council meeting on 30-07-2022.

1.3 Program Duration

For the MCA program, the regular courses including theory and practical are offered over a period of two years in four semesters. The normal duration to complete the MCA program is two years. However, a student can avail of the benefit of spill-over period of 2 years, that is the maximum duration of four years can be availed by a candidate to complete the MCA programme in a slower pace if he / she desires. The candidate failing to complete the requirements will be considered for the honorable exit as applicable.

1.4 Courses and Credits

The term course is used in a broader sense to refer to so called papers such as 'theory subject', 'Computing practice', 'Socially relevant project using Design thinking', 'Internship', 'Project' etc. A course can be of theoretical and / or of practical nature, and certain number of credits are allotted to it depending on the number of hours of instruction per semester. For a course offered in a semester, one hour of lecture (L) instructions carried out in a week is considered equivalent to one credit, whereas two hours of practical (P) sessions done in a week are considered equivalent to one credit respectively. Depending on the course two hours of tutorial (T) sessions may be considered equivalent to one credit. A student earns these credits when he / she successfully completes the course.

1.4.1 Content Delivery of a Course

Content delivery of a Course in the MCA Degree Program shall be through, either or all, of the following Methods:

i. **Lecture** - refers to Lecture Session(s) through classroom contact session wherein students will learn by listening. Denoted by "L".

ii. **Tutorial** - refers to transaction(s) consisting of Participatory discussion / Self-study / Desk work / Brief presentations by students along with such other novel methods that enable a student to efficiently & effectively absorb and assimilate the contents delivered in the lecture sessions. Denoted by "T".

iii. **Practice** - refers to Practice / Practical sessions and it consists of Hands on experience / Computing practices / Field Studies / Case Studies / Socially relevant project / Major Project / Internship, that equip the students to acquire the much required skill component. Denoted by "P".

1.5 MCA Degree

All students formally and conventionally enroll for MCA degree programme. They have to earn **80** credits for the award of degree as specified in the Curriculum.

1.5.1 A provision is also created for a candidate who is enrolled for MCA degree to further continue his / her course of study to M. Tech degree after completing the credit requirements of MCA. Annexure-2 provides the supplement regulations for onward continuation to M. Tech degree programme.

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

1.6 Composition of an Academic year

An academic year is composed of an Odd semester (20 – 22 weeks), an Even semester (20 – 22 weeks) and a Summer semester (6 – 8 weeks). The regular semester that begins in July / August is known as odd / first semester and the one that begins in December / January is known as even / second semester (Figure 1). The instructional days for a regular semester shall be a minimum of 90 working days exclusive of days earmarked for summative assessment.

YEAR OF 12 MONTHS											
1	**2**	**3**	**4**	**5**	6	7	8	9	**10**	**11**	**12**
July/ Aug.	Aug./ Sept.	Sept./ Oct.	Oct./ Nov.	Nov./ Dec.	Dec./ Jan.	Jan./ Feb.	Feb./ Mar.	Mar./ Apr.	Apr./ May	May/ June	June/ July
ODD SEM/ FIRST SEM					EVEN SEM/ SECOND SEM					SUMMER SEM	

Figure 1: Distribution of semesters during an Academic Year

1.6.1 Before the commencement of the semester, a candidate has to pay the stipulated tuition fee and submit an application detailing the courses he / she intended to register, valid for that respective Odd / Even semester. The maximum number of credits per semester will be 25 credits inclusive add-on credits. The intended semester wise coverage will be as presented in the curriculum.

1.6.2 Summer semester is a short duration semester program that will be generally conducted during the semester break between even semester and odd semester. The students having 'R' (Repeat grade) courses may register for the course work during this semester to get a chance for successfully completing the 'R' courses. In general, supplementary assessments are conducted in the later part of the summer semester. However, the courses offered in summer semester and the number of courses a student can register are subjected to academic and administrative convenience. A student may register up to a max. of 16 credits in a summer semester.

1.6.3 Exception to the routine practice of registering for 'R' courses in summer semester, a student can register in a course offered by a visiting expert during the summer vacation which may be equivalent to a department elective or an Add-on-course. The candidates can register for such courses within the scope of 16 credits. Candidate may also avail summer semester for summer internship opportunities, which may be considered as Add-on credits.

1.7 Semester wise provisions

A student may register for a max of 25 credits per semester as prescribed or otherwise he / she may include the Repeat courses in the event of having not successfully completed a course or courses in the earlier semester. However, a student may also opt to go in a slower pace to earn the credits less than the prescribed max of 25, including even 'Dropping' a semester for special reasons.

It should be clearly underscored that a candidate should on priority register for Repeat (R) credits if any, during a regular semester, within the said scope of 25 credits; in case he / she cannot be sure of completing the 'R' credits in Summer semester.

1.7.1 During the first two years from the date of admission to MCA a candidate has to pay the semester / annual fees as prescribed irrespective of the less number of credits that he / she would register or even opt to Drop a semester.

1.7.2 If a candidate gets into a spillover semester beyond two years up to a maximum of four years he / she has to pay semester fee proportional to the credits that he / she registered in that spillover semester as prescribed from time to time.

1.7.3 A candidate has to pay an additional fee proportional to the number of credits for registering in a summer semester as prescribed from time to time.

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

2. CURRICULUM

The Department prescribes semester-wise curriculum encompassing different courses. Every course offered will be designated in a L-T-P structure. The theory courses comprise of L (and / or T & P hours) whereas the practical courses include computing practice instructions (T) and practical sessions (P). Amalgamation of theory courses with practical sessions is predominantly seen in this curriculum.

2.1 Distribution of credits

The overall distribution of credits for various categories of courses in the curriculum of MCA programme is represented in Table (1) as given below.

Table 1 : Credits Distribution for Various categories of courses

Category of Courses	Number of Credits	Percentage of Credits
Professional Core	46	57.5%
Dept. Electives	11	13.75%
Humanities and Management	6	7.5%
Basic Sciences	3	3.75%
Projects	14	17.5%
Total	**80**	**100%**

2.2 Organization of course contents

Courses offered in the program is composed of two modules covering all the course contents required for a candidate to obtain knowledge and skill. Any contents in each module is further distributed among two units; wherein Unit -1 contains 'Fundamentals and Broad perceptive' of the module. Unit-2 comprises of the extension / advanced topics of Unit-1 as well as necessary practice models for validation / applying the knowledge gained during L / T sessions. The modular period is about 8 weeks. The first unit in a module may be covered in 3 to 4 weeks and the second unit of the module may be of 4 to 5 weeks (Figure 2). By the end of each module a candidate must be in a position to translate his / her L-based knowledge into P-based skill as prescribed in the curriculum. Individual formative assessment shall be in place for each module and a single semester-end summative assessment for the course composed of both the modules.

YEAR OF 12 MONTHS											
1	**2**	**3**	**4**	**5**	6	7	8	9	**10**	**11**	**12**
July / Aug.	Aug. / Sept.	Sept. / Oct.	Oct. / Nov.	Nov. / Dec.	Dec. / Jan.	Jan. / Feb.	Feb. / Mar.	Mar. / Apr.	Apr. / May	May / June	June / July
ODD SEM / FIRST SEM					EVEN SEM / SECOND SEM					SUMMER SEM	

Module- I		**Module- II**			**Module- I**		**Module- II**			
U1	**U2**	U1	U2		U1	U2	U1	U2		

Figure 2: Unit-wise distribution of course contents in a module and their mapping with Academic Calendar; U= Unit

3. CHOICE BASED CREDIT SYSTEM

The MCA programme comprises of a set of courses - Basic sciences, Humanities and Management, Professional core and Elective courses. VFSTR offers flexibility for students to choose courses of their choice and obtain the credits satisfying the minimum credits criterion.

3.1 Basic Sciences, Humanities and Management

One Basic science course was introduced to offer the knowledge of statistics that form the foundation for all computer science related courses for 3 credits. Courses of Humanities

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

and Management are also offered for 6 credits. The main purpose of offering Management courses is to impart Management skills to students so that they would pursue allied career opportunities.

3.2 Professional Core

Professional core courses are mandatory courses (46 credits). These are designed to offer the essential fundamental knowledge and skills required for the MCA Programme.

3.3 Electives

A candidate has a choice to choose the elective courses (11 credits). A list of elective courses is pooled together, enabling a candidate to choose the electives from a pool so that he/she can focus to a specific theme. Otherwise also he / she can exercise the choice to choose electives from across the pools. There may be courses which may not be listed under any pool, which are called 'Free elective courses'. Elective courses are spread over two semesters from second to third semester, to enable students earn credits from a chosen pool or otherwise. Out of 11 credits in electives category interested candidates can also earn 3 credits through MOOCs offered via Swayam platform.

3.4 Socially relevant project using Design Thinking

These projects are designed and executed by students during the first year of their program. Design thinking is a creative problem-solving approach that focuses identifying new products / processes by joining both human & social capital. It ensures that products, services and processes are rooted in the needs of people, communities and / or end users with an emphasis on solutions creating social and environmental value. The supervision of Socially Relevant Project using Design Thinking will be done by faculty of department who serve as supervisor. The minimum duration of the project is 64 hours including writing of project report and submission for assessment. A batch of maximum 2 students can take part in the project. Performance will also be assessed in the modular framework for formative and semester-end summative, successful completion will earn 2 credits.

3.5 Project

Students may opt for Project work in lieu of internship. Such students may avail research -internship support from any institution well known for research and development (R&D). They may also take up project work in VFSTR itself. Each candidate has to submit interim reports and a final report which are mandatory requirements towards the partial fulfillment of project credits requirements. It bears a weightage of 12 credits with a duration of 90 working days. During the semester the student under the guidance of a faculty member(s) will involve in an innovative design / research through the application of his / her knowledge gained in various courses studied. He / she is therefore expected to present a survey of literature on the topic, work out a project plan and carry it out through Experimentation / Modeling / Simulation / Computation. Through such a project work, the student is expected to demonstrate system analysis, design, presentation, and execution skills. Performance in the project will also be assessed in the modular framework for formative and semester-end summative.

3.6 Internship

A student can undertake internship in lieu of project work in the industry for one complete semester during fourth semester in lieu of major project work. It bears a weightage of 12 credits. This is aimed at training students in solving / understanding real-life problems through application of engineering analysis, design, evaluation, and creation, particularly in association with practitioners and experts in the industry. The procedures for obtaining the internship placements and allocation of the same to the students are as per University defined norms outlined in the 'internship programme operational guidelines' manual. Even during internship, a student is preferably expected to carry out a focused study on one topic / problem in consultation with the interning institute. Internship progress report should be submitted periodically and finally a detailed internship report should be submitted duly certified by a mentor from the internship institute. Performance in the internship will also be assessed in the modular framework for formative and semester-end summative.

4 ATTENDANCE

It is mandatory for the student to attend the course work in each semester as per the academic schedule of that semester. VFSTR expects 100% attendance. However, the attendance in each course shall not be less than 75 % of the aggregate of all L, T, P sessions conducted in that course.

a) The attendance calculations will be periodically reviewed at the end of every 4 weeks. The details of attendance status will be shared with the Parents / Guardian. The final status of attendance will be reported at end of 15th week granting the advantage of the attendance for the 16th week for the purpose of attendance shortage calculations.

b) The shortage of attendance may be condoned up to 10% on the ground of ill-health, social obligations, participating / representing in sports / cultural events, placement activities etc.

c) Documentary evidence like medical reports and certificates issued by concerned bodies is to be produced on time as support for the attendance shortage due to ill-health. These cases are subjected to the scrutiny of a committee constituted for this purpose by the Vice-Chancellor. The decision of the committee shall be final.

d) Prior approval has to be taken from the HoDs for the other types of leaves.

e) The courses where the student shortage of attendance was not condoned shall be considered as 'Repeat' category courses and will be under 'R' grade in the student's semester transcript. Student should re-register for these courses during the summer semester or whenever the course is offered next time during regular semesters. These re-registrations are subjected to the regulations at the time of re-registration. In case of core courses, the same core has got to be re-registered. However, in case of an elective a candidate may exercise a choice of choosing different elective in place of 'R' graded elective.

The students who are put into 'R' grade will not be allowed to take up the summative assessment in that semester.

5 ASSESSMENT

Teaching-Learning and Assessment should go hand in hand and complement each other. Continuous assessment plays a vital role to enable the student to get synchronized with the teaching-learning process. Assessment mechanism adopted in the institute is aimed at testing the learning outcomes in tune with the outcome based model of education. The focus, is thus on assessing whether the outcomes are realized by the end of the course.

The performance of a student in each course is assessed on a continuous basis during the semester through various in-semester and end-semester assessment models. The marks awarded through continuous assessment are referred to as Formative assessment marks. The marks awarded through end-semester tests are referred to as Summative assessment marks (Figure 3). Both the formative and summative assessment marks are considered for awarding the final marks and the grade point in a particular course.

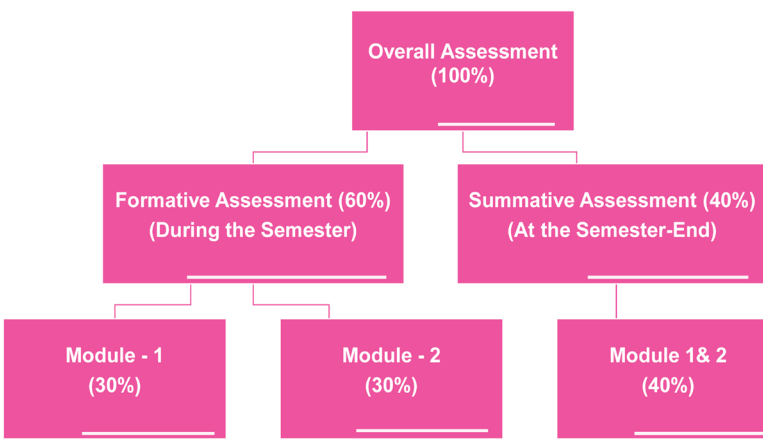

Figure 3: Categories of Assessments in place for R22

5.1 Marks distribution

For each course, the maximum sum of formative and summative assessment marks put together is 100, in the ratio of 60:40 respectively.

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

5.2 Qualifying criteria

To be declared successful in a course, a student must secure at least a grade 4.5 in a scale of 10 based on the total maximum marks which is inclusive of formative and summative assessment. The students should also get 40% from the maximum marks allotted for formative and summative assessments individually.

The hierarchy of qualifying criteria is as follows:

i. Attendance compliance should be 75% or within condonable range; else the candidate is put into 'R' grade

ii. In formative assessment, a candidate should secure a minimum of 40% ie. 24 marks out of 60; else the candidate is put into 'R' grade

iii. In summative assessment, a candidate should secure a minimum of 40% ie. 16 marks out of 40; else the candidate is put into 'I' (Incomplete) grade.

iv. Collectively the candidate should secure a min. grade of 4.5 in a scale of 10 after relative grading (section 7); else the candidate has to choose either 'R' or 'I' grade duly being counselled.

v. A candidate who has secured grade < 5 in a course may be permitted (optional) to volunteer to improve his / her grade by opting suitably 'R' or 'I' in that course.

The candidates with 'R' grade should re-register for 'R' courses either in Summer semester or in a regular semester as and when the courses are offered. The candidates in 'I' grade are allowed to appear for supplementary summative assessment whenever the semester-end assessments are conducted.

To assess Binary graded courses / special projects / courses, not fitting into the categories described here, a suitable assessment procedure will be evolved in consultation with experts of that area and adjudicated by the committee constituted for that purpose. The decision given by the committee will be final. The appended assessment scheme shall be announced by the course coordinator during the commencement of course.

YEAR OF 12 MONTHS											
1	**2**	**3**	**4**	**5**	6	7	8	9	**10**	**11**	**12**
July/ Aug.	Aug./ Sept.	Sept./ Oct.	Oct./ Nov.	Nov./ Dec.	Dec./ Jan.	Jan./ Feb.	Feb./ Mar.	Mar./ Apr.	Apr./ May	May/ June	June/ July
ODD SEM/ FIRST SEM					EVEN SEM/ SECOND SEM					SUMMER SEM	

Module- I		**Module- II**			**Module- I**		**Module- II**			
U1	**U2**	**U1**	**U2**		**U1**	**U2**	**U1**	**U2**		
Formative Assessment				SA	Formative Assessment				SA	

Figure 4: Schedules of formative and summative assessments in line with Academic calendar. SA = Summative assessment

5.3 L-based courses integrated with P / T

5.3.1 Formative Assessment

The scheme of formative assessment is designed to promote the continuous learning. Scheme consists of assessments planned at institute level and assessment that may be scheduled by the course instructor (Figure 4). Institute level assessments shall be scheduled by the office of AAA. Respective Faculty Member(s) shall declare the schedule of Continuous Practice Assessments (CPA), Quiz, Tutorials, Assignments, Seminars, Discussions, etc. Some of the components may also however take place in an unscheduled manner like Surprise Tests. However, students shall be made aware of the assessment modalities that are going to be followed in a course by the faculty, under information to the HoD.

To monitor the progress of students, continuous assessment comprising of five targets (T1, T2, T3, T4 and T5) is advocated in each module for a maximum of 60 marks. For a class (or section) of 60 to 70 students, formative assessment commences by the announcement of module bank containing 10 problems for each module in a course. Nature of problems in the module bank shall be at the level of creative / exploratory / design / thought provoking covering the complete syllabus of a module at somewhat advanced / challenging level.

The purpose of creating module bank of 10 problems is to assign one problem each to 2 batches of 3 - 4 members. The batches are composed of randomly picked up candidates. These batches remain same for all courses and also for the P-sessions in the courses in that semester and are created in the beginning of that semester.

The purpose of assigning one problem to two batches is to create a healthy competitive spirit between the two batches.

The modality of evaluation of five targets is listed here under:

a) **T1:** During 5th or 6th week of each module a classroom test shall be conducted. T1 consists of two parts: A and B.

Part A consists of one random problem from the module bank and vary from batch to batch. All the questions in the module bank shall be distributed among students and students shall know the question to be answered only on the day of test in the examination hall.

Part B consists of one common problem at fairly application / advanced level (not at all prior notified) from outside the module bank for all the students.

T1 shall be paper based and proctored test for a period of 60 Min (Maximum) which shall be assessed for 30 marks and downscaled to 10 marks.

For the students who for justifiable reasons could not attend the classroom test on the scheduled day, a re-test maybe conducted. However, Part-B will contain a new question and Part-B will have higher weightage than part-A or full weightage could even be allotted for Part-B in such an event.

b) **T2:** Immediately follows T1. Students in a specified batch who now have received the same question during T1 will work further on that problem for T2.

T2 is primarily an extension of problem received in T1 for carrying out validation study: Case studies / Simulations / Computing exercises. Each batch shall interact with the course instructor to finalize the nature of validation and expected to complete the exercise within 10 to 15 days after T1.

Course instructor should ensure assigning a different case study / a different scope for validation study for each batch in case the same problem is assigned to two batches.

Course instructor shall assess every student in a batch for a max. of 10 marks based on his observation, interaction and / or based on at least 2 reviews.

c) **T3:** T3 shall be conducted during the last week of each module. Student batches are expected to submit a report, clearly documenting the work executed during T2. The report should be in IEEE format and additionally a voice in-built PPT should be prepared and submitted.

The report and presentation shall be assessed by the course instructor for 10 marks for every student. In certain cases, a course instructor can call for a physical presentation also by a batch.

d) **T4:** T4 is a comprehensive module test, conducted for 30 min. comprising of 20 multiple choice questions (MCQs) covering the holistic content of module. T4 shall be evaluated for a max. of 10 marks @ ½ mark for each question. T4 will be conducted in ON-LINE mode

There shall be two tests in each course in a day and the best performance of the tests shall be considered for awarding the marks.

Two sets of question papers each containing 20 questions should be set. The theme of the questions could be similar across the sets. When the test is administered online, every student receives the questions in shuffled sequence and also the choices in shuffled sequence. Therefore, the choice like both 'a' & 'b' above. Neither of 'a' and 'b', all the three a, b, c will not be set.

e) **T5:** T5 assessment is based on Practice or Tutorial assignments. Implementation, Report presentation and Discussion shall happen in a continuous mode throughout the module period.

At least 4 such continuous practice assessments (CPA) / assignments per module shall be conducted by course instructor. The marks will be @ 5 marks per assignment totaling up to 20 per module.

The results of the tests are to be normally announced within three working days on completion of the assessment and the performance is to be discussed in the class.

f) The total marks per module is 60 - T1 (out of 10), T2 (out of 10), T3 (out of 10), T4 (out of 10) and T5 (out of 20).

g) Total marks for both the modules from formative assessment will be added up to 120, which will be suitably mapped down to a max. of 60 marks. The mapping policy should be decided by the lead instructor / instructors in consultation with the HoD. The mapping policy should be shared with Dean AAA for the purpose of documentation.

h) The marks scored in Module-1 for a max. of 60 should be entered / submitted latest by 9th week and of Module-2 latest by 17th week of the semester. Consolidated score of for a max. of 120 suitably mapped down to a max. of 60 marks should be submitted latest by 18th week of semester enabling the declaration of 'R'- grade before the commencement of summative assessment.

i) A candidate put under 'R' will not be permitted to take up the summative assessment.

5.3.2 Summative Assessment

a) An instructor may choose one of the two formats for conducting summative assessment for L-based courses integrated with T / P

i. 15 + 25 marks format or 20 + 20 marks format (following b, c, d below)

ii. 40 marks format (following c, d below)

b) If summative assessment is in two parts format:

i. Part-I will be the assessment of capstone project which is pre-assigned during the module-2 period or will be the exploratory review assessment of all course practice assignments.

ii. Part-II will be based on a written examination for a max. marks of 80, as in c& d below, which is suitably mapped down to 25 or 20 based on the selected pattern of format.

iii. A candidate should attend both the parts of summative assessments; else he will be put into I grade.

c) For each L-based course integrated with T / P, the summative assessment shall be conducted by the Institute for a duration of 150 min. and for a maximum of 80 marks. Contents for summative assessment shall cover the breadth and depth of the complete syllabus that is mentioned in the two modules of a course.

d) The question paper for end-semester theory examination consists of two parts as given in Table (2).

Table 2: Theory Examination Question Paper Pattern

Part No.	No. of Questions	Marks for each Question	Marks	Choice
A	4	10	40	No
B	2	20	40	No
Total Marks			**80**	

e) The questions will be comprehensive covering the entire course syllabus and any single question should not necessarily be limited to any particular unit / module.

f) The marks scored out of 80 is suitably mapped down into a score out of 25 or 20 based on the format.

g) Total marks of summative assessment will be for a max. of 40 irrespective of format of evaluation.

h) The award of 'I' grade is solely based on marks scored in summative assessment out of 40, if he / she does not score a min. 16 out of 40 (40%).

5.4 P-based Courses

The detailed information consisting of computing practices, batch formations, practice session schedules, etc., will be displayed / informed to the student in the first week of the semester so that the student come prepared for the practice sessions. Copies of the practice manual will be made available to the students along with the schedule. The practice manual will consist of the detailed procedure to solve the problem, format for record writing, outcomes for each practice exercise and possible set of short questions to help students gain critical understanding.

5.4.1 Formative Assessment

During practice sessions, a brief viva-voce is conducted for each student on the task he / she is carrying out on that day. Some of the parameters that could be included in the Continuous Practice Assessment (CPA) are given in Table (3). The set of parameters may slightly differ from one course to the other, and will be announced before the commencement of the practice session. These parameters are assessed for each practice session

Table 3: Suggested parameters for Continuous Practice Assessment (CPA)

S. No	Component	Marks
1	Report of about 1 page on proposed design layout and background theory before the start of practice session	4
2	Viva and interaction to evaluate understanding of concepts	4
3	Validation of results	4
4	Analysis of results and interpretation	4
5	Finalized report submitted on the next week	4
Total		**20**

This assessment is carried out for each practice session and the total marks of all practice sessions will be suitably mapped down to a max. of 60.

5.4.2 Summative Assessment

End semester examination for each practical course is conducted jointly by two examiners. The examiners are appointed by Dean, AAA from the panel of examiners suggested by the respective Heads of the Department. In some cases, one of the examiner may be from outside the institution and will be identified as external examiner. The scheme of assessment may vary depending on the nature of the course, which shall be shared with student by the course instructor. The summative assessment will be conducted for a max. marks of 40. The general scheme of assessment is given in Table(4).

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

Table 4: Suggested end-semester summative assessment pattern for P-based courses

Component	Marks		
	Examiner 1	Examiner 2	Total
Objective & Procedure write up including outcomes	4	4	08
Preparation of algorithmic lay out	4	4	08
Computation of results	4	4	08
Analysis of results and Interpretation	4	4	08
Viva Voce	0	8	08
Total Marks	**16**	**24**	**40**

5.5 Assessment and Grading of MOOCs based elective

Whenever a candidate opts for a course through MOOCS offered via the Swayam platform, he / she has to learn and undergo assessment as per Swayam norms. Upon the declaration by Swayam, that the candidate has successfully completed the course, the candidate is said to have earned the credits under credit equivalence and credit transfer. The online course committee will also translate the score awarded by Swayam into an equivalent grade in a scale of 10 for incorporation by Dean AAA.

In case the candidate is unsuccessful, and then if the candidate has secured less than 20% in the final examination conducted by Swayam, the candidate will be placed into 'R' grade, and if it is ≥ 20%, then the candidate will be placed into 'I' grade. The online committee decides the equivalence score for formative assessment for a max. of 60 based on both the scores he/ she has earned in the assessments and the final examination conducted by Swayam in case of 'I' grade. Supplementary examination for 'I' grade will be conducted by VFSTR. The candidate has to re-register for the same Swayam course or an alternative Swayam course or may choose any other elective offered by the department in place of MOOCs course in case he / she received a 'R' grade.

5.6 Socially Relevant Project using Design Thinking

Socially relevant project using design thinking work is undertaken in the 2nd semester and continued till the end of the semester for earning 2 credits by each candidate.

5.6.1 Formative Assessment

The assessment will be carried in four reviews in a systematic way wherein; first two reviews are conducted in Module-1 and remaining two reviews are conducted in Module-2 by a team of two senior faculty members appointed by the Head of the department.

The detailed assessment guidelines and scheme are to be announced along with the assessment schedule as mentioned in the Table(5).

Table 5: Schedule and parameters followed for formative assessment

Module	Schedule	Review number	Points to be considered	Max. Marks
Module -1	4th week	First review	• Identification of specific area out of broad areas under the supervisor • Identification of outcomes in line with programme objectives. • Feasibility of contributing to the attainment of outcomes	15
	8th week	Second review	• Identification of tools / equipment / surveys / training needs / etc.. • Understanding by individual students on the overall aspect of the project. • Completion of literature survey • Design of equipment set up • Acquisition / learning of the tool required • Readiness of about 25% documentation	15

Module -2	12th week	Third review	• Progress review as per mechanism / schedule identified • Evaluation or validation techniques of the project • Structure of project report • Individual student contribution in above activities.	15
	16th week	Fourth review	• Presentation of results, analyses and conclusions • Meeting of objectives defined in first review • Preparation of report • Understanding by individual students on the overall project • Submission of technical article	15

5.6.2 Summative Assessment

Summative assessment will be done jointly by two examiners one is from VFSTR as internal examiner and the other may be from other institution as external examiner. These examiners will be appointed by Dean-AAA from the panel of examiners suggested by the respective Head of the Department. The scheme of assessment will be report (15 marks), presentation (10 marks) and demonstration (15 marks) respectively. Points to be considered during the review.

a) Presentation of results, analyses and conclusions
b) Meeting of objectives defined in first review
c) Preparation of report
d) Understanding by individual students on the overall project
e) Individual student contribution
f) Progress of project as per schedule
g) Submission of technical article / prototype realization

5.7 Project

Those students who do not opt for the semester-long internship, carry out their major project at VFSTR and submit their report which is a mandatory requirement for the award of degree. These projects are usually done individually, during the IV semester, under the guidance of a faculty member. Every student, in consultation with the guide, should define the project and also the probable procedure of carrying it out and submit the same to a committee consisting of 2 to 3 faculty members appointed by Head of the Department. This is to avoid the repetition and also to come up with a roadmap for completion of the project within the time stipulated. The students are encouraged to select topics related to ongoing research and consultancy projects. The students are expected to carry out and present a survey of literature on the topic, work out a project plan and its implementation through experimentation / modelling / simulation / computation. They are also expected to exhibit system analysis, design, and presentation and evaluation skills. The entire process of grouping of student batches, and identification of respective guides etc., is to be completed by the end of III semester, so that students can start of their project work immediately after III semester.

5.7.1 Formative Assessment

The progress of project is reviewed twice in a module by the “Project Review Committee (PRC)” and formative assessment marks are awarded based on these reviews. The Project review committee consists of

a) Head of Department or his / her nominee – Chair person
b) A senior faculty member identified by the HoD – member
c) Project supervisor – member

Review schedules of PRC are to be announced by the department immediately after the commencement of class work. The review presentations are open to all the students of that section and attendance is compulsory. The first review should be of 15 minutes/batch; the

remaining reviews should be around 30 minutes / batch. Before every review the batches should submit their PPT along with a brief report of not exceeding two pages. It is to be expected by the committee that student communicates/publishes research article based on the project work prior to graduation. The following aspects may be considered by the committee for assessment Table(6).

Table 6: Schedule and suggested parameters to be considered for formative assessment

Module	Schedule	Review number	Points to be considered	Max. Marks
Module -1	4th week	First review	• Identification of specific area out of broad areas. • Identification of outcomes in line with programme objectives • Feasibility of contributing to the attainment of outcomes	15
	8th week	Second review	• Identification of tools/training needs / etc.. • Understanding by individual students on the overall aspect of the project • Completion of literature survey • Design of project set up	15
Module -2	12th week	Third review	• Acquisition / learning of the tool required • Readiness of the layout of the project report • Progress review as per mechanism / schedule identified • Individual student contribution in above activities	15
	16th week	Fourth review	• Presentation of results and conclusions • Meeting of objectives defined in first review • Submission of draft report • Understanding by individual students on the overall project • Individual student contribution • Progress of project as per schedule	15

5.7.2 Summative Assessment

At the end of the semester, during 18th to 20th week of the semester the summative assessment will be conducted in two phases

Phase – I (during 18th -19th week): this is an evaluation for a max of 20 marks. A committee of two members comprising of HoD's nominee and Guide will assess the project work which will involve going through the project report (6 marks), project presentation (7 marks) and demonstration of the project (7 marks).

Phase – II (during 20th week): A final presentation and defense assessment for a max. of 20 marks will be carried out by one-man committee composed of an external expert who is chosen by the Dean AAA from a panel of examiners suggested by the HoD. The format for evaluation will involve going through the project report's quality (6 marks), presentation (6 marks) and interaction and defense (8 marks).

The qualifying marks will be finalized considering the marks scored in both the phases (I & II) of summative assessment.

In case the candidate is placed in 'I' grade, he / she has to appear for both Phase-I and Phase-II assessments, which will be held within the 15 days after declaration of results. In the consecutive assessment also if the candidate fails to secure min. required score then he / she will be placed in 'R' grade.

5.8 Semester-long Internship

Internship work is undertaken in the IV semester by a student in an industry, under the joint supervision of industry personnel and an internal faculty member. 60% of the marks of Internship are allotted through continuous evaluation as formative assessment and the remaining 40% are based on end semester assessment.

Table 7: Formative Assessment scheme for Internship Reviews

Module	Review number	Schedule	Max. Marks
Module -1	First review	4th week	15
	Second review	8th week	15
Module -2	Third review	12th week	15
	Fourth review	16th week	20
Total			**60**

a) The progress of internship work is reviewed twice in every module by the "Internship Review-Committee" and marks for formative assessment are awarded based on these reviews.

b) The Internship Review Committee (IRC) consists of Head of Department or his / her nominee (Chairman), the internal and external (industry) supervisors.

c) The IRC may not be the same for all students; however, the same IRC should exist for entire duration of the internship program of any single student.

d) The schedule and the scheme of evaluation are to be announced with internship notification. The internship reviews may take place at the place of internship or at the university, as decided by the interning organization or may be conducted in the blended mode.

5.8.1 Formative assessment: Internal reviews at the place of internship

The internal supervisor will interact with the guide allotted at internship offering industry based on the schedule given to conduct the reviews. Scheduled reviews can be conducted by IRC on online mode for discussion / presentation. The 15 marks obtained by students for each review will be scaled to allotted marks as given in Table (7).

a) Students should submit a report (not more than two pages) explaining about the progress of their work, mentioning clearly details like the machines or software handled / adopted, type of data collected and his / her understanding and contribution in the programme, before the supervisors.

b) The candidate should clearly present the completion of stipulated assignments set by the industry supervisor for that period.

c) The evaluation will be based on a & b above and also based on regularity and discipline maintained in the internship venue.

Table 8: Suggested scheme of assessment for every review

Component	Total
Regularity and interaction	3
Application of knowledge	2
Gaining of new knowledge / skills / literature survey	5
Internship progress	3
Report	2
Total marks	**15**

5.8.2 Summative assessment – Internship

At the end of the semester, the student shall submit a comprehensive report of internship covering the work done and make a final presentation in two phases as follows:

Phase –I (during 18th -19th week): A committee of two members comprising of internal supervisor and HoD's nominee will assess the overall internship participation by the candidate and his final report through presentation made by the intern. The internship report (6 marks), presentation (7 marks) and overall impression (7 marks) during the internship will be evaluated respectively.

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

Phase –II (during 20th week): A final presentation and defense assessment for a max. of 20 marks will be carried out by one-man committee composed of an external expert who is chosen by the Dean AAA from a panel of examiners suggested by the HoD. The format for evaluation will involve going through the project report's quality (6 marks), presentation (6 marks) and interaction and defense (8 marks).

The qualifying marks will be finalized considering the marks scored in both the phases (I & II) of summative assessment.

In case the candidate is placed in 'I' grade, he / she has to appear for both Phase-I and Phase-II assessments, which will be held within the 15 days after declaration of results. In the consecutive assessment also if the candidate fails to secure min. required score then he / she will be placed in 'R' grade.

6 SEMESTER-END ASSESSMENT ACTIVITES

6.1 Setting of semester-end summative assessment question papers will be coordinated by the lead instructor assigned for a particular course. Two sets of question papers will be submitted latest by 12th week of the semester.

6.2 There shall be 'Summative Assessment Question Paper Scrutiny Committee' which would be constituted with external experts. Experts are empowered to modify / rephrase the questions to maintain a high standard of the semester-end assessment. The review should be completed by the 14th week of the semester. The review process will be coordinated by a committee of School Dean, HoDs and external experts.

6.3 The question wise marks scored in the summative assessment out of a total of 80 will be made available online within two weeks from the last date of examination and would be kept active for 24 hours. Latest by the end of 48 hours from the instant of notification any candidate can submit an appeal online providing question wise claim.

6.4 Claims for re-assessment on P-based courses are not allowed.

6.5 The appeals will be attended within next three working days. Fees for appeal, as decided from time to time, has to be remitted online along with the appeal.

6.6 Final results and grades will be computed as explained in the next section

6.7 Final results and grades shall be announced within four weeks of completion of the last examination of the summative assessment (within two weeks from the last date of appeal). Grades are published on the University website, and also informed to the parents and students through SMS.

6.8 Provisional Grade cards will be issued within two weeks after the announcement of grades. Grade card will contain three parts. Part 1: details of successfully completed courses. Part 2: Details of 'I' grade courses. Part 3: Details of 'R' grade courses.

7 COMPUTATION OF GRADING

7.1 Formative assessment decides the list of 'R'- candidates. Therefore, these candidates will not be considered for grading computation. Summative assessments decide the list of 'I' candidates. Therefore, these candidates will not be considered for grading computation

7.2 The candidates who have successfully completed both formative and summative assessments will be considered for computation of relative grading.

7.3 Threshold value (Th) for relative grading in each course is arrived after studying the marks distribution in that course by a committee constituted by Dean, AAA. The threshold value is decided by the upper bound marks of the major chunk of the class keeping the top outlier scores away from consideration (the least upper bound). The threshold value will be slightly greater than upper bound marks or may be equal to the upper bound marks.

7.4 The total marks M = marks scored in the formative assessment + marks scored in the summative assessment is transformed into relative grade expressed accurate to two decimal places as follows:

Relative grade point (P) = (M / Th) X 10 [and limited to 10]

7.5 If students require course wise percentage equivalence, then the calculation will be based on the following

Course wise percentage equivalence = (M / Th) X100 [truncated to two-digit integer and limited to 100]

7.6 After relative grading, a student is assigned a 'Letter Grade (G)' for each course as per Table (9). The grade and the corresponding letter grade represent the outcomes and assessments of a student's performance in a course.

Table 9: Grading information

Relative Grading Range (P)	Category	Grade (G)
≥ 9.50	Outstanding	O
≥ 8.50 to 9.49	Excellent	S
≥ 7.00 to 8.49	Very good	A
≥ 6.00 to 6.99	Good	B
≥ 5.00 to 5.99	Fair	C
≥ 4.00 to 4.99	Marginal	D
Transitional Grade	Repeat	R
Transitional Grade	Incomplete	I

8 SUPPLEMENTARY EXAMINATIONS

8.1 The supplementary examinations shall be conducted once in summer semester. Notifications will be released by the examination section informing the students about registration procedures, details of fee and timetables. Apart from these examinations the students who have courses with 'I'-grade can also write the supplementary examinations along with regular semester-end examinations of that academic (Odd / Even) semester.

8.2 Whenever a candidate clears courses with 'I' grade in a supplementary examination that are conducted during a regular semester, the Threshold value for computing his / her grade will be obtained from the same batch in which he / she had completed his / her formative assessment.

8.3 Whenever a candidate clears courses with 'R' / 'I' grade in a summer semester, the Threshold value for computing his / her grade will be carry forwarded from the preceding Odd/ Even semester for the respective courses.

8.4 Whenever a candidate clears courses with a 'R' grade in a regular semester along with his / her junior batch then for this candidate the Threshold value will be corresponding to his / her junior batch for computing grade.

8.5 The results of the summative assessment of Project / Internship will be announced only if the candidate successfully earn all the credits in courses registered during the program. If the candidate is with 'R'/ 'I' graded courses the results will be kept under 'Announced Later (AL)' status and will announced only after the candidate clear these courses.

9 GRADE POINT AVERAGE

The Academic Performance of a student is indicated every semester by the Semester Grade Point Average (SGPA) and finally by Cumulative Grade Point Average (CGPA).

9.1 SGPA

The Semester Grade Point Average (SGPA) shall be computed using the formula given below:

$$SGPA = \frac{\sum_{i=1}^{n} C_i P_i}{\sum_{i=1}^{n} C_i}$$

Where

n = number of courses a student successfully completed in the semester under consideration

Pi = Grade points secured for the ith course registered in the semester under consideration.

Ci = the number of credits assigned to ith course registered in the semester under consideration

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

9.2 CGPA

The Cumulative Grade Point Average (CGPA) shall be computed after successful completion of the programme.

The computations will be as below:

$$\text{CGPA} = \frac{\sum_{j=1}^{m} C_j P_j}{\sum_{j=1}^{m} C_j}$$

where

m = total number of courses prescribed for the completion of the programme

Cj = the number of credits assigned to the jth course

Pj = grade points secured in the jth course.

and ∑ Cj = 80

Percentage equivalence of SGPA & CGPA = (SGPA or CGPA) X10

10 AWARD OF CLASS

The students who have become eligible for award of degree shall be classified based on their CGPA secured, as per the Table (10) given below:

Table 10: Class / Division information

Sl. No.	CGPA	Class / Division
1	8.0 and above	First Class with Distinction
2	6.5 and above but less than 8.0	First Class
3	6.0 and above but less than 6.5	Second Class
4	Less than 6.0	Pass Class

a) For the purpose of rewarding the accomplishers with ranks and awards, toppers in each branch discipline are identified, based on their academic performance (CGPA) for MCA programme.

b) In addition, the 'Chairman's gold medal' and other 'Endowment Awards' are awarded to the 'outstanding students' based on the overall performance which includes academic, co-curricular and extra-curricular activities, campus placements and competitive examinations. A committee appointed by the Vice-Chancellor will recommend the eligible student for the award, selected from the nominations received from the departments.

c) In addition, the institution may recognize exceptional performance such as music, dance, sports etc. and display of exceptional bravery from time to time

d) The candidates who complete 80 credits in the first 4 successive semesters shall be eligible to receive awards / ranks

e) The candidates availing spill over semesters will not be eligible for the award of merit scholarships

11 AWARD OF DEGREE

On successful completion of prescribed requirements of the programme, the degree shall be conferred during the convocation of the VFSTR.

For the conferment of degree, the student has to fulfill the following requirements:

a) a bonafide student and undergone the course work of not less than two academic years and not more than four academic years from the date of joining.

b) successfully completed all the courses as prescribed in the respective curriculum.

c) acquired a minimum eligible credit i.e. 80 credits for the award of MCA degree.

d) obtained no due certificates as prescribed by VFSTR.

e) no in-disciplinary proceedings pending against him / her.

Consequent upon being convinced, following an enquiry, the Academic council may resolve to withdraw the degree / diploma / any other certification provided by the institute. The

aggrieved may however prefer for a review of such decision by the Academic Council, citing cogent reasons for review or go in for an appeal to the BoM of the institute.

12 LATERAL EXIT OPTIONS

12.1 Honorable exit with PG diploma

In line with NEP-2020, an optional exit is provided for a candidate who has earned a minimum number of credits and has completed all the requirements up to the end of respective years of study.

Post Graduate Diploma certification will be awarded for a candidate who has earned a minimum of 40 credits and completed all the requirements up to the end of second semester.

Semester-wise transcript and a consolidated transcript will be given along with a transit to the candidates during their exit from the registered program.

The candidate who has exited can seek re-entry to complete MCA by surrendering the graduate Diploma. A committee constituted by Vice-Chancellor will scrutiny all such re-entry requests and recommend the plan of action. However, the max. duration of programme should be limited to four years and further extension beyond the stipulated max. duration of study has to be approved by Academic Council, if the candidate appeals for an extension.

12.2 Volunteer 'Drop' with Sabbatical semester

A candidate may exercise his option to exit from MCA Degree programme temporarily for a semester, utilizing the DROP option. The DROP can be exercised to take up Internship/ Innovation / Exploratory / Entrepreneurship / Advanced research / Start-up and such related activities. Under such circumstances a candidate can normally avail DROP over two successive semesters. Such drop semester will be identified as sabbatical semester.

However, such a candidate has to pay the regular semester fee if such a Drop option is utilized during the first 4 semesters of MCA, and has to pay a nominal semester maintenance fee during the spillover period, if a candidate has not yet completed the credit requirements.

Upon returning from such a temporary exit, a candidate may continue his MCA studies utilizing the provision of spillover period. A candidate may also submit a claim for Credit equivalence for the activities undertaken during the DROP period. The equivalence committee would evaluate and assess the academic equivalence of the work carried out and would recommend the credit equivalence and credit transfer to be granted together with the grades that could be attributed, if applicable. However, the max. duration of programme should be limited to four years and further extension beyond the stipulated max. duration of study has to be approved by Academic Council, if the candidate appeals for an extension.

12.3 Volunteer 'Drop' of a Semester Drop option

A candidate may exercise his option to voluntarily exit from MCA programme temporarily for a semester during the MCA programme, utilizing the DROP option to meet the family / personal exigencies. All the norms mentioned in the section 12.2 shall be applicable for the candidates utilizing semester drop option.

13 INTERPRETATION OF RULES

a) The academic rules and regulations should be read as a whole for the purpose of any interpretation.

b) For the matter(s) NOT covered herein above or for unforeseen circumstances, but arising during the course of the implementation of the above regulations. The Vice-Chancellor shall be authorized to remove the difficulties and decide upon the matters. The same shall be reported in the next meeting of Academic Council for ratification and subsequently informed to BoM.

c) The Institution may change or amend the academic rules and regulations or curriculum at any time, and the changes or amendments made shall be applicable to all the students with effect from the dates, notified by the Institution.

d) Procedure and explanation to any section can be floated by the office of Dean AAA as applicable from time to time with due approval by the chairman of Academic Council.

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

ANNEXURE -1

MASTER OF COMPUTER APPLICATIONS

I Year I Semester

Course Code	Course Title	L	T	P	C
22MC101	Data Structures	3	-	2	4
22MC102	Probability and Statistics	2	2	-	3
22MC103	Database Systems	3	-	2	4
22MC104	Computer Organization and Operating systems	3	2	-	4
22MC105	Object oriented Programming with Java	3	-	2	4
22MC106	Python Programming	2	-	2	3
22MC107	Technical English communication	1	-	2	2
	Total	**17**	**4**	**10**	**24**
		31 hrs			

I year II Semester

Course Code	Course Title	L	T	P	C
22MC108	Software Engineering and Testing	3	-	2	4
22MC109	Web Technologies	3	-	2	4
22MC110	Computer Networks	2	2	-	3
	Elective – I	3	-	2	4
22MC111	Data Mining Techniques	3	-	2	4
22MC112	Data Visualization	-	-	4	2
22MC113	Socially relevant project using Design Thinking	-	-	4	2
22MC114	Soft skills laboratory	-	-	2	1
	Total	**14**	**2**	**18**	**24**
		34 hrs			

II year I Semester

Course Code	Course Title	L	T	P	C
22MC201	Cryptography and Network security	3	-	2	4
22MC202	Organization Behavior	2	2	-	3
22MC203	Cloud Computing	2	2	-	3
22MC204	Internet of things	2	2	-	3
	Elective – II	2	2	-	3
	Elective – II (MOOCs-1 (NPTEL/SWAYAM)/ Dept. Elective	3	-	2	4
	Total	**15**	**6**	**8**	**20**
		29 hrs			

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

II year II Semester

Course Code	Course Title	L	T	P	C
22MC205 / 22MC206	Internship / Project	-	2	22	12
	Total		**24**	**12**	

List of Department Elective

Course Code	Course Title	L	T	P	C
22MC801	Big Data Analytics	2	2	-	3
22MC802	Digital Image Processing	2	2	-	3
22MC803	Mobile Application Development	2	2	-	3
22MC804	Software Project Management	2	2	-	3
22MC805	Formal Languages and Automata Theory	3	2	-	4
22MC806	Design and Analysis of Algorithms	3	-	2	4
22MC807	Data Science using Python	3	-	2	4
22MC808	Machine Learning through Python	3	-	2	4
22MC809	Deep Learning	3	-	2	4
22MC810	Full Stack Technologies	3	-	2	4
22MC811	Medical Image Analysis	3	-	2	4
22MC812	Block Chain Technology	3	2	-	4
22MC813	Data Wrangling and Visualization	2	2	2	4

R22 MCA
2 YEAR
POST GRADUATE PROGRAMME

ANNEXURE – 2

SUPPLEMENT REGULATION FOR ONWARD CONTINUATION TO M.TECH DEGREE

The proposal to institute continuation to M. Tech degree after MCA is designed in line with the practices in Institutions of National Importance, is to elevate the Gross Qualification Index (GQI) of India, and specifically to enhance the technological / engineering competency of the Technocrats.

In order to attract the committed learners towards earning M.Tech degree immediately after MCA the following scheme is proposed:

1. He / she should be a candidate maintaining his studentship through proper registration process of his MCA program and fulfills all the credit requirements of MCA.
2. He / she has to appear for an aptitude test / interaction and the corresponding committee of experts constituted for the purpose has to recommended his/ her name.
3. The regulations that were in force during that Academic year will be applicable for his / her continuation M.TECH programme.
4. The candidate should earn all the credits pertaining to MCA programme before commencing the M.TECH programme. The program structure designed for M.Tech will be followed from V semester onwards.
    - MCA degree = 80 credits
    - M. Tech degree = 64 credits

FOREWORD

In the last few decades, there has been a significant development in the IT sector. Due to the innovative products and services provided, the industry has flourished and many new job opportunities have been generated. Students who want to work in the emerging field of Information Technology must be knowledgeable in both current and emerging fields of study.

Master of Computer Application is a 2-year postgraduate degree programme designed to give students practical knowledge of Computer Applications and software development. It delves deeply into the topic and equips students to deal with the rapidly evolving trends in Computer Science and Technology.

The MCA programme provides advanced knowledge and abilities in a number of programming languages, which are necessary for more effective and efficient application development. The top-notch curriculum helps students build the application skills and talents that will help them launch their careers in corporate roles in the software industry.

R22 curriculum comprises of:

- Six months Industry Internship to make students industry ready.
- Introduced socially relevant projects using design thinking
- Revision in tune with National Education Policy 2020.
- Various exit options.
- More emphasis on Problem solving skills.
- Enhanced skill based courses for improving employment opportunities.
- Advanced courses like Artificial Intelligence, Machine learning, Artificial Neural Networks, Deep Learning, Cloud Computing, Big Data Analytics and Internet of Things.
- Practice sessions embedded into as many courses as possible.

In R22 curriculum, every care has been taken to accommodate the knowledge and skill requirements of industry through practicing of proper activities. While making the graduates work ready, it also enables them to be successful in competitive examinations.

The Board of Studies consisting eminent personalities along with experienced faculty membersof the university have designed the curriculum to offer knowledge and skill of informationtechnology on the above mentioned areas. The curriculum includes concepts with skill basedtasks through integrated laboratory and activities combined with theory. The department aims tomake graduates ready for the industrial needs.

External BoS Members:

1. Dr.C.Raghavendra Rao, Professor, University of Hyderabad, Hyderabad
2. Dr. R.B.V. Subramaanyam, Professor & HOD, Department of CSE, NIT Warangal
3. Dr. B. Ramesh Babu, Sr. Asst. Professor. Department of CSE, MNIT Jaipur.

I thank all the BOS Members, Academic Council Members and University authorities especially Dean AAA and honourable Vice Chancellor for encouraging and supporting us in designing this innovative curriculum for MCA students.

Dr. K. V. Krishna Kishore
Head, Department of IT
VFSTR Deemed to be University

VIGNAN'S
Foundation for Science, Technology & Research
(Deemed to be University)
-Estd. u/s 3 of UGC Act 1956

VISION

To evolve into a Centre of Excellence in Science &Technology through creative and innovative practices in teaching-learning, towards promoting academic achievement and research excellence to produce internationality accepted, competitive and world class professionals who are psychologically strong & emotionality balanced, imbued with social consciousness & ethical values.

MISSION

To Provide high quality academic programmes, training activities, research facilities and opportunities supported by continuous industry-institute interaction aimed at promoting employability, entrepreneurship, leadership and research aptitude among students and contribute to the economic and technological development of the region, state, and nation.

Department of INFORMATION TECHNOLOGY

VISION

To become centre of excellence in technical and knowledge-based education utilizing the potential of emerging technologies in field of Information Technology with a deep passion of wisdom, culture, and values.

MISSION

M1: Impart modern teaching methodologies to provide quality education to the students

M2: Produce employable engineers based on skills required for industry

M3: Enable students and faculty members in research to develop IT based solutions to meet societal and industry requirements.

MASTER OF COMPUTER APPLICATOINS

Program Educational Objectives (PEOs)

Graduates of Master of Computer Applications should be able to,

PEO1: To prepare the graduates as successful professionals ready for Industry, Government sectors, Academia, Research, Entrepreneurial Pursuit and Consultancy firms.

PEO2: To prepare the graduates with Ethical Attitude, Effective Communication Skills

PEO3: To prepare the graduates with excellent computing ability so that to Comprehend, Analyse, Design and Create computing solutions for the real time problems

Program Specific Outcomes (PSOs)

MCA Graduates will be able to:

PSO1: Ability to pursue careers in IT industry/consultancy/research and development teaching and allied areas related to computer Science.

PSO2: Comprehend, explore and build up computer programs in the areas allied to Algorithms, Systems Software, Multimedia, Web Design and Big Data Analytics for efficient design of computer-based systems of varying complexity.

Program Outcomes (POs)

PO1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2. Problem analysis: Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and

synthesis of the information to provide valid conclusions.

PO5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

PO6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12. Life-long learning: Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

R22 CURRICULIM

(Applicable for students admitted into First year from academic year 2022-23)

MASTER OF COMPUTER APPLICATIONS

I YEAR I SEMESTER

I YEAR I SEMESTER						
S.No.	**Course code**	**Course Title**	**L**	**T**	**P**	**C**
1	22MC101	Data Structures	3	-	2	4
2	22MC102	Probability and Statistics	2	2	-	3
3	22MC103	Database Systems	3	-	2	4
4	22MC104	Computer Organization and operating systems	3	2	-	4
5	22MC105	Object oriented Programming with Java	3	-	2	4
6	22MC106	Python Programming	2	-	2	3
7	22MC107	Technical English communication	1	-	2	2
		Total	**17**	**4**	**10**	**24**
			31 hrs			

I YEAR II SEMESTER						
S.No.	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	22MC108	Software Engineering and Testing	3	-	2	4
2	22MC109	Web Technologies	3	-	2	4
3	22MC110	Computer Networks	2	2	-	3
4		Elective – I	3	-	2	4
5	22MC111	Data Mining Techniques	3	-	2	4
6	22MC112	Data Visualization		-	4	2
7	22MC113	Socially relevant project using Design Thinking	-	-	4	2
8	22MC114	Soft skills laboratory			2	1
		Total	**13**	**4**	**16**	24
			33 hrs			

R22 CURRICULIM

(Applicable for students admitted into First year from academic year 2022-23)

MASTER OF COMPUTER APPLICATIONS

II YEAR I SEMESTER						
S.No.	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	22MC201	Cryptography and Network security	3	-	2	4
2	22MC202	Organization Behaviour	2	2	-	3
3	22MC203	Cloud Computing	2	2	-	3
4	22MC204	Internet of things	2	2	-	3
5		Elective – II	2	2	-	3
6		Elective – III (MOOCs-1 (NPTEL/SWAYAM)/ Dept. Elective	3	-	2	4
		Total	**15**	**6**	**8**	20
			29 hrs			

II YEAR II SEMESTER						
S.No.	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	22MC205/ 22MC206	Internship/ Project	-	2	22	12
		Total		24		12

R22 CURRICULIM

(Applicable for students admitted into First year from academic year 2022-23)

MASTER OF COMPUTER APPLICATIONS

DEPARTMENT ELECTIVES						
S.No.	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	22MC801	Big Data Analytics	2	2	-	3
2	22MC802	Digital Image Processing	2	2	-	3
3	22MC803	Mobile Application Development	2	2	-	3
4	22MC804	Software Project Management	2	2	-	3
5	22MC805	Formal Languages and Automata Theory	3	2	-	4
6	22MC806	Design and Analysis of Algorithms	3	-	2	4
7	22M807	Data Science using Python	3	-	2	4
8	22MC808	Machine Learning through python	3	-	2	4
9	22MC809	Deep Learning	3	-	2	4
10	22MC810	Full stack technologies	3	-	2	4
11	22MC811	Medical Image analysis	3	-	2	4
12	22MC812	Block Chain Technology	3	2	-	4
13	22MC813	Data Wrangling and Visualization	2	2	2	4

22MC101-DATA STRUCTURES

Hours per week:5

L	T	P	**C**
3	0	2	**4**

PREREQUISITE KNOWLEDGE: C Programming.

COURSE DESCRIPTION AND OBJECTIVES:

This course is aimed at offering fundamental concepts of programming language to the students. It starts with the basics of C-programming and deals with the structure and various attributes required for writing a 'C' program. It also introduces various operators and control statements used in programming. Then it switches to functions and arrays. It goes on with strings, pointers, files & the user defined data types. As a first-level course in computer science, it forms the basis to understand usage of various attributes in writing a program.

MODULE – 1

UNIT – 1 **12L+0T+0P=12 Hours**

BASIC PROGRAMMING AND INTRODUCTION

Structure of C program – Basic Data types, Operators Control statements, functions and Arrays

UNIT – 2 **12L+0T+16P=28 Hours**

INTRODUCTION TO STRINGS, POINTERS AND STRUCTURES

Strings, pointers, structures, sorting and searching.

PRACTICES:

- Compute the factors of a number.
- Compute the average of 'n' numbers.
- Find whether a number is palindrome or not.
- Find whether a number is a power of 2 or not.
- Compute the factorial of a number.
- Swap two values using call by value and call by reference.
- Find the frequency of each number in the array.
- Reverse the contents of the array.
- Find the factorial of a number using recursion.
- Access the structure and union members.
- Quick, Merge, Heap and Radix sorting techniques.
- Linear and Binary search algorithms.

MODULE – 2

UNIT – 1 **12L+0T+0P=12 Hours**

LINKED LISTS

Introduction, Types of Linked List - Singly Linked List, Doubly Linked List, Circular Linked List; Operations - Insertion, Deletion, Traverse forward/reverse order. STACKS AND QUEUES: Stacks - Introduction, Array and Linked representations, Implementation; Queues - Introduction, Array and Linked representations, Implementation, Circular queue.

UNIT – 2 **12L+0T+16P=28 Hours**

TREES

Introduction, Properties, Binary Tree - Introduction, Properties, Array and Linked representations; Tree traversals and their Implementation, Expression trees, BST Definition and implementation. GRAPHS: Introduction, Properties, Modeling problems as graphs representations - Adjacency matrix, Adjacency list; Traversals - Breath first search and Depth first search.

PRACTICES:

- Singly linked list, doubly linked list and circular linked list.
- Stack using an array and linked list.
- Queue using an array and linked list.
- Tree using an array and linked list.
- Check if given expression is fully parenthesis or not using stack.
- Tree traversing techniques.
- BST using an array and linked list.
- Graph traversal techniques.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Explore the organization of several ADTs and the manipulation (searching, insertion, deletion, traversing) of data stored in various data structures.	Apply	1,2	1
2	Apply different data structures to solve a given problem	Apply	1,2	1
3	Analyze the efficiency of using different data structures and choose the efficient data structure for solving a given problem.	Analyze	1,2	2
4	Develop new algorithms to solve various problems.	Create	1,2	3,4

TEXT BOOKS:

1. Ajay Mittal, "Programming in C - A practical Approach", 1st edition, Pearson Education India, 2015.
2. Reema Thareja, "Introduction to C Programming", 2nd edition, Oxford University Press India, 2015.
3. Reema Thareja, "Data Structures Using C", 2nd edition, Oxford University Press, 2014.

REFERENCE BOOKS:

1. Herbert Schildt, C, "The Complete Reference", 4th edition, Tata McGraw-Hill, 2000.
2. E. Balagurusamy, "Programming in ANSI C", 4th edition, Tata McGraw- Hill, 2008.
3. Richard F. Gilberg and Bhrouz A. Forouzan, "Data Structures: A Pseudocode Approach with C", 2nd edition, Cengage Learning, 2004.
4. Jean Paul Tremblay and Paul G. Sorenson, "An Introduction to Data Structures with Applications", 2nd edition, Tata Mc-Graw Hill, 2004.
5. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", 2nd edition, Pearson Education, 2006.

22MC102-PROBABILITY AND STATISTICS Hours per week:4

L	T	P	C
2	2	0	**3**

PREREQUISITE KNOWLEDGE: Basic knowledge in statistics and mathematics.

COURSE DESCRIPTION AND OBJECTIVES:

This course deals with descriptive statistics, correlation, regression, and their applications, probability, theoretical distributions and testing of hypothesis. It also enables the student to understand and apply statistical techniques, curve fitting, correlation and regression, probability and also to make the student familiar with discrete, continuous distributions and testing of hypothesis.

MODULE – 1

UNIT – 1 **8L+8T+0P=16 Hours**

DESCRIPTIVE STATISTICS

Basic Definitions, Frequencies, Graphical Representation, Histogram, Ogive curves; Measures of Central tendency, Arithmetic mean, Median, Mode, Mean deviation, Standard deviation; Symmetry and Skewness, Karl Pearson's Coefficient of skewness.

UNIT – 2 **8L+8T+0P=16 Hours**

PROBABILITY AND RANDOM VARIABLES

Probability: Introduction, Definition (Classical and Axiomatic approach), Addition theorem, Conditional probability, Multiplication theorem and Bayes theorem.

Random Variables: Random variables, Discrete and Continuous variables and distribution function.

PRACTICES:

- Various graphical presentation techniques
- Measures of central tendency
- Skewness
- Karl Pearson's coefficient of skewness
- Definitions of probability
- Applications of addition theorem
- Applications of multiplication theorem
- Applications of conditional probability
- Random variables and types of random variables
- Distribution function

MODULE – 2

UNIT – 1 **8L+8T+0P=16 Hours**

REGRESSION ANALYSIS AND DISTRIBUTIIONS

Correlation and regression: Correlation, Types, Pearson's Coefficient of correlation, Regression, Regression lines.

Distributions: Introduction to Distributions: Binomial, Poisson and Normal distributions with properties and applications.

UNIT – 2 **8L+8T+0P=16 Hours**

TESTING OF HYPOTHESIS

Testing large samples-one mean, two means, one proportion and two proportions. Testing small samples- one mean, two means (independent and paired samples), Chi square tests-goodness of fit and independence of attributes.

PRACTICES:

- Correlation
- Types of correlation
- Karl Pearson's coefficient of correlation
- Regression and regression lines
- Binomial distribution
- Poisson distribution
- Normal distribution
- Testing the large sample tests-one mean and two sample means
- One proportion and two proportion tests
- Testing small samples-one, two samples and paired tests
- Chi-square test for goodness of fit
- Chi-square test for independence of attributes

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Apply measures of central tendency, skewness, and Karl Pearson's coefficient of skewness to study the statistical data sets.	Apply	1	1,2
2	Apply the probability theory and their applications to measure the uncertainty.	Apply	1	1,2
3	Study the relations between statistical variables and can fit the mathematical models for association	Analyze	2	1,2,3
4	Test the statistical significances for various samples.	Evaluate	2	1,2,4
5	Understand the small sample tests and chi-square tests	Evaluate	**2**	1,4,5

TEXT BOOKS:

1. Sheldon M. Ross, An Introduction to Probability and Statistics for Engineers and Scientists, 3 rd Edition, Academic Press, Elsevier, 2010.
2. S. C. Gupta and V. K. Kapoor, "Fundamentals of Mathematical Statistics", Sultan Chand &Sons, 2012.

REFERENCE BOOKS:

1. P. R. Vittal, "Mathematical Statistics", Margham Publications, Chennai, 2018.
2. Kishore S. Trivedi, "Probability and Statistics with Realiability, Queueing and Computer Science Applications", 2nd edition, Wiley Student edition, 2008.
3. A. Singaravelu, "Probability and Statistics", 22nd edition, Meenakshi Agency, 2015..

22MC103-DATABASE SYSTEMS Hours per week:5

L	T	P	**C**
3	0	2	**4**

PREREQUISITE KNOWLEDGE: Computer Programming and Data Structures.

COURSE DESCRIPTION AND OBJECTIVES:

This course presents an introduction to database management systems with an emphasis on how to organize, maintain and retrieve data efficiently from a relational database. It also focuses on requirements gathering and conceptual, logical, physical database design. The objective of the course is to enable the student to understand database design, expressing queries using SQL, transaction processing.

MODULE – 1

UNIT – 1 **12L+0T+8P=20 Hours**

DATABASES AND DATABASE USERS

Introduction; Characteristics of the database approach; Actors on the scene; Advantages of using the DBMS approach.

DATABASE SYSTEM CONCEPTS AND ARCHITECTURE

Data models, Schemas, and instances; Three-Schema architecture and data Independence; Database languages and interfaces; The database system environment; Centralized and Client-Server architectures for DBMS.

CONCEPTUAL DATA MODELING AND DATABASE DESIGN

Entity types, Entity sets, Attributes, and keys; Relationship types, Relationship Sets, Roles, and structural constraints; Weak entity types; Relationship types of degree higher than two.

THE RELATIONAL DATA MODEL AND RELATIONAL DATABASE CONSTRAINTS

Relational model concepts; Relational model constraints and Relational database schemas

BASIC SQL: SQL data definition and data types; Specifying constraints in SQL, Basic retrieval queries in SQL

UNIT – 2 **12L+0T+8P=20 Hours**

MORE SQL

COMPLEX QUERIES, TRIGGERS, VIEWS: More complex SQL retrieval queries; Views (virtual tables) in SQL , Introduction to PL/SQL Procedures, Functions Specifying constraints as assertions and actions as triggers.

PRACTICES:

- Design Conceptual database schema using ER Modeling Software Tools.
- Development of Relational Database schemas for Company/Student/Sailors/ using DDL
- constructs of SQL.
- Specifying various DML Commands such as select, insert, update etc. of SQL on Relational Database.
- Specifying various DCL and TCL constructs of SQL on Relational Database.
- Development of Relational Database schemas by specifying different types of Constraints

- Specifying queries using Relational Database operators (Arithmetic, Logical & comparison)
- and string matching constructs of SQL.
- Expressing queries using Aggregate Functions of SQL on Relational Database.
- Queries on Relational Database using GROUP BY, HAVING and ORDER BY clauses of SQL.

MODULE – 2

UNIT – 1 **12L+0T+8P=20 Hours**

BASICS OF FUNCTIONAL DEPENDENCIES AND NORMALIZATION FOR RELATIONAL DATABASES

Informal design guidelines for relation schemas; Functional dependencies-inference rules, equivalence and minimal cover; Normal forms based on primary keys; Boyce-Codd normal form; multivalued dependency and 4NF; Join dependencies and 5NF; Properties of relational decompositions.

UNIT – 2 **12L+0T+8P=20 Hours**

INTRODUCTION TO TRANSACTION PROCESSING CONCEPTS AND THEORY

Introduction to transaction processing; Transaction and system concepts; Desirable properties of transactions; Characterizing schedules based on serializability.

CONCURRENCY CONTROL TECHNIQUES

Two-phase locking techniques for concurrency control; Concurrency control based on timestamp ordering.

PRACTICES:

- Design and Development of company database and expressing Nested queries using SQL.
- Design and Development of sailors database and specifying queries using different types of JOINs.
- Creation and dropping of VIEWS.
- Implementation of PL/SQL programs with Control Structures.
- Implementation of PL/SQL programs with Procedures.

Case study on Transaction Processing and Concurrency Control

I. Transactions & TPS Activities in Indian Railways

A**) Passenger Reservation System**

- Checking current position on any train
- Make a reservation from any origin to any destination in India
- Pay for the ticket using any option like debit card, credit card, etc
- Changing/ Cancellation of bookings and getting money refund

B) Freight Operation Information System

- Checking real-time position of goods trains
- Checking number of rakes being used and number of idle rakes
- Volume and weight of goods being carried

C) Crew Management System

- Monitoring of activities of crew members
- Checking the location, time of arrival and waiting time of crew members

- Maintaining records of payment schedule information, duty schedule, training schedule of the crew members

D) Integrated Coach Management System

- Checking real-time location and movement of trains
- Scheduling trains movement near bottlenecks and heavy traffic zones
- Improving train arrival time accuracy

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Devise queries using SQL Subsets.	Apply	1	1,2,3,4,5,9,10,12
2	Evaluating Transaction Techniques	Analyze	1, 2	1,2,3,4,5,9,10,12
3	Express queries using database tools like Oracle, MYSQL.	Analyze	2	1,2,3,4,5,9,10,11
4	Develop an E-R model for real life applications	Create	1	1,2,3,4,5,9,10,11, 12
5	Design and normalize databases for real time applications.	Create	1, 2	1,2,3,4,5,9,10,11. 12

TEXT BOOKS:

1. Ramez, Elmasri and Shamkant B. Navathe, “Fundamentals of Database Systems”, 7 th Edition, Pearson Education, 2016.
2. Hector Garcia Molina, Jeffrey Ullman, Jennifer Widom, “Database Systems: The Complete Book”, 2 nd Edition, Pearson Prentice Hall, 2009.

REFERENCE BOOKS:

1. Raghu Rama Krishnan and Johannes Gehrke, “Database Management Systems”, 3 rd Edition, Tata McGraw Hill, 2013.
2. Abraham Silberschatz, Henry F.Korth and S.Sudarshan, “Database System Concepts”, 6 th Edition, Tata McGraw Hill, 2010.
3.Corlos Coronel, Steven Morris, Peter Robb, “Database Principles Fundamentals of Design Implementation and Management”, 1 st Edition, Cengage Learning, 2014.

22MC104-COMPUTER ORGANIZATION AND OPERATING SYSTEMS

Hours per week:5

L	T	P	C
3	2	0	**4**

PREREQUISITE KNOWLEDGE: Basics of Computers, Digital Electronics, Data structures, Programming.

COURSE DESCRIPTION AND OBJECTIVES:

This course aims at modern Computer Organization and Architecture. The emphasis is on understanding the design of computer and its components. The student will learn the concepts of data representation, micro operations, memory organizations and input output organization. Case study of 8086 helps the students to visualize the basic concepts of the course and also it covers the concepts and principles of Operating Systems, its overall responsibility in acting as an interface between the system's hardware components and the user/application software. Further, it also helps students to understand the different scheduling policies, process synchronization mechanisms, deadlock handling mechanisms, memory management techniques and file management system.

MODULE – 1

UNIT – 1 **12L+8T+0P=20 Hours**

FUNDAMENTAL CONCEPTS

Register Transfers, Bus and Memory Transfers, Logic Micro operations, Shift Micro operations, Arithmetic Logic Shift Unit.

BASIC COMPUTER ORGANIZATION AND DESIGN

Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory-Reference Instructions, Input-Output and Interrupt.

CENTRAL PROCESSING UNIT

Register Organization, Stack Organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control.

UNIT – 2 **12L+8T+0P=20 Hours**

INPUT-OUTPUT ORGANIZATION

Input-Output Interface, Asynchronous Data Transfer, Modes of Transfer, Priority Interrupt, Direct Memory Access.

MEMORY ORGANIZATION

Associative memory, Cache memory, Virtual Memory.

MODULE – 2

UNIT – 1 **12L+8T+0P=20Hours**

INTRODUCTION TO OPERATING SYSTEMS

Introduction to Operating System Concept: Types of Operating Systems, Operating Systems Concepts, Operating System Operations.

PROCESS MANAGEMENT

Process concept, Process State Diagram, Process control block, Process Scheduling, Inter process Communication, Threads- Threading Issues, Scheduling- Basic Concepts, Scheduling Criteria, Scheduling Algorithms.

PROCESS SYNCHRONIZATION

The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Principles of deadlock: System Model, Deadlock characterization, Deadlock handling, Deadlock Prevention, Detection and Avoidance, Recovery Starvation, Critical Regions form Deadlock.

UNIT – 2 **12L+8T+0P=20 Hours**

MEMORY MANAGEMENT

Swapping, Contiguous Memory Allocation, Paging, structure of the Page Table, Segmentation Virtual Memory Management- Demand Paging, Page-Replacement Algorithms, Thrashing. File-System Interface: File Concept, Access Methods, Directory structure, File-System mounting, Files Sharing, Protection. File-System implementation- File-System Structure, Allocation Methods, Free-Space Management, Disk Structure, Disk Scheduling.

PRACTICES:

- Simulate the Following CPU Scheduling Algorithms
- A) FCFS B) SJF C) Priority D) Round Robin
- Multiprogramming-Memory Management- Implementation of fork(), wait(), exec() and exit()
- Simulate The Following a. Multiprogramming with A Fixed Number Of Tasks (MFT) b. Multiprogramming with A Variable Number Of Tasks (MVT)
- Write a program to implement first fit, best fit and worst fit algorithm for memory management.
- Simulate Bankers Algorithm for Dead Lock Avoidance
- Simulate Bankers Algorithm for Dead Lock Prevention.
- Simulate The Following Page Replacement Algorithms.
- FIFO b) LRU c) LFU
- Simulate the Following File Allocation Strategies
- Sequenced b) Indexed c) Linked

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Identify different pipeline hazards and apply pipelined data and instruction paths for a given instruction set architecture.	Apply	1	1,2,3,4,9,10,12
2	Classify the organization and architecture of computer systems, analyse the performance of memory, I/O and pipelined systems. Develop different digital circuits required to perform the micro operations.	Analyze	1	1,2,3,4,9,10,12

3	Apply the concepts of process scheduling algorithms and process synchronization techniques to derive the efficiency of resource utilization.	Apply	2	1,2,3,4,9,10,12
4	Design the various memory management schemes and file system structure for a given scenario	Create	1,2	1,2,3,4,9,10,12

TEXT BOOK:

1. Abraham Silberschatz Peter B. Galvin and Greg Gagne, "Operating System Concepts", 9th Edition, Wiley, 2013.

REFERENCE BOOKS:

1. Garry. J. Nutt, "Operating Systems: A Modern Perspective", 3 rd Edition, Addison-Wesley, 2016.
2. Andrew S. Tanenbaum and Herbert Bros, "Modern Operating Systems", 4 th Edition, Pearson, 2015.
3. William Stallings, "Operating Systems: Internals and Design Principles", 7 th Edition, Prentice Hall of India, 2013.
4. Dhananjay M. Dhamdhere, "Operating Systems: A Concept-Based Approach", 3 rd Edition, Tata McGraw-Hill, 2017..

22MC105-OBJECT ORIENTED PROGRAMMING Hours per week:5

L	T	P	**C**
3	0	2	**4**

PREREQUISITE KNOWLEDGE: Computer Programming.

COURSE DESCRIPTION AND OBJECTIVES:

This course is about the fundamentals of Object-Oriented Programming (OOP) Concept and OOP-based software development methodology. Java as a class-based and pure OOP language is used to demonstrate and implement appropriate concepts and techniques. The students are exposed to the concepts, fundamental syntax, and the thought processes behind object oriented programming. By end of the course, students will acquire the basic knowledge and skills necessary to implement object-oriented programming techniques in software development using Java.

MODULE – 1

UNIT – 1 **12L+0T+8P=20 Hours**

INTRODUCTION TO OOP

Java buzzwords, OOP principles, Data types, Operators, Control statements, Type conversion and casting, Arrays. String class, String Tokenizer

UNIT – 2 **12L+0T+8P=20 Hours**

CLASSES AND METHODS

Introduction to classes and methods, objects, Constructors, Overloading Methods and Constructors, Usage of static, Access control, this key word, Polymorphism, Method overriding

PRACTICES:

- Reading different types of data from the user and display that data using Scanner class.
- Illustrating type conversions.
- Implementing different operators.
- Generating electricity bill
- Implementing different patterns.
- Implementing logical programs.
- Implementing Arrays.
- Implementing String class.
- Implementing String Tokenizer class.
- Implementing constructor overloading using this keyword
- Implementing method overloading and overriding.

MODULE – 2

UNIT – 1 **12L+0T+8P=20 Hours**

INHERITANCE

Basics of Inheritance, Types of inheritance, Abstract classes, Interfaces, Usage of final, Usage of super key word,

INTERFACES

Implementing interface, extending interfaces, creating, defining and accessing Packages, importing packages.

UNIT – 2 **12L+0T+8P=20 Hours**

EXCEPTION HANDLING

Concepts of exception handling, Types of exceptions, Built-in exceptions, Usage of try, catch, throw, throws and finally keywords

MULTITHREADING

Concepts of Thread, Thread priorities, multithreading, Daemon thread, Synchronization.

PRACTICES:

- Implementing super keyword.
- Implementing forms of Inheritance
- Implementing runtime polymorphism.
- Create an abstract class Media (id, description). Derive classes Book (page count) and CD (play time). Define parameterized constructors. Create one object of Book and CD each and display the details.
- Define an interface, operations which has method area (), volume (). Define a constant PI having value 3.14. Create class a Cylinder which implements this interface (member-id, height). Create one object and calculate area and volume.
- Implementing packages.
- Implementing Exception handling
- Implement java program which accepts withdraw amount from the user and throws an exception "In Sufficient Funds" when withdraw amount more than available amount.
- Creating Thread.
- Implementing multithreading.
- Create three threads and that displays "good morning", for every one second ,"hello" for every 2 seconds and "welcome" for every 3 seconds by using extending Thread class.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Analyze Object oriented concepts and JVM	Analyze	1	1,2,3
2	Apply inheritance and polymorphism	Apply	1	1, 2, 3
3	Apply packages and interfaces to develop real time applications	Apply	1	1, 2, 3
4	Develop Interfaces and Packages.	Create	2	1, 2, 3,4, 5
5	Design multithreaded applications	Create	2	1, 2, 3,4, 5

TEXT BOOKS:

1. Herbert Scheldt, "Java the complete reference", 12th Edition, McGraw Hill, Education, 2021.
2. P. Radha Krishna, "Object Oriented Programming through Java", 1st Edition, Universities Press, 2007.

REFERENCE BOOKS:

1. J. Nino and F.A. Hosch, “An Introduction to programming and OO design using Java”, 3rd Edition, John Wiley & sons, 2008.
2. P. Radha Krishna, “Object Oriented Programming through Java”, 1st Edition, Universities Press, 2007.
3. R. A. Johnson, “Java Programming and Object Oriented Application Development”, 1st Edition, Cengage Learning, 2006.

22MC106-PYTHON PROGRAMMING Hours per week:4

L	T	P	**C**
2	0	2	**3**

PREREQUISITE KNOWLEDGE: Computer Programming.

COURSE DESCRIPTION AND OBJECTIVES:

This course offers sufficient knowledge required to understand the fundamental concepts of Python programming language. This course enables students to choose appropriate data structures (lists, dictionaries, tuples, sets, strings) for the given problem. In addition, the students will be able to create reliable, modular and reusable applications using Object-Oriented Programming approaches. At the end they will get an idea of how to access database using python programming, develop web applications, and using web Services using python programming.

MODULE – 1

UNIT – 1 **8L+0T+8P=16 Hours**

INTRODUCTION TO PYTHON

Python Installation and Working of it, get familiar with python variables and data types, Operator understanding and its usage, detail study of python blocks, Hands on with conditional blocks using if, else and elif

UNIT – 2 **8L+0T+8P=16 Hours**

VARIETIES OF DATA STRUCTURES

Hands on string handling and looping with range, list, Tuples, Sets and dictionaries. hands on to organize python code with function, modular approach in python.

PRACTICES:

1. a) Installation of python and relevant packages in windows.
   b) Installation of python and relevant packages in Linux.
   c) Practice Execution of python statements in REPL (shell).
2. a) Implement a python program to display all the python keywords and display each of them in separate lines.
   b) Develop a python program to read two integers and perform all possible arithmetic operations on those two numbers.
3. a) Develop a program to accept three numbers as command line arguments and find biggest, smallest and average of those three numbers.
   b) Implement a python program to find first n Prime Numbers.
   c) Implement a program that prints the decimal equivalents of 1/2, 1/3, . . . ,1/n.
   d)Implement a python program to read n and find sum of even and odd numbers.
4. Write python code to achieve the following
   a) to remove vowels in the given string using control transfer statements.
   b) to count number of uppercase and lowercase letters in the given string.
   c) to remove all punctuation characters from given string.
5. Implement python code to illustrate the following on Lists and Tuples
   a) Creation b) Accessing elements c) apply operators d) Usage of different methods
6. Implement python code to illustrate the following on Sets and Dictionary
   a) Creation b) Accessing elements c) apply operators d) Usage of different

methods

7. a) Implement python code to illustrate the following
   i) Positional arguments ii) Keyword arguments
   iii) Default arguments iv) Variable length arguments
   b) Implement a function to find nth Fibonacci number.
   c) Develop a recursive function to find the factorial of a given number.
   d) Implement function to compute GCD, LCM of two numbers (use Lambda function).

MODULE – 2

UNIT – 1 **8L+0T+8P=16 Hours**

EXCEPTION HANDLING

Handling and helping file operations, coding with the exceptional handling

UNIT – 2 **8L+0T+8P=16 Hours**

OBJECT-ORIENTED PROGRAMMING

Object-Oriented Programming, Classes and working with instances, Method overloading, Polymorphism

PRACTICES:

1. a) Develop a python code to handle the following built-in exceptions
   i) Value Error ii) Zero Division Error iii) Type Error iv) Name Error
   b) Implement python code to handle multiple exceptions.
   c) Implement Python code to raise an exception.
2. a) Implement python code to read contents of a file and write the contents to another file.
   b) Create a class called Student and perform operations such as display, calculate percentage, add, delete and modify student data.
   c) Design python code to depict the following oops concepts:
   i) Data hiding ii) Inheritance iii) Overriding
3. Develop python code to calculate the following statistical parameters using python 'numpy'.
   a) Mean b) Hormonic Mean c) Median d) Mode
   e) Standard Deviation f) Variance g) Percentile
4. Design python code to illustrate the following plots using 'matplotlib' package
   a) Line plot b) Bar plot c) Histogram d) Scatter Plot
5. Implement python program for the following problems on Pandas DataFrame
   a) Write a Pandas program to create and display a DataFrame from a specified dictionary data which has the index labels.
   Sample Python dictionary data and list labels:
   exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael','Matthew', 'Laura', 'Kevin', 'Jonas'],
   'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
   'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
   'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}
   Labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']
   b) Write a Pandas program to select the 'name' and 'score' columns from the following Data Frame.
   c)Write a Pandas program to select the specified columns and rows from a given

data frame.

d) Write a Pandas program to select the rows where the number of attempts in the examination is greater than 2.

e) Write a Pandas program to count the number of rows and columns of a Data Frame.

f) Write a Pandas program to change the name 'James' to 'Adhvik' in name column of the Data Frame.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	**Experiment with** the basic terminology used in computer programming to write, compile and debug programs in python language.	Apply	1	1
2	**Make use of** different data types to design programs involving decisions, loops, and functions..	Apply	1	1, 2
3	**Apply** functional, reliable and user-friendly python programs for a given problem application.	Apply	1	3
4	**Analyze** the usage of different data structures for practical and contemporary applications which uses data stored in files.	Analyze	1,2	1, 2, 4
5	**Develop** solutions using the concepts of object oriented programming paradigm.	Apply	2	2

TEXT BOOKS:

1. Reema Thareja, "Python Programming: Using Problem Solving Approach", Oxford University Press, 2019.

REFERENCE BOOKS:

1. Kenneth A. Lambert, "The Fundamentals of Python: First Programs", Cengage Learning, 2011.
2. Allen B. Downey, "Think Python" 1st edition, Orielly publishing,2015.
3. James Payne, "Beginning Python using Python 2.6 and Python 3", Wrox publishing.
4. Vamsi Kurama, "Python Programming: A Modern Approach", 1st Edition, Pearson Publishers,2018.

22MC107-TECHNICAL ENGLISH COMMUNICATION Hours per week:3

L	T	P	C
1	0	2	**2**

PREREQUISITE KNOWLEDGE:

Basic sentence formation, Understanding contextual meanings, Basic writing skills and Moderate fluency in English.

COURSE DESCRIPTION AND OBJECTIVES:

In this course students will read, analyze, and interpret material from technical and general fields, and practice reading, writing, listening and speaking skills to gain exposure and functional English on a variety of contemporary topics. The overall course objective is to provide English for Specific Purposes(ESP) instruction to enhance students' reading, writing, listening and speaking skills through a practice in the language. It will aim to build students' confidence and motivation through exposure to academic skills like Note making/taking, Paraphrasing, Summarizing, Report Writing, Making Presentations etc., so as to generate interest in the language from an ESP perspective. Finally, students are expected through the course to gain key strategies and expression for communicating with professionals and non-specialists

MODULE – 1

UNIT – 1 **4L+0T+8P=12 Hours**

GENETICS

READING: Reading for **Note Making** Sub skills: Reading for global understanding (skimming), specific information (scanning), understanding main ideas and supporting ideas, guessing contextual meanings from the text. -**Vocabulary building:** commonly used roots, prefixes, and suffixes.

WRITING: Note making, organising main points and sub points, numbering and sequencing, suggesting titles, paraphrasing and summarising. **Functional grammar:** Common Errors in Articles and Prepositions (Handout)

LISTENING: Listening for Note Taking: top down and bottom up approach, listening for main ideas and supporting points.

SPEAKING: PRESENTATION in teams - ideas on the topic summarised, making a PPT, effective introductions and conclusions, logical organisation of content, using appropriate structure and cohesive devices

UNIT – 2 **4L+0T+8P=12 Hours**

GENETICS

READING: Reading: predicting, skimming, scanning, reading for inference, extrapolative reading

VOCABULARY BUILDING: Academic vocabulary from the text: synonyms, antonyms, Words often confused

WRITING: Paragraph writing; writing a topic sentence, supporting sentences, effective introductions and conclusions, use of cohesive devices. Types of Paragraphs: Descriptive, narrative, argumentative and expository.

FUNCTIONAL GRAMMAR: Common Errors in Verb forms and Conditional sentences (Handout)

LISTENING: Listening for identifying parts from a description, listening to and sorting information, listening for specific information.

SPEAKING: Narrating/Retelling an incident, using suitable cohesive devices/discourse markers Speaking of past and present habits/ activities/events - Speaking of future plans.

PRACTICES:

- Note making
- Summarizing
- Paragraph Writing
- Error correction and Restructuring
- Vocabulary building
- Listening comprehension
- Note taking

MODULE – 2

UNIT – 1 **4L+0T+8P=12Hours**

SOCIAL MEDIA – HEALTH AND NUTRITION

Reading: Reading for factual information researching for supporting evidence - skimming, scanning, **Vocabulary building:** One-word substitutes.

Writing: Letter Writing - E-mail writing – New age communication – Format, protocol, and style- WhatsApp, Facebook and Twitter **Functional grammar:** Common Errors in Sub-Verb Agreement and Modals

Listening: Listening to a **Business Presentation**: Listening for deducing information, for abstract details and specific details, listening for taking a message.

Speaking: Making a presentation with a PPT on a topic assigned- organising the presentation using appropriate discourse markers - presenting a point of view - Extempore

UNIT – 2 **4L+0T+8P=12Hours**

FASHION

Reading: Reading for data interpretation and information transfer from graphical aids to text reports (pictograms. tables, graphs, pie charts, flow charts), deducing specific information and general information -Vocabulary building**:** business vocabulary, collocations**,** idioms and phrasal verbs

Writing: Writing a Report: Drafting general and factual reports - **w**riting an overview - an effective introduction - organising information into paragraphs (Stages of writing: planning /organising /writing /editing /rewriting)

Functional grammar: transformations and miscellaneous common errors

Listening: Listening to a Ted talk and sorting information – taking notes from a discussion.

Speaking: Group Discussion – prerequisites -generating content - initiating a discussion - expressing one's opinion ~ leading a discussion - agreeing/ disagreeing to someone's view - cutting into a speech - body language and voice modulation.

PRACTICES:

- E-mail writing
- Letter writing
- Report writing
- Messaging in Social media
- Extempore
- Making PPTs

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

COs	Course Outcomes	Blooms Level	Module No	POs
1	Possess comprehensive skills in listening and reading business texts in formal context	Applying	2	7
2	Communicate effectively both in their academic as well as professional environment	Applying and creating	1&2	10
3	Clear grasp on the register of business language	Applying	1	8
4	Possess the ability to write business reports and proposals clearly and precisely to succeed in their future	Applying and creating	1	12
5	Make effective presentations and participate in formal context	Applying and creating	2	10

TEXT BOOK:

1. N P Sudharshana & C Savitha, English For Technical Communication, Cambridge University Press, 2016.

REFERENCE BOOKS:

1. Balasubramanian T, A Text book of Phonetics for Indian Students. Orient Longman, New Delhi, 1989.
2. Krishnaswamy, N and Sriraman, T, Current English for Colleges. Trinity publications, 2016.
3. Mohan Krishna and Meera Banerjee, Developing Communication Skills. Macmillan India Ltd. New Delhi, 1990.
4. Ashraf Rizvi M, Effective Technical Communication, 2nd Edition, McGraw Hill Education, 2017.
5. Narayanaswamy V R, Strengthen your Writing. Third Edition Orient Black Swan, New Delhi, 2005.
6. Naterop, Jean, B. and Rod Revell, Telephoning in English. Cambridge University Press, Cambridge, 1997.

22MC108-SOFTWARE ENGINEERING AND TESTING

Hours per week:5

L	T	P	C
3	0	2	4

PREREQUISITE KNOWLEDGE:

Data Base Management Systems, Operating Systems, Object Oriented Programming.

COURSE DESCRIPTION AND OBJECTIVES:

This course focuses on the concepts of software life cycle, role of process models and methods to prepare software requirement specification document. In addition to that, it also imparts knowledge of design, development and testing of software. The objective of this course is to enable the student to develop efficient, cost effective, feasible software as per user requirements.

MODULE – 1

UNIT – 1 **12L+0T+0P=12 Hours**

INTRODUCTION TO SOFTWARE ENGINEERING: GENERIC VIEW OF PROCESS-PROCESS MODELS

The evolving role of software, Software, Changing Nature of Software, Software myths. Software Engineering - A layered technology, A process framework, Software Development Life Cycle (SDLC), The Capability Maturity Model Integration (CMMI), Process Assessment. The Waterfall Model, Incremental Process Models, Evolutionary Process Models (Spiral and Prototype models).

UNIT – 2 **12L+0T+16P=28 Hours**

AN AGILE VIEW OF PROCESS- REQUIREMENTS ENGINEERING-BUILDING THE ANALYSIS MODEL

Agile process models - The Unified process, Extreme Programming, Scrum. Inception, Elicitation, Elaboration, Negotiation, Specification (SRS Document, IEEE Standards for SRS), Validation, Requirements management, Feasibility Study. Data modeling - Data objects, Attributes, Relationship, Cardinality and modality; Class based modeling - Identify analysis classes, specify attributes and Define operations.

PRACTICES:

- Identifying the Requirements from Problem Statements.
- Estimation of Project Metrics.
- Modeling UML Use Case Diagrams and Capturing Use Case Scenarios.
- State chart and Activity Modeling.
- Modeling UML Class Diagrams and Sequence diagrams.

MODULE – 2

UNIT – 1 **12L+0T+0P=12 Hours**

INTRODUCTION TO SOFTWARE TESTING

A strategic approach to software testing; Test strategies for conventional software; Validation testing; System testing. Black-Box and White-Box testing. Reactive vs Proactive risk strategies; Software risks; Risk identification; Risk projection; Risk

refinement; RMMM; RMMM plan. Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software Quality Assurance, Configuration management, assessing and controlling software quality.

UNIT – 2 **12L+0T+16P=28 Hours**

SOFTWARE TESTING & TOOLS

A strategic approach to software testing; Test strategies for conventional software; Validation testing; System testing. Black-Box and White-Box testing. Reactive vs Proactive risk strategies; Software risks; Risk identification; Risk projection; Risk refinement; RMMM; RMMM plan. Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software quality Assurance, Configuration management, assessing and controlling software quality.

PRACTICES:

- Modeling UML Class Diagrams and Sequence diagrams.
- Estimation of Test Coverage Metrics and Structural Complexity.
- Design of Test Cases**.**

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Apply the basic concepts of software engineering.	Apply	1	1,11
2	Compare different process models and identify appropriate process model based on project requirements.	Apply	1, 2	2,4
3	Build Software Requirement Specification (SRS) document for any software product	Analyze	1	3,5
4	Design of solution using UML diagrams like use case, sequence diagrams etc.	Create	1,2	3,4,5
5	Create different testing techniques to ensure bug free software and measure metrics such as software size and quality of the product.	Create	2	4,5,11

TEXT BOOKS:

1. Roger S. Pressman, “Software Engineering, A practitioner’s Approach”, 6th edition, McGrawHill International edition, 2008
2. Booch G., Rumbaugh J. and Jacobsons I, “The Unified Modeling Language User Guide”, 2nd edition, Addison Wesley, 2005.

REFERENCE BOOKS:

1. Simon Sennet, Steve McRobb and Ray Farmer, “Object Oriented Systems Analysis and Design, 2nd edition, 2004.
2. Deepak Jain “Software Engineering, Principles and Practices”, Oxford University Press,2010.
3. Pankaj Jalote, “Software Engineering, A Precise Approach”, Wiley India, 2010.

22MC109-WEB TECHNOLOGIES

Hours per week:5

L	T	P	**C**
3	0	2	**4**

PREREQUISITE KNOWLEDGE: Computer Programming.

COURSE DESCRIPTION AND OBJECTIVES:

This course offers the basic concepts used to develop static web pages and it also provides knowledge of Internet programming concepts. Further, this course is to build web applications using HTML, CSS, and client side script technologies that span multiple domains.

MODULE – 1

UNIT–1 **12L+0T+8P=20 Hours**

BASIC HTML TAGS

Coding Basics: HTML Syntax, html, head, title, and body tags, Headings, paragraphs and lists, The strong and em tags, The doctype, The lang attribute, The meta tag, and the Unicode character set.

Coding Links: Absolute & Relative URLs: Anchor tags and hrefs, Linking to other websites, Linking to pages within a website, Opening a link in a new browser window/tab.

Adding Images: The break tag, The image tag, and source attribute, Using the width, height, and alt attributes, Using horizontal rules, tables, forms, and frames.

UNIT–2 **12L+0T+8P=20 Hours**

CASCADING STYLE SHEETS & DIV TAGS

Cascading Style Sheets (CSS): The style tag, Tag selectors, font size, font family, color, & line-height properties, and Hexadecimal color codes.

CSS Properties: Text, background, border, list and font.

CSS Class Selectors: The class attribute, CSS class selectors, The span tag, CSS opacity.

Div Tags, ID Selectors, & Basic Page Formatting: Dividing up content with the div tag, Assigning IDs to divs, Setting width & max-width, CSS background color, Adding padding inside a div, Centering content, CSS borders, CSS shorthand & the DRY principle.

PRACTICES:

- Practice Basic HTML tags
- Create links on same page and other pages
- Insert images on a web page
- Create lists on a web page
- Create Tables on a web page
- Create forms such as login form and registration form etc.
- Working with Frames
- Add different types of CSS to web pages
- Usage of div tag in the web page
- Create a personal website using HTML and CSS.

MODULE – 2

UNIT–1 **12L+0T+8P=20Hours**

JAVA SCRIPT

Introduction, Document Object Model, Language Syntax, Variable declaration, Operators, Control Statements, Understanding Arrays, Function Declaration.

Built-in Functions: Standard Date and Time, String, Array and Math

UNIT–2 **12L+0T+8P=20Hours**

HTML FORM VALIDATIONS

HTML Document Object Model, Working with HTML form and its elements.

Working with Objects and Classes: Working with Objects, Call method in JavaScript, Inheritance in JavaScript using prototype.

Java script Events: Keyboard events, mouse events, form events.

PRACTICES:

- Practice basic JavaScript programs such as the variable declaration and operators
- Usage of Control Statements in JavaScript
- Creating and accessing arrays in JavaScript
- Working with functions in JavaScript
- Perform validations on HTML forms using JavaScript
- Working with Cookies
- Create JavaScript Objects and Classes
- Apply JavaScript on HTML and CSS webpages

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Familiar with HTML Tags	Apply	1	1
2	Create static web pages using forms	Apply	1	1, 2, 3
3	Apply Cascading Style Sheets and Div Tags to HTML static webpages.	Apply	1	1, 2, 3
4	Familiar with JavaScript functions and form validations.	Analyze	2	1, 2, 3,4, 5
5	Design and develop dynamic websites	Evaluate	2	1, 2, 3,4, 5

TEXT BOOKS:

1. Jon Duckett, “Beginning Web Programming with HTML, XHTML, and CSS”, 2 nd Edition, Wiley India Pvt. Ltd, 2008.
4. Julie C. Meloni , “HTML, CSS, and JavaScript All in One”, Sams Teach Yourself, 3 rd Edition,Pearson, 2015.

REFERENCE BOOKS:

1. Chris Bates, “Web Programming, Building Internet Applications”, 3 rd Edition, Wiley Dream Tech, 2006.
2. Jon Duckett, “HTML & CSS: Design and Build Websites”, 1 st Edition, John Wiley & Sons, 2011.

22MC110-COMPUTER NETWORKS

Hours per week:4

L	T	P	C
2	2	0	**3**

PREREQUISITE KNOWLEDGE: Basics of Networks.

COURSE DESCRIPTION AND OBJECTIVES:

This course focuses on imparting knowledge about various protocols involved in LANs and WANs. In addition, it gives good foundation on different protocols such as data link protocols, internet protocols and transport protocols present in the respective layers of computer networks.

MODULE – 1

UNIT–1 **8L+8T+0P=16 Hours**

INTRODUCTION TO NETWORKS AND PHYSICAL LAYER

Uses of computer networks, Network hardware, Network software, Reference Models, Guided transmission media, Data link layer design issues, Elementary data link protocols, The channel allocation problem**.**

UNIT–2 **8L+8T+0P=16 Hours**

DATA LINK LAYER AND MAC LAYER

Medium allocation methods, Error detection and correction, Sliding window protocols, Example data link protocols, Multiple access protocols.

PRACTICES:

- Bit stuffing and byte stuffing
- Error detection and correction
- Performance calculation in MCA protocols
- Performance evaluation in sliding window protocol
- Study on physical addressing

MODULE – 2

UNIT–1 **8L+8T+0P=16 Hours**

NETWORK LAYER

Network layer design issues, Routing algorithms - optimality principle, shortest path routing, flooding, distance vector routing, link state routing and hierarchical routing; The network layer in the internet - IP addresses, Services provided to the upper layers.

UNIT–2 **8L+8T+0P=16 Hours**

TRANSPORT AND APPLICATION LAYER

Elements of transport protocols, Congestion control algorithms, OOS improving techniques, leaky bucket and token bucket algorithms, Internetworking, IPv4, IPv6, IMCP, ARP, DHCP and mobile IP, Congestion Control, UDP and TCP, Performance Issues, The world wide web, Domain name system, E-mail, Streaming Audio and Video, Content Delivery

PRACTICES:

- Logical addressing division
- Performance evaluation routing and congestion control algorithms
- TCP/IP programming
- UCP/IP programming
- Configuration of email system

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Apply the suitable network devices and methodologies to establish required network for given scenario	Apply	1	1, 2, 12
2	Simulate and demonstrate the OSI reference model layer services	Analyze	1	1, 2, 3, 5, 12
3	Analyze the various computer network addressing techniques like physical addressing, logical addressing, port addresses and special addressing.	Apply	2	1, 2, 5, 12
4	Design and develop simple network applications using TCP/IP suite in one high level programming language	Analyze	2	1, 2, 12

TEXT BOOK:

1. Andrew S.Tanenbaum, “Computer Networks”, 5th Edition. Pearson Education, 2014.

REFERENCE BOOKS:

1. Behrouz A. Forouzan, “Data communications and Networking”, 4th Edition, TMH, 2017.
2. William Stallings, “Data and Computer Communications”, 10th Edition, Pearson Education, 2017.

22MC111-DATA MINING TECHNIQUES

Hours per week:5

L	T	P	**C**
3	0	2	**4**

PREREQUISITE KNOWLEDGE: Basics of statistics, Linear algebra.

COURSE DESCRIPTION AND OBJECTIVES:

This course introduces the concepts, principles, methods and implementations of data mining subject, with a focus on 3 major data mining functions: 1. Association rule mining 2. Classification 3. Cluster Analysis. In the first part of the course student will learn terminology of data mining and different kinds of data pre-processing techniques. In the second part of the course student will learn major data mining techniques, their implementation and applications.

MODULE – 1

UNIT–1 **10L+0T+6P=16 Hours**

INTRODUCTION TO DATA MINING

Introduction to the need of data mining, Different kinds of data and patterns that can be mined, basic statistical descriptions of data, matrix representation of data, similarity and dissimilarity, need of the data pre-processing, handling missing values and noise in data, identifying redundant and correlated attributes, finding frequent item sets and generating association rules using Apriori and FP growth algorithms

UNIT–2 **14L+0T+10P=24 Hours**

DATA HANDLING USING STATISTICAL METHODS

Computing basic statistical descriptions of different kinds of data.(mean, median, mode, variance, and standard deviation), distance and similarity metrics(Euclidean, Manhattan, cosine).Missing values handling methods, computing correlation of nominal and numerical data using Chi-square & Pearson method, computing Frequent patterns using Apriori and FP growth methods.

PRACTICES:

- Computing mean, median, mode, variance, Standard deviation measures.
- Matrix representations of data
- Computing Euclidean, Manhattan, Cosine distance measures.
- Missing values handling methods.
- Chi-square method
- Pearson method
- Finding frequent patterns and association rules using Apriori algorithm
- Finding frequent patterns and association rules using Frequent Pattern growth algorithm

MODULE – 2

UNIT–1 **10L+0T+6P=16 Hours**

CLASSIFICATION

Introduction to classification and different approaches to perform classification. Concept of Information Gain, gain ratio and Gini Index, Bayes theorem and its terms, linearly separable

and inseparable cases, concept of Neural Networks and working model of Multilayer Perceptron classifier, model evaluation techniques, classification through Encembling Techniques.

UNIT–2 **14L+0T+10P=24 Hours**

CLUSTERING

Introduction to clustering and different approaches for clustering. K means and K medoid methods of partitioning based clustering, Agglomerative and Divisive methods of hierarchical clustering. Cluster evaluation methods.

PRACTICES:

- Decision tree classifier
- Naïve Bayes classifier
- KNN classifier
- SVM classifier
- Forward propagation of NN
- Updating the weights for backward propagation of NN
- K means and K medoid Clustering technique
- Agglomerative and Divisive clustering technique

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Build the data analysis and visualization reports for large data sets.	Apply	1	1, 2, 12
2	Generate Association rules by finding frequent patters from large transactional datasets.	Create	1	1, 2, 5, 12
3	Apply classification techniques to build the classifiers for binary and multiclass problems.	Apply	2	1, 2, 3, 5, 12
4	Apply various clustering techniques to label the unlabeled data	Apply	2	1, 2, 12

TEXT BOOK:

1. Jiawei Han, Micheline Kamber and Jian Pei, “Data mining Concepts and Techniques”, 3rd edition, Morgan Kaufmann. 2012.

REFERENCE BOOKS:

1. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, “Introduction to Data Mining”, 2nd edition, Pearson, 2018.
2. Jure Leskovec, Anand R aja raman and Jeffrey D Ullman, “Mining of Massive Datasets”, 5th edition, Stanford University, 2014.

22MC112-DATA VISUALIZATION

Hours per week:4

L	T	P	**C**
0	0	4	**2**

PREREQUISITE KNOWLEDGE: Python programming.

COURSE DESCRIPTION AND OBJECTIVES:

Statistics is critical for machine learning and this course imparts sufficient statistical knowledge required for machine learning. Statistics allow us to gather insights from data and determine whether our assumptions are valid or not. Using statistics, we can make educated assumptions and forecasts based on real-world data. This course makes students familiar with python libraries to apply statistical analysis and covers advanced python data structures and visualization methods. In addition, it teaches how to use python to implement advanced statistical analysis.

MODULE – 1

UNIT–1 **0L+0T+16P=16 Hours**

PYTHON FOR STATISTICS

Why Statistics? Python Packages for Statistics, Numpy and Pandas data structures for statistics; Data Input: Load data from CSV, Excel, ASCII and Text Files; Saving data into files; Data types.

UNIT–2 **0L+0T+16P=16 Hours**

SAMPLING TECHNIQUES

Populations and Samples: Population Vs Sample, Need for Sampling,

Sampling Techniques: Random Sampling, Clusters Sampling, Systematic Sampling, Stratified Sampling Techniques.

Descriptive Statistical Analysis: Measures of Central Tendency: Mean, Median, Mode, Geometric Mean; Measures of Dispersion: Range, Percentiles, Standard Deviation and Variance.

PRACTICES:

- **Write code to perform the following operations on Numpy arrays:**
  a) Create a 2D Numpy array with 24 elements of size 4x6 and retrieve the last three rows, retrieve the first two column values, retrieve the sum of the second row, retrieve the sum of first column, and display the max value index in the array
  b) Create a 2D Numpy array with 42 elements of size 7x6, add a new row, Delete an existing column, replace a specific value, and identify how many values are less than given x
  c) Create a 1D-array with 64 elements, Reshape the array into 4, 2x8 arrays, also reshape the array into other possible shapes, Convert the data type into float, Split the array into three sub-arrays of same size
  d) Create a 2D Numpy array with 35 elements of size 7x5, identify unique values in the array, identify the existence of duplicates, perform conditional replace operations, insert NaNs, replace NaNs,

- **Load a CSV file into Numpy array and apply the following:**
    a) Write the code routine to print the masked (gray) colored sub-array
    b) Print the maximum of the fifth row.
    c) Reshape the array (change columns to rows, rows to columns)
    d) Extract all the odd number using conditional logic
    e) Find the column wise mean, std and variance
- **Pandas library for analyzing tabular data:**
    a) Load data from CSV files and understand your data
    b) Query and index operations on the above data frame
    c) Insert, delete and update your data
    d) Apply various filters on the data
    e) Group, merge and aggregate data in the data frames
    f) Identify and Fix missing values in the data
- **Apply the following operations on the given csv file**
    a) Load data from CSV files
    b) Retrieve first 10, last 10 rows, 3rd Column and a subgroup
    c) Query and index operations on the above data frame
    d) Insert, delete and update your data
    e) Apply aggregate operations
    f) Apply various filters on the data
    g) Group, merge, and aggregate data in the data frames
- **Visualize data with the help of the following graphical representations**:
    (a) Line plots (b) Bar plots (c) Error Plots (d) Scatter plots
    (e) KDE Plots (f) Heat Maps (g) Box Plots (h) Pie graph
    (i) Histogram (j) multiple graphs in single figure (k) saving figures

MODULE – 2

UNIT–1 **0L+0T+16P=16 Hours**

HYPOTHESIS TESTS FOR STATISTICAL ANALYSIS

Typical analysis Procedure: Data Screening and Outliers, Normality Check; Hypothesis Concept, Errors, p-Value, Interpretation of the p-Value, Types of Error, Sensitivity and Specificity

Hypothesis Tests For Statistical Analysis: z-test, Student's t-Test, One-Way Chi-Square Test, Chi-Square Contingency Test, Analysis of Variance (ANOVA): One-Way ANOVA, Two-Way ANOVA.

UNIT–2 **0L+0T+16P=16 Hours**

LINEAR REGRESSION MODELS

Linear Regression Models: Linear Correlation-Correlation Coefficient, Rank Correlation; General Linear Regression Model, Coefficient of Determination, Linear Regression Analysis with Python

PRACTICES:

- **Sampling and Resampling**:
    a) Generate a population of random numbers
    b) Generate multiple samples using Random sampling with and without random sampling
    c) Load a balanced dataset and visualize the class distribution

d) Load an imbalanced dataset and visualize the class distribution

- **Interpreting Data Using Descriptive Statistics:** Compute Mean, Median, Mode, Standard Deviation, Variance, Co-variance, Interquartile Range and Skewness for two different datasets and write your interpretations about these statistical measures. Which measure is best suitable? Justify.
- **Generating Samples from Probability Distributions:**
    a) Generate a set of random numbers (which corresponds to a uniform distribution) using the function rand and plot its histogram. What is the shape of this histogram and why?
    b) Investigate how the shape of the histogram is affected by the number of random numbers you have generated.
    c) Similarly generate numbers using Bernoulli, Binomial distributions and plot a histo gram and check the shape.
    d) Generate numbers using exponential and poisson distributions and plot a histo gram and check the shape.

- **Hypothesis tests:** Implement the following three popular statistical techniques for hypothesis testing:
    a) Chi-square test, T-test and ANOVA test (Calculate the Test Statistic and P-value by running a Hypothesis test that well suits your data and Make Conclusions).
- **Linear Regression Analysis**: Download house prediction dataset and explore the data, Prepare the dataset for training, Train a linear regression model, and Make predictions and evaluate the model.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Apply data acquisition tools to collect and visualize	Apply	1	1, 2, 12
2	Analyze data by evaluating various statistical measure	Analyze	1	1, 2, 5, 12
3	Develop statistical models for data analysis	Create	2	1, 2, 3, 5, 12
4	Installation and Usage of Python Tools for data visualization and statistical analysis	Apply	2	1, 2, 12

TEXT BOOKS:

1. Thomas Haslwanter, “An Introduction to Statistics with Python With Applications in the Life Sciences, - Springer- Springer International Publishing, 2016.
2. Zed A. Shaw, “Learn Python 3 the Hard Way”, 1st edition, Pearson Education Inc., 2018.

REFERENCE BOOKS:

1. Peter Bruce, Andrew Bruce, Peter Gedeck, “Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python” 2nd edition, Oreilly Publishers, 2020.
2. Bharti Motwani, “Data Analytics using Python” , 1st edition, Wiley Publisheres, 2021

22MC114-SOFT SKILLS LABORATORY

Hours per week:2

L	T	P	C
0	0	2	**1**

PREREQUISITE KNOWLEDGE: Basic sentence formation, Understanding contextual meanings, Basic writing skills and Moderate fluency in English.

COURSE DESCRIPTION AND OBJECTIVES:

- To impart employability skills like resume preparation and facing interviews
- To enable trainees to develop interpersonal and leadership skills
- To train them on work place skills like making presentations, participating in group discussions etc.

MODULE – 1

UNIT – 1 **0L+0T+8P=8 Hours**

PERSONALITY DEVELOPMENT

a) Soft Skills: Need for soft skills, professionalism, employability skills–

b) Communication: Need for effective communication - the process of communication, levels of communication, flow of communication, choice of diction and style with reference to setting (formal, semi-formal or informal) -communication networks, barriers to communication, miscommunication, noise and ways to overcome the barriers

Practice: Self Introduction

c) Career Planning:

- Job vs. career,
- SWOT analysis

Practice: Personal and Academic SWOC

d) Johari Window

Practice: Giving and taking opinions of Self Vs others and assessing oneself.

e) Goal setting

Practice: Short, Mid and Long Term goals planning the semester

f) Time management: four quadrant system

Practice: Stephen Covey Time Management Matrix planning a semester

g) Stress-management

Practice: Questionnaire to assess level of stress

UNIT – 2 **0L+0T+8P=8Hours**

VOCABULARY BUILDING

a) Vocabulary Building: Word etymology, roots, prefixes & suffixes, synonyms & antonyms, collocations, one-word substitutes, analogies, idioms and phrases, contextual guessing of unfamiliar words, task-oriented learning

Practice: (50 words) relating to resume preparation and Interviews, newly coined words

b) Reflects of language on Personality

- Gender sensitive language in MNCs
- Mind your language

Practice: Gender sensitive words and Words acceptable in Indian context and objectionable international context

C) Seven essential skills for a team player; attentive listening, intelligent questioning, gently persuading, respecting other's views, assisting others, sharing, participating actively

MODULE – 2

UNIT–1 **0L+0T+8P=8 Hours**

FUNCTIONAL ENGLISH

a) Functional English: Situational dialogues, Role plays (including small talk)

Practice: Opening and closing a telephonic conversation, making an appointment, making a query, Offering/Passing on information, communicating with superiors, expressing agreement/objection, opening bank account (combination of prepared and impromptu situations given to each student)

b) Group Discussion: Articulation and flow of oral presentation, dynamics of group discussion, intervention, summarizing and conclusion, voice modulation, content generation, Key Word Approach (KWA), Social, Political, Economic, Legal and Technical Approach (SPELT), View Point of Affected Part (VAP), language relevance, fluency and coherence – 11th and 12th weeks

Practice: Group Discussions on various topics

c) Resume preparation: Structure and presentation, defining career objective, projecting one's strengths and skill-sets, summarizing, formats and styles and covering letter-

d) Statement of Purpose

Practice: Preparing one's SoP and Resume

UNIT – 2 **0L+0T+8P=8 Hours**

PRESENTATION SKILLS & INTERVIEWS

a) Facing Interviews:

Interview process, understanding employer expectations, pre-interview planning, opening strategies, impressive self-introduction, answering strategies, other critical aspects such as body language, grooming, other types of interviews such as stress-based interviews, tele-interviews, video interviews, frequently asked questions (FAQs) including behavioral and HR questions and the aspect looked at by corporate during interviews

Practice: Mock interviews on the FAQs including feedback

b) **Presentation Skills:** Selection of a topic, preparing an abstract, gathering information, organizing the information, drafting the paper, citing reference sources – writing striking introductions, discussing the methodology used, developing the argument, presentation style, language, presenting the paper and spontaneously answering audience questions

Practice: oral presentation with the help of technology (Preparing PPT and presenting)

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No	Course Outcomes	Blooms Level	Module No	POs
1	Have the ability to introspect on individual strengths and weaknesses, and emerge as a balanced personality with improved self-awareness and self-worth	Applying	1	12
2	Observe gender sensitive language and workplace etiquette in his professional life	Applying	1	9
3	Be able to prepare a resume and gain the confidence to face an interview	Creating	1&2	10
4	Possess the interpersonal skills to conduct himself/herself effectively in everyday professional and social contexts	Appling	2	8
5	Be able to bring professionalism into his/her daily activities	Applying	2	8

TEXT BOOKS:

1. Krishna Mohan & NP Singh, "Speaking English effectively" 1st edition, Macmillan, 2008.
2. Dr. S.P. Dhanvel, English and Soft skills, Orient Blackswan, 2011
3. Rajiv K. Mishra, Personality Development, Rupa& Co. 2004.

REFERENCE BOOKS:

1. Edward Holffman, Ace the corporate personality, McGraw Hill, 2001

2.Adrian Furnham, Personality and intelligence at work, Psychology Press, 2008.

3.John Adair Kegan Page, "Leadership for innovation" 1st edition, Kogan, 2007.

22MC201-CRYPTOGRAPHY AND NETWORK SECURITY

Hours per week:5

L	T	P	C
3	0	2	**4**

PREREQUISITE KNOWLEDGE:

Computers Networks, Number theory and computational Complexity.

COURSE DESCRIPTION AND OBJECTIVES:

Cryptography refers to secure information and communication techniques derived from mathematical concepts and a set of rule-based calculations called algorithms, to transform messages in ways that are hard to decipher. To understand Cryptography Theories, Algorithms and Systems and understand necessary Approaches and Techniques to build protection mechanisms in order to secure computer networks.

MODULE – 1

UNIT - 1 **12L+0T+8P=20 Hours**

SECURITY CLASSICAL ENCRYPTION

Security trends - Legal, Ethical and Professional Aspects of Security, Need for Security at Multiple levels, Security Policies - Model of network security – Security attacks, services and mechanisms – OSI security architecture – Classical encryption techniques: substitution techniques, transposition techniques, steganography).- Foundations of modern cryptography: perfect security – information theory – product cryptosystem – cryptanalysis.

UNIT - 2 **12L+0T+8P=20 Hours**

MATHEMATICS OF SYMMETRIC KEY CRYPTOGRAPHY

Algebraic structures- Modular Arithmetic-Euclid's algorithm- Congruence and matrices - Groups, Rings, Fields- Finite fields- SYMMETRIC KEY CIPHERS: SDES – Block cipher Principles of DES – Strength of DES – Differential and linear cryptanalysis - Block cipher design principles – Block cipher mode of operation -Evaluation criteria for AES – Advanced Encryption Standard - RC4 – Key distribution.

PRACTICES:

- Understand the OSI Model
- Understand Types of Network Devices.
- Know Network Defenses.
- Segregate Your Network.
- Place Your Security Devices Correctly.
- Use Network Address Translation.
- Don't Disable Personal Firewalls.
- Use Centralized Logging and Immediate Log Analysis

MODULE – 2

UNIT - 1 **12L+0T+8P=20 Hours**

MATHEMATICS OF ASYMMETRIC KEY CRYPTOGRAPHY : Primes – Primality Testing – Factorization – Euler's totient function, Fermat's and Euler's Theorem - Chinese Remainder Theorem – Exponentiation and logarithm - **ASYMMETRIC KEY CIPHERS:**

RSA Cryptosystem – Key distribution – Key management – Diffie Hellman key exchange -ElGamal cryptosystem – Elliptic curve arithmetic-Elliptic curve cryptography. Authentication requirement – Authentication function – MAC – Hash function – Security of hash function and MAC – SHA –Digital signature and authentication protocols – DSS-Entity Authentication: Biometrics, Passwords, Challenge Resp

UNIT - 2 **12L+0T+8P=20 Hours**

SECURITY PRACTICE AND SYSTEM SECURITY

Electronic Mail security – PGP, S/MIME – IP security – Web Security – SYSTEM SECURITY: Intruders – Malicious software – viruses – Firewalls.

PRACTICES:

- Protect your data.
- Avoid pop-ups, unknown emails, and links.
- Use strong password protection and authentication.
- Connect to secure Wi-Fi.
- Enable firewall protection at work and at home.
- Invest in security systems.
- Install security software updates and back up your files.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Understand the fundamentals of networks security, security architecture, threats and vulnerabilities	Analyze	1	1, 2, 12
2	Apply the different cryptographic operations of symmetric cryptographic algorithms	Apply	1	1, 2, 6, 12
3	Apply the different cryptographic operations of public key cryptography	Apply	1	1, 2, 4, 6, 12
4	Apply the various Authentication schemes to simulate different applications.	Apply	2	1, 2, 12

TEXT BOOK:

1. William Stallings, Cryptography and Network Security: Principles and Practice, PHI 3rd Edition, 2006.

REFERENCES:

1. C K Shyamala, N Harini and Dr. T R Padmanabhan: Cryptography and Network Security, Wiley India Pvt.Ltd, 2017.
2. BehrouzA.Foruzan, Cryptography and Network Security, Tata McGraw Hill, 2007.
3. Charlie Kaufman, Radia Perlman, and Mike Speciner, Network Security: Communication in a PUBLIC World, Prentice Hall, 2008.

22MC202-ORGANIZATION BEHAVIOR Hours per week:4

L	T	P	**C**
2	2	0	**3**

PREREQUISITE KNOWLEDGE:

Fundamentals of Behavior in an organization.

COURSE DESCRIPTION AND OBJECTIVES: To comprehend the dynamics of Management practices in international context, to evaluate behavioral frameworks that suits for Corporate world from Global perspective and various business model that suits for all international context

MODULE – 1

UNIT - 1 **8L+8T+0P=16 Hours**

INTRODUCTION TO OB

Nature of OB: Nature and scope of OB - contributing disciplines to OB - Environmental and Organizational context of Organizational Behavior.

UNIT - 2 **8L+8T+0P=16 Hours**

PERCEPTION, PERSONALITY AND ATTITUDE

Perception - Process: Individual and Organizational factors that influence perceptual process. Role of perception in managerial activities and organizational processes.
Personality and Attitudes: Personality as continuum – Meaning of Personality – Johari window and Transactional Analysis Nature and Dimension of Attitudes.

PRACTICES:

- Survey on resistance to changing policies in The Banking Sector, The IT Sector
- Undertake a study to find out the various non-financial incentives used to motivate employees.
- A study in job enrichment and factors contributing to absenteeism and employee turnover in any industry of your choice.
- Analyze the characteristics and components of attitudes.
- Perform a study on the determinants of personality of a group of individuals.
- **Case study:** Organizational Behavior by Steven L McShane, Mary Ann VonGlinow and Radha R Sharma, TaTa McGraw Hill companies, Fouth Edition, Pg-6

MODULE – 2

UNIT - 1 **8L+8T+0P=16 Hours**

GROUP DYNAMICS

Group Dynamics: The Nature of groups. Kinds of groups – Stages of Group Development – Factors Contributing to Groups Cohesiveness - Meaning & types of stress – Effect of Stress – Strategies of cope with stress Principles of Learning & Reinforcement - Observational Learning - Cognitive Learning - Organizational Behavior Modification - Steps in Organizational Behavior Modification process.

UNIT - 2 **8L+8T+0P=16 Hours**

CONFLICT MANAGEMENT

Nature of conflict – Dynamics of Conflict – Conflict resolution modes – approaches to conflict management – sources of conflict in organization.

PRACTICES:

- Analyze the organizational culture and climate in the BPO industry.
- Conduct a study on the reasons for attrition in the BPO industry.
- Studying organizational structures of any 10 companies and classifying them into different types of organizations which are those organizations.
- Preparing the leadership profiles of any 5 business leaders and studying their leadership qualities and behaviors with respects to the trait, behavioral and contingency theories studied.
- **Case Study:**"Nuts and Bolts", Principles of Management, Cengagelearning , William , Manjunath , Sandhya Page no 531-532.
- Identifying any five job profiles and listing the various types, abilities required for those jobs and also the personality traits/attributes required for the jobs identified.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Understand nature and scope of OB	Understanding	1	1, 2, 12
2	Become aware of perceptual process and possible errors and act accordingly	Analyze	1	1, 2, 6, 12
3	Identify differences in personalities and attitudes	Evaluate	1	1, 2, 4, 6, 12
4	Act according to the group dynamics and handle stress	Apply	2	1, 2, 12
5	Resolve certain issues by applying conflict management	Apply	2	1, 2, 4, 6, 8

TEXT BOOKS:

1. Luthans, Fred, "Organizational Behaviour", 12th edition, Mcgraw Higher Ed, 2013.

REFERENCE BOOKS:

1. Debra L. Nelson, James Campbell Quick, "Organization Behavior", 8th edition, Cengage, 2013.
2. Afsaneh Nahavandi , "Organization Behavior", SAGE Publications, Inc; 1st edition, 2014.
3. Jr. Schermerhorn Richard N. Osborn, James G. Hunt, John R. "Organization Behavior", Wiley; 10th edition, 2009.

22MC203-CLOUD COUNTING

Hours per week:4

L	T	P	C
2	2	0	3

PREREQUISITE KNOWLEDGE: Computer Networks and Operating System.

COURSE DESCRIPTION AND OBJECTIVES:

- To understand the concept of cloud computing.
- To appreciate the evolution of cloud from the existing technologies.
- To have knowledge on the various issues in cloud computing.
- To be familiar with the lead players in cloud.
- To expose with the Server, Network and storage virtualization
- To appreciate the emergence of cloud as the next generation computing paradigm.
- To expose with the Server, Network and storage virtualization

MODULE – 1

UNIT - 1 **10L+6T+0P=16 Hours**

SYSTEMS MODELING, CLUSTERING AND VIRTUALIZATION

Scalable Computing over the Internet, Technologies for Network based systems, System models for Distributed and Cloud Computing, Software environments for distributed systems and clouds, Performance, Security and Energy Efficiency.Technologies for Network Based System -Implementation levels of Virtualization – Virtualization Structures/Tools and Mechanism – Virtualization of CPU, Memory, and I/O Devices - Virtual Clusters and Resource Management - Virtualization for Data-Center Automation.

UNIT - 2 **10+6T+0P=16 Hours**

CLOUD PLATFORM ARCHITECTURE OVER VIRTUALIZED DATA CENTERS

Cloud Computing and Service Models – Data-Center Design and Interconnection Networks – Architectural Design of Compute and Storage Clouds – Public Cloud Platforms: GAE, AWS, and Azure – Inter-cloud Resource Management. – Cloud Security and Trust Management.

MODULE – 2

UNIT - 1 **10L+6T+0P=16 Hours**

SERVICE-ORIENTED ARCHITECTURES FOR DISTRIBUTED COMPUTING

Services and Service-Oriented Architecture - Message-Oriented Middleware- Portals and Science Gateways - Discovery, Registries, Metadata, and Databases - Workflow in Service-Oriented Architectures.

UNIT - 2 **10L+6T+0P=16 Hours**

CLOUD APPLICATIONS

Cloud Applications: Scientific applications - Healthcare: ECG analysis in the cloud - Biology: protein structure prediction - Geoscience: satellite image processing - Business and consumer applications - CRM and ERP - Social networking - Multiplayer online gaming.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Able to interpret various Cloud computing models and services.	Analyze	1	1, 2, 3,4,5,7
2	Able to identify the significance of implementing virtualization techniques.	Evaluate	1	2,3,4,5,6,8
3	Able to understand the need of security in Cloud computing.	Apply	2	1, 2, 3,4, 6, 7
4	Understand the concept SOA and cloud based storage in Cloud computing.	Apply	2	2,3,4,5,6

TEXT BOOK:

1. Kai Hwang, Geoffrey C. Fox, Jack G. Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.

REFERENCE BOOKS:

1. Rittinghouse, John W., and James F. Ransome, “Cloud Computing: Implementation, Management and Security”, CRC Press, 2017.
2. Toby Velte, Anthony Velte, Robert Elsenpeter, "Cloud Computing - A Practical Approach”, Tata Mcgraw Hill, 2009.
3. George Reese, "Cloud Application Architectures: Building Applications and Infrastructure in the Cloud: Transactional Systems for EC2 and Beyond (Theory in Practice)”, O'Reilly, 2009

22MC204-INTERNET OF THINGS

Hours per week:4

L	T	P	C
2	2	0	3

PREREQUISITE KNOWLEDGE: Basic Knowledge of Hardware and Programming.

COURSE DESCRIPTION AND OBJECTIVES:

Students will be explored to the concepts and applications of Internet of Things, interconnection and integration of the physical world and the cyberspace. They are also able to design & develop IOT Devices and applications.

MODULE – 1

UNIT - 1 **10L+6T+0P=16 Hours**

INTRODUCTION TO IOT

Introduction to Internet of Things, Physical design of IoT, Logical design of IoT, IoT Enabling Technologies. Levels of IoT, IoT Applications, IoT Challenges, Sensors, Actuators.

UNIT - 2 **10L+6T+0P=16 Hours**

DOMAIN SPECIFIC IOTS

Home Automation, Cities, Environment, Retail, Logistics, Agriculture, Industry, Health & Life Style.

MODULE – 2

UNIT - 1 **10L+6T+0P=16 Hours**

ARDUINO INTERFACING, MACHINE-TO-MACHINE COMMUNICATION

Arduino Programming: Features, Types, Board details, IDE. Setup, Function Libraries, Examples programs. M2M: Introduction- Difference between IoT and M2M- Software Defined Networking (SDN), Network function virtualization, Need for IOT systems management, IOT system Management with NETCONF-YANG,

UNIT - 2 **10L+6T+0P=16 Hours**

DOMAIN SPECIFIC IOTS

IOT Design Methodology, Programming of IOT devices. Exemplary Device Raspberry Pi..**Case Studies:** Smart Lighting- Intrusion Detection - Weather monitoring- Indoor Air Quality Monitoring- Smart Parking- Smart Irrigation.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Analyze the application areas of IOT	Analyze	1	2
2	Analyze the building blocks of Internet of Things and characteristics	Analyze	1	3
3	Realize the revolution of Internet in Mobile Devices, Cloud & Sensor Networks.	Evaluate	2	2
4	Design and develop IoT applications for given specific problem	Apply	2	4

TEXT BOOKS:

1. Arsh deep Bahga, Vijay Madisetti, “Internet of Things: A Hands-on-Approach”, 1 st Edition, VPT,2014.
2. Shriram K Vasudevan, Abhishek S Nagarajan, RMD Sundaram, “Internet of Things”, 1 st Edition, John Wiley & Sons., 2019.

REFERENCE BOOKS:

1. Adrian McEwen, “Designing the Internet of Things”, 1 st Edition, Wiley Publishers, 2013.
2. Daniel Kellmereit, “The Silent Intelligence: The Internet of Things”,1 st Edition, DND Ventures LLC, 2013.
3. Maciej Kranz , ”Building the Internet of Things”, Wiley; 1st edition, 2016.

22MC801-BIG DATA ANALYTICS

Hours per week:4

L	T	P	C
2	2	0	3

PREREQUISITE KNOWLEDGE: Data warehousing, Python / Java Programming, SQL.

COURSE DESCRIPTION AND OBJECTIVES: This course gives an overview of Big Data, i.e., storage, retrieval, and processing of big data. In addition, it also focuses on the "technologies", i.e., the tools/algorithms that are available for storage, and processing of Big Data. It also helps a student to perform a variety of "analytics" on different data sets and to arrive at positive conclusions.

MODULE – 1

UNIT–1 **10L+6T+0P=16 Hours**

INTRODUCTION TO BIG DATA ANALYTICS

Introduction: Data, Characteristics of data and types of digital data, Sources of data, working with unstructured data, Evolution, and definition of big data, Characteristics and need of big data, and Challenges of big data.

BIG DATA ANALYTICS: Overview of business intelligence, Data Science, and analytics, Meaning and characteristics of big data analytics, Need for big data analytics, Classification of analytics, Challenges to big data analytics, Importance of big data analytics, Basic terminologies in the big data environment.

UNIT–2 **6L+10T+0P =16 Hours**

HADOOP, HDFS

Introducing Hadoop, Need of Hadoop, Limitations of RDBMS, RDBMS versus Hadoop, distributed computing challenges, History of Hadoop, Hadoop overview, the use cases of Hadoop, Hadoop distributors, HDFS, Processing data with Hadoop, managing resources and applications with Hadoop YARN, Hadoop ecosystem.

PRACTICES:

- HDFS basic command-line file operations.
- HDFS monitoring User Interface.
- Word Count Map Reduce program using Hadoop.
- Implementation of word count with combiner Map Reduce program.

MODULE – 2

UNIT–1 **10L+6T+0P =16 Hours**

MAPREDUCE, PIG, HIVE, SPARK

Introduction to mapper, reducer, combiner, partitioner, searching, sorting, compression, real-time applications using MapReduce, combiner, partitioner, Introduction to Hive, Hive architecture, Hive data types, Hive file format, Pig versus Hive. The anatomy of pig, Pig on Hadoop, Pig philosophy, Use case for pig, ETL processing, Pig Latin overview, Data types in pig, Relational operators, Piggy bank, Introduction, features of spark, components of spark.

UNIT – 2 **6L+10T+0P =16 Hours**

PROGRAMMING USING MAPREDUCE, PIG, HIVE, SPARK

Hive query language (HQL), Partitions and bucketing, RCFile Implementation, working with XML files, User-defined Function(UDF) in Hive, programming with Resilient Distributed Datasets (RDD), Running pig, Execution modes of pig, HDFS commands, Word count example using pig.

PRACTICES:

- Practice on Map Reduce Monitoring User Interface.
- Implementation of Sort operation using Map Reduce.
- Map Reduce program to count the occurrence of similar words in a file by using partitioner.
- Design Map Reduce solution to find the years whose average sales is greater than 30. input file format has year, sales of all months and average sales. Year Jan Feb Mar April May Jun July Aug Sep Oct Nov Dec Average.
- Map Reduce program to find Dept. wise salary,Empno, Emp Name Dept. Salary.
- Install and Run Pig then write Pig Latin scripts to sort, group, join, project and filter the data.
- Implementation of Word count using Pig.
- Creation of Database and tables using Hive query language.
- Creation of partitions and buckets using Hive.
- Practice of advanced features in Hive Query Language: RC File & XML data processing.
- Implement of word count using spark RDDs.
- Filter the log data using Spark RDDs.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Classify fundamentals of various big data analytics techniques	Apply	1	1, 2,3,4,9,10,12
2	Analyze the Big Data frameworks like Hadoop to efficiently store and process Big Data to generate Analytics.	Apply	2	1, 2,3,4,9,10,12
3	Analyze the HADOOP and Map Reduce technologies associated with big data analytics	Analyze	1	1, 2,3,4,9,10,12
4	Apply data analytics on real time datasets using Hive and Pig	Analyze	2	1, 2,3,4,9,10,12

TEXT BOOKS:

1. Seema Acharya and Subhashini Chellappan, “Big Data and Analytics”, 1st Edition, Wiley, Publishers, 2015.
2. Holden Karau, Andy Konwinski, Patrick Wendell and Matei Zaharia, “Learning Spark”, 1st Edition, Oreilly, 2015.

REFERENCE BOOKS:

1. Anand Rajaraman and Jeffrey David Ullman, “Mining of Massive Datasets”, Cambridge, University Press, 2012.
2. Boris lublinsky, Kevin t. Smith and AlexeyYakubovich, “Professional Hadoop Solutions”, 1st Edition, Wiley, 2015.
3. Chris Eaton and Dirkderoosetal, “Understanding Big data”, 1st Edition, McGraw Hill, 2012.
4. Tom White, “HADOOP: The definitive Guide”, 1st Edition, O Reilly 2012.
5. Vignesh Prajapati, “Big Data Analytics with R and Hadoop”,1st Edition, Packet Publishing, 2013.

22MC802-DIGITAL IMAGE PROCESSING Hours per week:4

L	T	P	C
2	2	0	3

PREREQUISITE KNOWLEDGE: Calculus and probability, Signals and systems.

COURSE DESCRIPTION AND OBJECTIVES:
To introduce the fundamentals of image processing at Low-Level and Mid-Level; and it covers Spatial and Frequency domain image enhancement, Edge detection, Segmentation, Image compression and Morphological image processing. To provide the student with programming experience from enhance the image and object recognition applications in MATLAB.

MODULE–1

UNIT–1 **8L+8T+0P=16 Hours**

DIGITAL IMAGE FUNDAMENTALS

Digital Image Fundamentals: Steps in Digital Image Processing, Components, Elements of Visual Perception, Image Sensing and Acquisition, Image Sampling and Quantization, Relationships between pixels, Color image fundamentals, RGB, HSI models, Two-dimensional mathematical preliminaries, 2D transforms - DFT, DCT.

UNIT–2 **8L+8T+0P=16 Hours**

IMAGE ENHANCEMENT, FREQUENCY DOMAIN AND IMAGE RESTORATION

Image Enhancement: Spatial Domain: Gray level transformations, Histogram processing, Basics of Spatial Filtering– Smoothing and Sharpening Spatial Filtering.

Frequency Domain: Introduction to Fourier Transform, Smoothing and Sharpening frequency domain filters, Ideal, Butterworth and Gaussian filters, Homomorphic filtering, Color image enhancement.

Image Restoration: Image Restoration, degradation model, Properties, Noise models, Mean Filters, Order Statistics, Adaptive filters, Band reject Filters, Band pass Filters, Notch Filters, Optimum Notch Filtering, Inverse Filtering, Wiener filtering.

PRACTICES:
Explain Steps in Digital Image Processing.
Explain Image Sampling and Quantization.
Discus on DFT, DCT.
Explain Gray level transformations.
Explain Fourier Transform for Smoothing and Sharpening.
Discus on Image Restoration - degradation model.

MODULE–2

UNIT–1 **8L+8T+0P=16 Hours**

IMAGE SEGMENTATION:

Image Segmentation: Edge detection, Edge linking via Hough transform, Thresholding, Region based segmentation, Region growing, Region splitting and merging, Morphological processing, erosion and dilation, Segmentation by morphological watersheds, basic concepts, Dam construction, Watershed segmentation algorithm.

UNIT–2 **8L+8T+0P=16 Hours**

IMAGE COMPRESSION AND RECOGNITION

Image Compression and Recognition: Need for data compression, Huffman, Run Length Encoding, Shift codes, Arithmetic coding, JPEG standard, MPEG. Boundary representation, Boundary description, Fourier Descriptor, Regional Descriptors, Topological feature, Texture Patterns and Pattern classes, Recognition based on matching.

PRACTICES:

- Explain Edge detection, Edge linking via Hough transform.
- Explain Morphological processing.
- Discus on Watershed segmentation algorithm.
- Discus on Need for data compression.
- Explain Arithmetic coding.
- Discus on JPEG standard, MPEG. Boundary representation.

COURSEOUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Apply medical image processing and image reconstructing for image analysis.	Apply	1	1, 2, 3, 9, 10, 12
2	Evaluate various image formation such as image rotation, transformation and scaling.	Evaluate	1, 2	1, 2, 3, 9, 10, 12
3	Analyse to create region of interest using Thresholding -Region based segmentation.	Analyse	1, 2	1, 2, 3, 4, 9, 10
4	Apply Fourier Transform for Smoothing and Sharpening frequency domain filters of image.	Apply	2	1, 2, 3, 9, 10, 12
5	Apply Image Segmentation using Neural Networks, Backpropagation Neural Network for Classification	Apply	1, 2	1, 2, 3, 9, 10, 12

TEXT BOOK:

1.Rafael C. Gonzalez, Richard E. Woods, “Digital Image Processing”, 3rd Edition, Pearson, 2010.

REFERENCE BOOKS:

1. Kenneth R. Castleman, “Digital Image Processing”, 1st Edition, Pearson, 2006.

2. Rafael C. Gonzalez, Richard E. Woods, Steven Eddins, “Digital Image Processing using MATLAB”, 2nd Edition, Pearson Education, 2011.

3. D,E. Dudgeon and RM. Mersereau, “Multidimensional Digital Signal Processing”, 1st Edition, Prentice Hall Professional Technical Reference, 1990.

4. William K. Pratt, “Digital Image Processing”, 1st Edition, John Wiley, New York, 2002.

22MC803-MOBILE APPLICATION DEVELOPMENT

Hours per week:4

L	T	P	C
2	2	0	3

PREREQUISITE KNOWLEDGE: Basics of Object-Oriented Programming through JAVA, and XML.

COURSE DESCRIPTION AND OBJECTIVES:

This course helps a student to design effective mobile applications using the Android development environment. The main objective of this course is to create user-friendly applications that involve design of layout, windows components, and multiple screens with one- touch options.

MODULE–1

UNIT–1 **8L+8T+0P=16 Hours**

GETTING STARTED WITH ANDROID

Introduction to Android: versions of android, features of android, architecture, devices in the market, developer community.

UNIT–2 **8L+8T+0P=16 Hours**

ACTIVITIES, FRAGMENTS, INTENTS

Understanding activities, linking activities using intents, fragments, calling built in apps using intents.

GETTING TO KNOW ANDROID UI: Understanding the components of screen -views and view groups, liner layout, absolute layout, table layout relative layout, frame layout, scroll view.

PRACTICES:

- Installation of Android studio, its required tools and Android Virtual Device (Emulator).
- Displaying the welcome message in AVD.
- Creating a basic Activity and applying themes, styles to it.
- Displaying various types of Dialog objects.
- Linking activities with Intents.

MODULE – 2

UNIT–1 **8L+8T+0P=16 Hours**

ANDROID UI DESIGN & DISPLAY ORIENTATION

DISPLAY ORIENTATION: Anchoring views, Resizing and repositioning views, managing changes to screen orientation, utilizing the action bar, Creating UI programmatically.

UNIT–2 **8L+8T+0P=16 Hours**

DESIGNING UI WITH VIEWS:

Basic views:- Text view, Button, Image Button, Edit text, check Box, Toggle button, Radio button, and Radio group views, Progress bar view and Auto complete text view.

PRACTICES:

- Passing data using intent object.
- Usage of Fragments and adding them dynamically to the application.
- Communication between fragments.
- Creating various layouts.
- Displaying Action bar.
- Handling view events.

COURSEOUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Define, explain and understand the android mobile application design models and styles	Apply	1	1, 2,3,4,9,10,12
2	Apply activities, dialog boxes, fragments, intents, views and layouts to android apps	Apply	1,2	1, 2,3,4,9,10,12
3	Analyze various mobile applications during the design of mobile apps	Analyze	1,2	1, 2,3,4,9,10,12
4	Create user-friendly mobile user interfaces and views	Analyze	2	1, 2,3,4,9,10,12

TEXT BOOK:

1. Wei-Meng Lee, “Beginning Android Application Development”, 1 st Edition, John Wiley & Sons, Inc., 2012.

REFERENCE BOOKS:

1. Raimon Refols Montane, Laurence Dawson, “Learning and Android Application Development”, 1 st Edition, PACKT Publishing, 2016.
2. Reto Meier, “Professional Android 4 Application Development”, 3rd Edition, Wrox, 2012.
3. Adam Gerber and Clifton Craig, “Learn Android Studio”, 1st Edition, Apress, 2015.

22MC804-SOFTWARE PROJECT MANAGEMENT

Hours per week:4

L	T	P	C
2	2	0	3

PREREQUISITE KNOWLEDGE: Processes and conditions.

COURSE DESCRIPTION AND OBJECTIVES:

This course is aimed at introducing the primary important concepts of project management related to managing software development projects. They will also get familiar with the different activities involved in Software Project Management. Further, they will also come to know how to successfully plan and implement a software project management activity, and to complete a specific project in time with the available budget.

MODULE – 1

UNIT–1 **8L+8T+0P=16 Hours**

SOFTWARE MANAGEMENT AND ECONOMICS

Conventional Software Management: The Waterfall Model, Conventional Software Management Performance; Evolution of Software Economics - Software economics, Pragmatic software cost estimation, reducing software product size, Improving software processes.

UNIT–2 **8L+8T+0P=16 Hours**

THE OLD AND THE NEW WAY OF PROJECT MANAGEMENT

Principles of Conventional Software Engineering: Improving team effectiveness, improving automation through software environment, achieving required quality; Peer inspections – A pragmatic view, the principles of conventional software engineering, Principles of modern software management, Transitioning to an iterative process.

SOFTWARE MANAGEMENT PROCESS FRAME-WORK: Life cycle phases, The artifact sets, Management artifacts, Engineering artifacts, Pragmatic artifacts; Model-Based Software Architectures - A management perspective and A technical perspective.

MODULE – 2

UNIT–1 **8L+8T+0P=16 Hours**

PROJECT ORGANIZATION AND PLANNING

Iterative Process Planning: Work breakdown structures, Planning guidelines, The cost and schedule estimating process, The iteration planning process, Pragmatic planning, Line-of-Business organizations, Project organizations, Evolution of organizations; Process automation - Automation building blocks, The project environment.

UNIT–2 **8L+8T+0P=16 Hours**

PROJECT CONTROL AND PROCESS INSTRUM-ENTATION

Core metrics: The Seven-Core metrics, Management indicators, Quality indicators, Life-Cycle expectations, Pragmatic software metrics, Metrics automation, Modern project profiles, Next generation software economics, Modern process transitions.

COURSEOUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Identify the different project contexts and suggest an appropriate management strategy	Apply	1	1, 2,3,4,9,10,12
2	Practice the role of professional ethics in successful software development.	Apply	2	1, 2,3,4,9,10,12
3	Identify and describe the key phases of project management.	Analyze	1	1, 2,3,4,9,10,12
4	Determine an appropriate project management approach through an evaluation of the business context and scope of the project.	Analyze	2	1, 2,3,4,9,10,12

TEXT BOOK:

1. Walker Royce, "Software Project Management", 1st Edition, Pearson Education, 2006.

REFERENCE BOOKS:

1. Bob Hughes and Mike Cotterell, "Software Project Management", 5th Edition, Tata McGraw Hill Edition, 2009.
2. Joel Henry, "Software Project Management", 1st Edition, Pearson Education, 2008.
3. Pankaj Jalote, "Software Project Management in practice", 1st Edition, Pearson Education, 2005.

22MC805-FORMAL LANGUAGES AND AUTOMATA THEORY

Hours per week:5

L	T	P	C
3	2	0	4

PREREQUISITE KNOWLEDGE: It is recommended that the candidate has done a course in Data Structures and Algorithms.

COURSE DESCRIPTION AND OBJECTIVES:

This course focuses on the basic theory of Computer Science and formal methods of computation like automata theory, formal languages, grammars and Turing Machines. The objective of this course is to explore the theoretical foundations of computer science from the perspective of formal languages and classify machines by their power to recognize languages.

MODULE–1

UNIT–1 **12L+8T+0P =20 Hours**

INTRODUCTION TO AUTOMATA

Alphabets, Strings and languages, Automata and Grammars, Regular languages, Deterministic finite automata (DFA)-Formal definition, Simplified notation, State transition graph, Transition table, Language of DFA; Nondeterministic finite automata (NFA), NFA with epsilon transition, Language of NFA, Equivalence of NFA and DFA, Minimization of finite automata**.**

UNIT–2 **12L+8T+0P =20 Hours**

REGULAR EXPRESSION

Definition, Operators of regular expression and their precedence, Algebraic laws for Regular expressions, Kleen's Theorem, Regular expression to FA,Kleen's Theorem, Regular expression to FA, DFA to regular expression, Arden theorem, Non regular languages, Pumping lemma for regular languages, Application of pumping lemma, Closure properties of regular languages.

PRACTICES:

- Draw a DFA for the language accepting strings ending with 'abb' over input alphabets $\sum$ = {a, b}
- Draw a DFA for the language accepting strings starting with 'ab' over input alphabets $\sum$ = {a, b}
- Check if the Language L = {w $\in$ {0, 1}* : w is the binary representation of a prime number} is a regular or non-regular language.
- Prove that the Language L = {1^n : n is a prime number} is a non-regular Language.
- Prove that the language L = = {0^n1^n : n >= 0} is not a regular language

MODULE–2

UNIT–1 **12L+8T+0P =20 Hours**

GRAMMAR FORMALISM

Regular Grammars-Right linear and left linear grammars, Equivalence between regular linear grammar and FA; Context free grammar, Definition, Examples, Derivation, Derivation trees.

Push Down Automata (PDA): Description and definition, Instantaneous description, Language of PDA, Acceptance by final state, Acceptance by empty stack, Deterministic PDA.

UNIT–2 **12L+8T+0P =20 Hours**

TURING MACHINES (TM)

Basic model, Definition and representation, Instantaneous Description, Language acceptance by TM, Computable functions, Types of Turing machines, universal TM, Church's Thesis, Recursive and recursively enumerable languages.

PRACTICES:

- Provide an algorithm for converting a left linear grammar to a right linear grammar.
- Give pushdown automata that recognize the following languages
  - $A = \{w \in \{0, 1\}* \mid w \text{ contains at least three } 1^s\}$
  - $B = \{w \in \{0, 1\}* \mid w = w^R \text{ and the length of w is odd}\}$
- Consider the language $L = \{w \in \Sigma^* \mid w \text{ is a string of balanced digits}\}$ over $\Sigma = \{0, 1\}$
- Design Turing Machine for the language $L = \{0^{2^n}, n \geq 0\}$
- Design Turing machines for the following language, The set of strings with an equal number of 0's and 1's

COURSEOUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Understand the basic properties of formal languages and grammars.	Apply	1	1,2
2	Construct automata, regular expression for any pattern.	Analyze	1,2	2,3
3	Acquire concepts relating to the theory of computation and computational models including decidability and intractability.	Analyze	1	8,9
4	Design Turing machines for any language.	Apply	1,2	10,12

TEXT BOOKS:

1. Hopcroft and Ullman, “Introduction to Automata Theory, Languages and Computation”, 2nd Edition, Pearson/Prentice Hall India, 2007.
2. Peter Linz, "An Introduction to Formal Language and Automata", Narosa Pub. House, 2011.

REFERENCE BOOKS:

1. Martin J. C., “Introduction to Languages and Theory of Computations”, 4th Edition, TataMcGraw Hill, 2010.
2. Papadimitrou, C. and Lewis, C.L., “Elements of the Theory of Computation”, 2nd Edition, Pearson/Prentice Hall India, 2009.

22MC806- DESIGN AND ANALYSIS OF ALGORITHMS

Hours per week:4

L	T	P	C
3	0	2	4

PREREQUISITE KNOWLEDGE: Data Structures and Algorithms.

COURSE DESCRIPTION AND OBJECTIVES:

Algorithm design and analysis provide the theoretical backbone of computer science and are a must in the daily work of the successful programmer. The goal of this course is to provide a solid background in the design and analysis of the major classes of algorithms. At the end of the course students will be able to develop their own versions for a given computational task and to compare and contrast their performance.

MODULE – 1

UNIT–1 **9L+0T+6P =15 Hours**

BASICS OF ALGORITHMS

Introduction: Algorithm, Pseudo-code for expressing algorithms; Performance analysis; Asymptotic notations; Recursive Algorithms; Randomized Algorithms; Time complexity by Master's Theorem; Disjoint sets; Graph representations; Connected components and Bi-connected components.

Divide & Conquer and Greedy Methods:

Divide and conquer general method; Greedy general method.

UNIT–2 **15L+0T+10P=25 Hours**

APPLICATIONS OF DIVIDE & CONQUER AND GREEDY METHODS

Applications: Towers of Hanoi Problem; Binary search; Quick sort; Merge sort; and Stassen's matrix multiplication; Max-min problem; Job sequencing with deadlines; Knapsack problem; Minimum cost spanning trees.

PRACTICES:

- State Towers of Hanoi problem. Implement Towers of Hanoi Problem. Repeat the experiment for different values of n (number of disks) by taking n value from a sorted list or array. **(RECURSIVE ALGORITHMS)**
- Implement binary search algorithm to find the existence of a particular element in the list or array, repeat the experiment for different values of n (number of elements in the given list or array). **(DIVIDE AND CONQUER)**
- Sort the given list of elements using the following methods, repeat the experiments for different values of n (number of elements in the given list or array) **(DIVIDE AND CONQUER)**
- Merge sort.
- Quick sort.
- Implement divide and conquer strategy to find the max and min elements of given list or array. Repeat the experiment for different values of n (number of elements in the given list or array). **(DIVIDE AND CONQUER)**

- Implement Strassen's matrix multiplication to multiply two 2X2 matrices. **(DIVIDE AND CONQUER)**
- State the problem of Job sequencing with deadlines and write a program to solve it using Greedy method. **(GREEDY METHOD)**
- State Knapsack problem and write a program to solve the knapsack problem using Greedy approach. **(GREEDY METHOD)**

MODULE – 2

UNIT–1 **6L+0T+4P=10 Hours**

DYNAMIC PROGRAMMING, BACK TRACKING AND BRANCH & BOUND METHODS

Introduction: Dynamic programming general method; back tracking general method; branch and bound general method; introduction to the applications of dynamic programming, back tracking and branch and bound.

UNIT–2 **18L+0T+12P=30 Hours**

APPLICATIONS OF DYNAMIC PROGRAMMING, BACK TRACKING AND BRANCH & BOUND METHODS

Applications: Optimal binary search trees; Matrix chain multiplication; 0/1 knapsack problem; All pairs shortest path problem; Travelling sales person problem; Reliability design problem; n-queen problem; Sum of subsets problem; Graph colouring; Hamiltonian cycles; Traveling sales person problem; 0/1 knapsack problem; Job assignment problem.

PRACTICES:

- State all pairs shortest path problem and write a program to solve it using dynamic programming. **(DYNAMIC PROGRAMMING)**
- State Travelling sales person problem. Find the tour of minimum cost of travelling sales person by using dynamic programming. **(DYNAMIC PROGRAMMING)**
- State the problem of Matrix chain multiplication and write a program to find the cost of optimal order of matrix chain multiplication problem using dynamic programming. **(DYNAMIC PROGRAMMING)**
- State n-queens' problem and write a program to solve the same using backtracking approach. **(BACK TRACKING METHOD)**
- State the problem of sum of subsets and write a program to solve it using backtracking approach. **(BACK TRACKING METHOD)**
- Distinguish Fractional Knapsack and 0/1 Knapsack problem. Write a program to solve knapsack problem using Branch and Bound. **(BRANCH AND BOUND METHOD)**

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Apply various algorithm design paradigms (divide-and-conquer, greedy method, dynamic-programming and backtracking and	Apply	1	1,2

	branch and bound) to solve the given problem.			
2	Analyze best, average and worst-case running times of algorithms using asymptotic analysis.	Analyze	1,2	2,3
3	Determine suitability of algorithm design strategy to develop an algorithm for solving the given novel problem.	Analyze	1	8,9
4	Develop new algorithms for solving the real time applications using different algorithm design strategies.	Apply	1,2	10,12

TEXT BOOK:

1. Ellis Horowitz, Satraj Sahni and Rajasekharam, "Fundamentals of Computer Algorithms", 2nd Edition, Galgotia publications, 2006.

REFERENCE BOOKS:

1. Anony Levitin, "Introduction to Design and Analysis of Algorithms", 3rd Edition, Pearson Education, 2016.
2. Donald Eknuth, "The Art of Computer Programming", Volume 3, 2nd Edition, Addison Wesley Longman Inc,1998.
3. Ronald L. Graham, Donald E. Knuth and Oren Patashnik, "Concrete Mathematics", 2nd Edition, Addison-Wesley Publishing Company,1998.

22MC807-DATA SCIENCE USING PYTHON

Hours per week:5

L	T	P	C
3	0	2	4

PREREQUISITE KNOWLEDGE: Python programming.

COURSE DESCRIPTION AND OBJECTIVES:

This course introduces the basic concepts of data science. This course makes students familiar with python libraries to handle various datasets and the methods for cleaning, transforming and enrichment of data. In addition, this course enables the students to train the applications of data science and perform data transformations.

MODULE–1

UNIT–1 **12L+0T+8P=20 Hours**

INTRODUCTION TO DATA SCIENCE

Steps in doing Data Science - Data Science relation to other -fields- Data Science and Information Science- Computational Thinking - Skills and tools needed to do Data Science - Storing data - Combining bytes into larger structures - Creating data sets – Identifying data problem - Understanding data sources - Exploring data models- Introduction to Big Data.

UNIT–2 **12L+0T+8P =20 Hours**

DATA ACQUISITION

Import data into Pandas from various data sources: Fetching stored data- CSV, Excel, Pdf, text, multiple text files, RDBMS (SQL Tables), pickle and JSON; importing data from clipboards, Working with binary data formats, Web scraping-beautiful soup for reading and parsing of web pages, reading data from XML, reading data from an API, Reading Image files using PIL, read multiple files using Glob.

PRACTICES:

Write code to perform the following operations:

- Load the data stored in different files formats: CSV, Excel, txt, ZIP, database, JSON, pickle.
- Saving data into different files formats: CSV, Excel, txt, ZIP, database, JSON, pickle
- Web Scraping- Use requests module to retrieve data from any website of your interest, improve readability of the extracted data using Beautiful Soup library.
- Reading Image Folders: retrieve images stored in our local folders and visualize few samples from each class.
- Merge multiple data frames, split a data frame into multiple data frames, apply different joins on two data frames.
- Data pre-processing- formatting data to standardize it and make it consistent, normalizing data, grouping data values into bins.

MODULE–2

UNIT–1 **12L+0T+8P=20 Hours**

DATA CLEANING

Sub-Setting, Filtering, And Grouping: Sub-setting the Data Frame, the unique Function, Conditional Selection and Boolean Filtering, Setting and Resetting the Index, The Group By Method, Aggregating

Detecting Outliers and Handling Missing Values: Outlier detection, Missing Values in Pandas, Filling and dropping missing Values in Pandas, Outlier Detection and removing duplicates

UNIT–2 **12L+0T+8P =20 Hours**

DATA ENRICHMENT

Data Integration and transformation: Combining, merging, and joining data sets, string and text processing using regular expressions, Transforming Numerical features – power transformation, binning, banalization, data transformation based on mappings, Encoding Categorical data- One-hot encoding, Ordinal encoding, Label encoding of the target variable; Scaling- Normalization, Standardization, Robust scaling

Learning The Hidden Secrets: Advanced list comprehension and zip function, Date and time data types and tools, Time series basics, Time zone handling.

PRACTICES:

- Apply the following operations on the given csv file.
    - Load data from CSV files .
    - Retrieve first 10, last 10 rows, 3rd Column and a subgroup.
    - Query and index operations on the above data frame.
    - Insert, delete and update your data.
    - Apply aggregate operations.
    - Apply various filters on the data.
    - Group, merge, and aggregate data in the data frames.
- Apply the following operations on the given csv file.
    - Load the csv and convert to data Frame.
    - Identify the total number of missing values.
    - Replace the missing values with a constant, with the Mean of that column, with the mode of that column.
    - Remove missing values on the original csv file.
    - Apply fill options and replace.
- Load a CSV file, change column names, apply required transformations on the data.
- Load a CSV file, apply binning to transform numeric data to categorical data.
- Load a CSV file, encode the categorical values- transforms non-numerical labels into numerical labels, One-Hot encoding, and Binary encoding.
- Merge multiple data frames, split a data frame into multiple data frames, apply different joins on two data frames.
- Data pre-processing- formatting data to standardize it and make it consistent, normalizing data, grouping data values into bins.

- Work with Date Time Format in Python - Time Series Data, Convert integer to Date Time, Extract Date Time into Different Columns, Assemble Date Time from Different Columns.
- Create Pandas Date Time Series using date range () method.
- Load data that contains dates and times.
    - Convert strings to date time.
    - Assemble a date time from multiple columns.
    - Get year, month and day.
    - Get the week of year, the day of week, and leap year.
    - Get the age from the date of birth.
    - Improve performance by setting date column as the index.
    - Select data with a specific year and perform aggregation.
    - Select data with a specific month and a specific day of the month.
    - Select data between two dates.
    - Handle missing values.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Apply data acquisition tools to collect and visualize.	Apply	1	1,2
2	Analyze the data and apply suitable pre-processing and transformation techniques.	Analyze	1,2	2,3
3	Designing standard datasets for data science projects.	Analyze	1	8,9
4	Tool usage for data acquisition and engineering	Apply	1,2	10,12

TEXT BOOKS:

1. Jeffrey S. Saltz, Jeffrey M. Stanton, “An Introduction to Data Science”, SAGE Publications, 2018.
2. Wes McKinney, “Python for data analysis”, 1st Edition, O’Reilly Media, 2012.

REFERENCE BOOKS:

1.Jake VanderPlas, “Python Data Science Handbook”, 1st Edition, O'Reilly, 2016.

2.Joel Grus, “Data Science from Scratch”, 2nd Edition, O Reilly, 2019.

3.Foster Provost & Tom Fawcett, “Data Science for Business”, O Reilly, 2013.

22MC808-MACHINE LEARNING THROUGH PYTHON

Hours per week:5

PREREQUISITE KNOWLEDGE: Python Programming.

L	T	P	**C**
3	0	2	4

COURSE DESCRIPTION AND OBJECTIVES:

The objective of this course is to familiarize the students with some basic learning algorithms and techniques and their applications, as well as general questions related to analyzing and handling large data sets. Several software libraries and data sets publicly available will be used to illustrate the application of these algorithms. The emphasis will be thus on machine learning algorithms and applications, with some broad explanation of the underlying principles.

MODULE–1

UNIT–1 **12L+0T+4P=16 Hours**

INTRODUCTION AND SUPERVISED LEARNING

Machine Learning Fundamentals –Types of Machine Learning - Supervised, Unsupervised, Reinforcement- The Machine Learning process. Terminologies in ML- Testing ML algorithms: Overfitting, Training, Testing and Validation Sets Confusion matrix -Accuracy metrics- ROC Curve- Basic Statistics: Averages, Variance and Covariance, The Gaussian- The Bias-Variance trade off- Applications of Machine Learning.

UNIT–2 **12L+0T+12P=24 Hours**

REGRESSION

Linear Regression – Multivariate Regression- Classification: Linear Discriminant Analysis, Logistic Regression- K-Nearest Neighbor classifier. Decision Tree based methods for classification and Regression- Ensemble methods.

PRACTICES:

- Installation of Python Libraries/tools for Machine Learning.
- Data pre-processing using Python Machine Learning libraries**.**
- Design a model to predict the housing price from Boston Dataset using Multivariate Linear Regression.
- Build a classifier using Logistic Regression, k- Nearest Neighbor / Decision Tree to classify whether the given user will purchase a product or not from a social networking dataset.

MODULE – 2

UNIT–1 **12L+0T+4P=16 Hours**

INTRODUCTION TO CLUSTERING

Clustering: K-Means clustering, Hierarchical clustering - The Curse of Dimensionality - Dimensionality Reduction - Principal Component Analysis - Probabilistic PCA- Independent Components analysis.

Perceptron: Multilayer perceptron- Back Propagation – Initialization, Training and Validation Support Vector Machines(SVM) as a linear and non-linear classifier - Limitations of SVM.

UNIT–2 **12L+0T+12P=24 Hours**

BAYESIAN CONCEPT OF LEARNING

Introduction : Bayesian Networks - Learning Naive Bayes classifiers - Markov Models – Hidden Markov Models Sampling – Basic sampling methods – Monte Carlo - Reinforcement Learning.

PRACTICES:

- Segment a customer dataset based on the buying behavior of customers using K-means/Hierarchical clustering.
- Dimensionality reduction of any CSV/image dataset using Principal Component Analysis.
- Recognition of MNIST handwritten digits using Artificial Neural Network.
- Build an email spam classifier using SVM.
- Classify the given text segment as 'Positive' or 'Negative' statement using the Naïve Bayes Classifier.
- Predict future stock price of a company using Monte Carlo Simulation.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Analyze various machine learning algorithms and terminologies and perform data pre-processing using standard ML library.	Apply	1	1,2
2	Design a predictive model using appropriate supervised learning algorithms to solve any given problem.	Analyze	1,2	2,3
3	Develop an application using appropriate unsupervised learning algorithms for performing clustering and dimensionality reduction.	Analyze	1	8,9
4	Solve complex problems using artificial neural networks and kernel machines.	Apply	1,2	10,12

TEXT BOOKS:

1. Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, "Machine Learning", Pearson Education, 2018.
2. Christopher Bishop, "Pattern Recognition and Machine Learning" Springer, 2011.

REFERENCE BOOKS:

1. Kevin P. Murphy, "Machine Learning: A Probabilistic Perspective", MIT Press, 2012.
2. Stephen Marsland, "Machine Learning –An Algorithmic Perspective", CRC Press, 2009.
3. Andreas C. Muller, "Introduction to Machine Learning with Python: A Guide for Data Scientists", O'Reilly, 2016.

22MC809-DEEP LEARNING

Hours per week:5

L	T	P	C
3	0	2	4

PREREQUISITE KNOWLEDGE: Machine Learning, Python Programming.

COURSE DESCRIPTION AND OBJECTIVES:

This course offers sufficient details required to understand the basic building blocks of various deep learning based models. Especially, focuses on different types of neural network models like feed forward neural networks, convolutional neural networks, recurrent neural networks, and deep auto encoders. During this course the students build, train, and evaluate deep neural network models for various applications in image, text, and speech domains. In addition, throughout this course student will be able to understand hyper parameter tuning and other best-practices to be followed while training deep neural network models.

MODULE–1

UNIT–1 **12L+0T+8P=20 Hours**

EVOLUTION OF DEEP NEURAL NETWORKS

Deep Learning Intuition: History of Deep Learning, what is Deep Learning? Applications of Deep Learning.

Neural Network Basics: McCulloch–Pitts neuron, Perceptron learning rule, Perceptron convergence theorem, Sigmoidal neuron, Multi-layer feed forward neural network, back propagation algorithm, Gradient descent method, Stochastic gradient descent method. Shallow Neural Networks and Deep Neural Networks.

Regularization and optimization for training Deep Models: Optimization methods - Adagrad, Adadelta, RMSProp, Adam; Regularization Methods-Dropout, Drop connect, Batch normalization; Activation functions - Linear, sigmoid, sigmoid, ReLU and variations of ReLU; Loss Function, Improving the training process – Dataset Augmentation, Noise Robustness, Weight Initialization methods, Early stopping, Parameter sharing and tying, bagging and other ensemble methods.

UNIT–2 **12L+0T+8P=20 Hours**

CONVOLUTIONAL NEURAL NETWORKS (CNNs)

Foundations of Convolutional Neural Network, Popular Deep CNN Models: LeNet, AlexNet, VGGNet, ResNet, Google Net and other architectures.

PRACTICES:

- Practice Assignments can be implemented using the Keras / Tensorflow APIs of Python.
- Relevant data sets can be downloaded from standard repositories such as Kaggle/UCI or can be developed by the students.
- Implement Logistic Regression with Neural Network Mindset

- logistic regression classifier for classification.
- Plot the loss over each epoch.
- Plot the accuracy over each epoch.
- Report final Accuracy.

- Implement Shallow Neural Network model:
  - Implement a binary classification neural network with a single and multiple hidden layers.
  - Implement a Multi-class classification neural network with a single and multiple hidden layers.
  - Vary the number of neurons at suitable layers.
- Hyper parameter Tuning of a Neural Network model implemented for hand-written digit classification:
  - Vary the type of activation functions.
  - Choose suitable Loss functions.
  - Vary the number of neurons at suitable layers.
  - Vary Weight Initialization methods.
  - Save the Best Model and load the saved model.
- Building a Deep Neural Network:
  - Implement a multi-class classification neural network with number of layers of your choice.
  - Include Batch Normalization layers.
  - Vary Optimization methods.
  - Add drop out layers.
- Convolutional Neural Network Models.
  - Design a Convolutional neural network with the layers of your choice.
  - Compare the performance by changing the
    - Kernel size.
    - Number of feature maps at each convolutional layer.
    - Stride.
    - Padding.
    - Number of fully connected layers.
- Visualization of CNN Models.
  - Design a Convolutional Neural Network Model for image classification.
  - Plot Model Architecture.
  - Visualize feature maps after training of CNN .
  - Visualize class activation maps.

MODULE–2

UNIT–1 **14L+0T+10P=24 Hours**

DEEP UNSUPERVISED LEARNING

Transfer learning Approaches: Deep Pre-trained architectures- AlexNet, VGG16, VGG19, ResNEt. Use deep Convolutional architectures for feature extraction and fine-tuning tasks.

Deep Unsupervised Learning: Auto encoders- Under Complete Auto encoders, regularized auto encoders, Representation power, layer size and depth, stochastic encoders and decoders, Denoising auto-encoders, Sparse auto encoder, Contractive auto-encoders.

UNIT–2 **10L+0T+6P=16 Hours**

RECURRENT NEURAL NETWORKS

Architecture of an RNN, unfolding of an RNN, Backpropagation through time, Long short term memory (LSTM), Gated recurrent units, Applications- Text Classification, Sentiment Analysis.

PRACTICES:

- **Using Deep pre-trained CNN model for feature extraction:**
  - Extract features from the FC1 of VGG network.
  - Train any traditional ML model like SVM for classification.
  - Repeat the above by considering FC2 of VGG for feature extraction.
- **Fine-tuning Deep pre-trained CNN for Classification:**
  - Fine-tune VGG network for the task under consideration
  - Check the performance by making.
    - all the layers trainable.
    - freezing the initial layers.
    - freezing the entire network except the final layer.
- Design MLFFNN with 3-level stacked auto encoder based pre-training for Black and white image data, display features extracted by different levels of stacked auto encoder at the end of pre-training.
- **Sentiment Analysis**
  - Pre-process the text.
  - Convert the text into word embedding.
  - Implement the classification network using LSTMs/ GRUs
  - Pre-process the text
  - Convert the text into word embedding.
  - Implement the classification network using LSTMs/ GRUs.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Implementation of Deep learning models to solve various real-time problems.	Apply	1	1,2
2	Analyze performance of a deep network and tune its capacity and hyper parameters.	Analyze	1,2	2,3

3	Leveraging tools to Build deep networks and apply them for real word tasks.	Apply	1	8,9
4	Developing core components for deep learning algorithms.	create	1,2	10,12

TEXT BOOKS:

1. Ian Goodfellow and Yoshua Bengio and Aaron, **"**Deep Learning**",** 1st Edition, An MIT Press Book, 2016.
2. Charu C. Aggarwal "Neural Networks and Deep learning" Springer International Publishing, 2018.

REFERENCE BOOKS:

1. Francois Chollet, "Deep learning with python", 1st edition, Manning Publications, 2017.
2. S. Haykin, "Neural Networks and Learning Machines", 3rd edition, Prentice Hall of India, 2011.
3. Josh Patterson and Adam Gibson, "Deep Learning: A Practitioner's Approach", 1st Edition, O'Reilly, 2017.
4. Satish Kumar, "Neural Networks, A Classroom Approach", Tata McGraw -Hill, 2007.

22MC810-FULL STACK TECHNOLOGIES

Hours per week:5

L	T	P	C
3	0	2	4

PREREQUISITE KNOWLEDGE: Web Programming.

COURSE DESCRIPTION AND OBJECTIVES:

This course explores and demonstrate Understand the advantages of front end and back end tech. The objective of this course is to offer the knowledge and skill on the basic principles and concepts in complete web development.

MODULE–1

UNIT–1 **12L+0T+8P=20 Hours**

INTRODUCTION TO WEB PROGRAMMING

Git : An Introduction to Version Control, Git, Command-line Scripting, Basic HTML, CSS. **Bootstrap** - Overview, Environment setup, Precompiled Bootstrap, Source Code, Grid System, Bootstrap CSS Overview, Typography, Code, Tables, Forms, Helper Classes, Responsive Utilities, Glyph icons, Dropdowns, Navigation Elements, Breadcrumb, Pagination, Badges, Progress bars, Plugins Overview, Transition Plugin, Model Plugin, Dropdown Plugin, Scroll spy Plugin, Tab Plugin, Tooltip Plugin, Popover Plugin, Alert Plugin, Button Plugin, Collapse Plugin, Carousel Plugin, Affix Plugin**.**

UNIT–2 **12L+0T+8P=20 Hours**

CLOUD AND DEVOPS

Motivation-Cloud as a Platform, Operations, Deployment Pipeline: Over all Architecture Building and Testing, Deployment, Case study: Migrating to Micro services.

Origin of DevOps - The developers versus operations dilemma, Key characteristics of a DevOps culture, deploying a Web Application, Creating and configuring an account, creating a web server, managing infrastructure with Cloud Formation, Adding a configuration management system.

PRACTICES:

- Configure the web application in Bootstrap.
- Development of web application using DevOps.
- Configure the web server for web application using DevOps.

MODULE–2

UNIT–1 **12L+0T+8P=20 Hours**

DJANGO AND MONGODB

Introduction to Django - Creating the Project, Running the Development Server, Creating the Application, designing a Model, setting up the Database, Setting up the Application, Dynamic Web Sites, Communication, Data Storage, Presentation.

Introduction to Mongo dB - JSON and MongoDB, adopting a Non, relational Approach, opting for Performance vs. Features Running the Database Anywhere, Generating or Creating a Key, Using Keys and Values, Implementing Collections.

UNIT–2 **12L+0T+8P=20 Hours**

ANGULAR JS, NODE JS AND EXPRESS FRAMEWORK

Introduction to Angular JS - Introduction, Features, Angular JS Model-View, Controller, Expression, Directives and Controllers, Angular JS Modules, Arrays, Working with ng-model,
Working with Forms, Form Validation, Error Handling with Forms, Nested Forms with ng-form, Other Form Controls.

Node JS and Express Framework - Introduction, Using the Terminals, Editors, Building a Webserver with Node, The HTTP Module, Views and Layouts, Middleware, Routing, Form Handling with Express, The Request and Response Objects, Handle bars, Comments and Blocks, Polymorphism.

PRACTICES: (Make all your application in your Git account)

- Battleships Multiplayer Gaming Application.
- Angular 2 chess game.
- Build a web application using Bootstrap.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Experiment with Git and use Bootstrap, Django fundamentals concepts to build and deploy robust web applications and apps.	Apply	1	1,2
2	Make use of different data types to design programs involving DevOps practices.	Analyze	1,2	2,3
3	Apply functional, reliable and user-friendly Angular JS programs for a given problem application.	Apply	1	8,9
4	Develop solutions using Mongo DB and Analyze the usage of Node JS and Express.	Create	1,2	10,12

TEXT BOOKS:

1. Len Bass, Ingo Weber and Liming Zhu, "DevOps: A Software Architect's Perspective", Pearson Education, 2016.
2. Adam Freeman – "ProAngular JS", 1st Edition, Apress, 2014.

REFERENCE BOOKS:

1. AgusKurniawan, "AngularJS Programming by Example", 1st Edition, PE Press, 2014.
2. Kyle Banker, Peter Bakkum, Shaun Verch, Dough Garrett, Tim Hawkins, "MongoDB in Action", 2nd Edition, Manning Publications, 2016.
3. Steve Hoberman, "Data Modeling for MongoDB", 1st Edition, Technics Publication, 2014.
4. ShyamSeshadri, Brad Green, "AngularJS: Up and Running: Enhanced Productivity with Structured Web Apps", Apress, 2015.
5. Evan M. Hahn,"Express in Action", 1st Edition, Manning Publications, 2014.

22MC811-MEDICAL IMAGE ANALYSIS

Hours per week:5

L	T	P	C
3	0	2	4

PREREQUISITE KNOWLEDGE: Digital Image Processing.

COURSE DESCRIPTION AND OBJECTIVES:

This course introduces the medical image terminology, medical image modalities, image reconstruction methods, biomedical image analysis, segmentation, image representation and classification.

MODULE–1

UNIT–1 **12L+0T+4P=16 Hours**

BASICS OF CONTINUOUS TIME SIGNALS AND SYSTEMS

Medical Imaging: A Collaborative Paradigm, Medical Imaging Modalities, Medical imaging from Physiology to Information Processing, Understanding Physiology and Imaging Medium, Data Acquisition and Image Reconstruction, Image Analysis and Applications.

Image Formation: Image Coordinate System, 2-D Image Rotation, 3-D Image Rotation and Translation Transformation, Linear Systems, Point Source and Impulse Functions, Probability and Random Variable Functions, Conditional and Joint Probability Density Functions, Independent and Orthogonal Random Variables, Image Formation, PSF and Spatial Resolution, Signal-to-Noise Ratio, Contrast-to-Noise Ratio, Pin-hole Imaging**.**

UNIT–2 **12L+0T+12P=24 Hours**

LTI SYSTEM AND FOURIER SERIES

Medical Imaging Modalities: X-Ray Imaging: X-Ray Imaging, X-Ray Generation, X-Ray 2-D Projection Imaging, X-Ray Mammography, X-Ray CT, Spiral X-Ray CT, Contrast Agent, Spatial Resolution, and SNR.

Medical Imaging Modalities: Magnetic Resonance Imaging: MRI Principles, MR Instrumentation, MRI Pulse Sequences, Spin-Echo Imaging, Inversion Recovery Imaging, Echo Planar Imaging, Gradient Echo Imaging, Flow Imaging, fMRI, Diffusion Imaging, Contrast, Spatial Resolution, and SNR.

PRACTICES:

- Image sampling and quantization.
- Analysis of spatial and intensity resolution of images.
- Intensity transformation of images.
- DFT analysis of images.
- Transforms. (Walsh, Hadamard, DCT, Haar)
- Histogram Processing.

MODULE–2

UNIT–1 **12L+0T+4P=16 Hours**

MEDICAL IMAGING MODALITIES

Ultrasound Imaging: Propagation of Sound in a Medium, Reflection and Refraction, Transmission of Ultrasound Waves in a Multi-layered Medium, Attenuation, Ultrasound Reflection Imaging, Ultrasound Imaging Instrumentation, Imaging with Ultrasound: A-Mode, M-Mode, B-Mode, Doppler Ultrasound Imaging, Contrast, Spatial Resolution, and SNR Image Reconstruction: Radon Transform and Image Reconstruction, The Central Slice Theorem, Inverse Radon Transform, Back projection Method, Iterative Algebraic Reconstruction Methods, Estimation Methods, Fourier Reconstruction Methods.

UNIT–2 **12L+T0+12P=24 Hours**

IMAGE ANALYSIS

Introduction: Adaptive Arithmetic Mean Filter, Image Sharpening and Edge Enhancement, Feature Enhancement Using Adaptive Neighbourhood Processing, Frequency Domain Filtering, Wiener Filtering, Constrained Least Square Filtering, Low-Pass Filtering, High-Pass Filtering, Homomorphic Filtering, Wavelet Transform for Image Processing, Image Smoothing and Enhancement.

PRACTICES:

- Image Enhancement-Spatial filtering.
- Image Enhancement- Filtering in frequency domain.
- Image segmentation – Edge detection, line detection and point detection.
- Basic Morphological operations.
- Basic Thresholding functions.
- Analysis of images with different colour models.
- Analysis of medically acquired images.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Apply medical image processing and image reconstructing for image analysis.	Apply	1	1, 2, 4, 5, 9, 10, 12
2	Evaluate various image formation such as image rotation, transformation and scaling.	Evaluate	1, 2	1, 2, 5, 9, 10
3	Analyze Echo Planar Imaging, Gradient Echo Imaging and Spin-Echo Imaging for image modalities.	Analyse	1, 2	1, 2, 3, 5, 9, 10
4	Analyze ultrasound Imaging for A-Mode, M-Mode, B-Mode.	Apply	2	1, 2, 5, 9, 10, 12

5	Apply different image filtering approach such Wiener Filtering, Constrained Least Square Filtering, Low-Pass Filtering, High-Pass Filtering, Homomorphic Filtering.	Apply	1, 2	1, 2, 3, 4, 5, 9, 10, 12

TEXT BOOKS:

1. Atam P. Dhawan, "Medical Image Analysis", 2nd Edition, John Wiley, 2011.

REFERENCE BOOKS:

1. Klaus D. Toennies, "Guide to Medical Image Analysis Methods and Algorithms", Springer London, 2012.
2. Kevin Zhou, Hayit Greenspan, Dinggang Shen "Deep Learning for Medical Image Analysis" Academic Press, 2017.
3. Kevin Zhou "Medical Image Recognition, Segmentation and Parsing: Machine Learning and Multiple Object Approaches" MICCAI Society book Series, 1st Edition, 2016.

22MC812-BLOCK CHAIN TECHNOLOGIES

Hours per week:5

L	T	P	**C**
3	2	0	4

PREREQUISITE KNOWLEDGE: Object oriented programming.

COURSE DESCRIPTION AND OBJECTIVES:

This course builds understand how block chain systems (mainly Bit coin and Ethereum) work and to securely interact with them. Design and deploy smart contracts and distributed applications. Integrate ideas from block chain technology into their own projects. Demonstrate the foundation of the Block chain technology and understand the processes in payment and funding. Identify the risks involved in building Block chain applications. Review of legal implications using smart contracts. Choose the present landscape of Blockchain implementations and Understand Crypto currency markets. Examine how to profit from trading crypto currencies.

MODULE–1

UNIT–1 **12L+8T+0P=20 Hours**

EVOLUTION OF BLOCKCHAIN AND BLOCKCHAIN CONCEPTS

Introduction, Scenarios, Challenges Articulated, Blockchain, Blockchain Characteristics, Opportunities Using Blockchain, History of Blockchain.

Evolution of Blockchain: Evolution of Computer Applications, Centralized Applications, Decentralized Applications, Stages in Blockchain Evolution, Consortia, Forks, Public Blockchain Environments, Type of Players in Blockchain Ecosystem, Players in Market.

UNIT–2 **12L+8T+0P=20 Hours**

BLOCKCHAIN CONCEPTS

Introduction, Changing of Blocks, Hashing, Merkle-Tree, Consensus, Mining and Finalizing locks, Currency aka tokens, security on Blockchain, data storage on Blockchain, wallets, coding on Blockchain: smart contracts, peer-to-peer network, types of Blockchain nodes, risk associated with Blockchain solutions, life cycle of Blockchain transaction.

MODULE – 2

UNIT–1 **12L+8T+0P=20 Hours**

ARCHITECTING BLOCKCHAIN SOLUTIONS

INTRODUCTION: Obstacles for Use of Blockchain, Blockchain Relevance Evaluation Framework, Blockchain Solutions Reference Architecture, Types of Blockchain Applications. Cryptographic Tokens, Typical Solution Architecture for Enterprise Use Cases, Types of Blockchain Solutions, Architecture Considerations, Architecture with Blockchain Platforms, Approach for Designing Blockchain Applications.

UNIT–2 **12L+8T+0P=20 Hours**

ETHEREUM BLOCKCHAIN IMPLEMENTATION

Introduction, Tuna Fish Tracking Use Case, Ethereum Ecosystem, Ethereum Development, Ethereum Tool Stack, Ethereum Virtual Machine, Smart Contract Programming, Integrated Development Environment, Truffle Framework, Ganache, Unit Testing, Ethereum Accounts, MyEtherWallet, Ethereum Networks/Environments, OpenZeppelin Contracts Hyperledger Blockchain Implementation, Introduction, Use Case – Car Ownership Tracking, Hyperledger Fabric, Hyperledger Fabric Transaction Flow, FabCar.

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Apply source code for Blockchain applications.	Apply	1	1,2
2	Apply Hyperledger Fabric Transaction Flow for blockchain High-Pass Filtering, Homomorphic Filtering.	Analyze	1,2	2,3
3	Analyze life cycle of blockchain transaction.	Apply	1	8,9
4	Analyze peer-to-peer network for Blockchain applications.	create	1,2	10,12
5	Create architecture for Blockchain Platforms.	Apply	2	11,12

TEXT BOOKS:

1. Ambadas, Arshad Sarfarz Ariff, Sham “Blockchain for Enterprise Application Developers”, Wiley, 2020.

2. Andreas M. Antonpoulos, “Mastering Bitcoin: Programming the Open Blockchain”, O’Reilly, 2021.

REFERENCE BOOKS:

1. Joseph Bambara, Paul R. Allen, “Blockchain: A Practical Guide to Developing Business, Law, and Technology Solutions, Mc Graw Hill, 2018.

2. Melanie Swan, “Blockchain: Blueprint for a New Economy”, O’Reilly, 2015.

22MC813-DATA WRANGLING AND VISUALISATION

Hours per week:6

L	T	P	C
2	2	2	4

PREREQUISITE KNOWLEDGE: Python programming.

COURSE DESCRIPTION AND OBJECTIVES:

This course offers several standard steps of the data wrangling process like importing data, tidying data, string processing. In a data science project, more likely that the data is available in raw format, and it is very much essential to transform that raw data into usable format. This course covers the basics of how large data sets are managed to extract meaningful information and converting the raw data to the tidy form which is a critical step for any data scientist. This course imparts knowledge required to understand subtle patterns, trends, and correlations necessary to understand the data. Effective data visualization is an important tool as it allows us to quickly examine large amounts of data, expose trends efficiently, exchange ideas with key players, and influence decisions. This course allows the students to work with various tools for visualization of data from a variety of fields.

MODULE–1

UNIT–1 **8L+8T+8P=24 Hours**

DATA STRUCTURES

NumPy: Creating Arrays, Arrays Operations, Multidimensional Arrays, Arrays transformation, Array Concatenation, Array Math Operations, Multidimensional Array and its Operations, Vector and Matrix operations.

Pandas Data structures- Series and data fames, working with 1D and 2D data- Creation, accessing manipulation, various operations**.**

UNIT–2 **8L+8T+8P=24 Hours**

DATA LOADING,STORAGE,AND FILE FORMATS

Data Acquisition: Loading and saving data from files of Excel, CSV, and text formats; Interacting with HTML and web API's

Data Wrangling: Sub-setting the Data Frame, the unique Function, Conditional Selection and Boolean Filtering, Setting and Resetting the Index, The Group By Method, Aggregating, Detecting Outliers and Handling Missing Values: Outlier detection, Missing Values in Pandas, Filling and dropping missing Values in Pandas, Outlier Detection and removing duplicates.

PRACTICES:

- Create a 2D Numpy array with 24 elements of size 4x6 and retrieve the last three rows, retrieve the first two column values, retrieve the sum of the second row, retrieve the sum of first column, and display the max value index in the array.
- Create a 2D Numpy array with 42 elements of size 7x6, add a new row, Delete an existing column, replace a specific value, and identify how many values are less than given x.
- Create a 1D-array with 64 elements, Reshape the array into 4, 2x8 arrays, also reshape the array into other possible shapes, Convert the data type into float, Split the array into three sub-arrays of same size.

- Create a 2D Numpy array with 35 elements of size 7x5, identify unique values in the array, identify the existence of duplicates, perform conditional replace operations, insert NaNs, replace NaNs.

2	**3**	**4**	**5**	**6**
10	**11**	**12**	**13**	**14**
18	**19**	**20**	**21**	**22**
26	**27**	**28**	**29**	**30**
31	**33**	**34**	**35**	**36**
37	**38**	**39**	**40**	**41**

- Create the following 2D array using Numpy and perform below operations:
  i. Write the code routine to print the masked (gray) colored sub-array.
  ii. Print the maximum of the fifth row.
  iii. Reshape the array. (change columns to rows, rows to columns)
  iv. Extract all the odd number using conditional logic.
  v. Find the column wise mean, std and variance.
- **Load the data stored in different files formats: CSV, Excel, txt.**
- **Saving data into different files formats: CSV, Excel, txt.**
- **Apply the following operations on the given csv file.**
  i. Load data from CSV files.
  ii. Retrieve first 10, last 10 rows, 3rd Column and a subgroup.
  iii. Query and index operations on the above data frame.
  iv. Insert, delete and update your data.
  v. Apply aggregate operations.
  vi. Apply various filters on the data.
  vii. Group, merge, and aggregate data in the data frames.
- **Apply the following operations on the given csv file.**
  i. Load the csv and convert to data Frame.
  ii. Identify the total number of missing values.
  iii. Replace the missing values with a constant, with the Mean of that column, with the mode of that column.
  iv. Remove missing values on the original csv file.
  v. Apply fill options and replace.

MODULE–2

UNIT–1 **8L+8T+8P=24 Hours**

DATA VISUALIZATION

Elements of data visualization, Exploration plots: Scatter plots, Line plots, bar plots, box plots, Error-plots, histograms, Kernel-density-estimation plots, Cumulative frequencies, Error-bars, box-plots, bubble-plot, grouped bar charts, pie charts, Advanced plots: correlation, regression, waffle charts, word clouds, Bi-variate, and multivariate plots.

UNIT–2 **8L+8T+8P=24 Hours**

DEEP DIVE INTO DATA WRANGLING

Data Integration-Combining, merging, and joining data sets, string processing using regular expressions, Random sampling and random reordering of rows and columns; Data Cleaning- selecting numerical and categorical variables, Cleaning the numerical features and categorical features; Transforming Numerical features – power transformation, binning, binarization, data transformation based on mappings, Encoding Categorical data- One-hot encoding, Ordinal encoding, Label encoding of the target variable; Scaling- Normalization and Standardization.

PRACTICES:

- **Visualize data with the help of the following graphical representations**:
- Line plots (b) Bar plots (c) Error Plots (d) Scatter plots (e) KDE Plots (f) Heat Maps (g) Box Plots (h) Pie graph (i) Histogram (j) multiple graphs in single figure (k) saving Figures.

Write code to perform the following operations:

- Load a CSV file, change column names, apply required transformations on the data
- Load a CSV file, apply binning to transform numeric data to categorical data
- Load a CSV file, encode the categorical values- transforms non-numerical labels into numerical labels, One-Hot encoding, and Binary encoding
- Merge multiple data frames, split a data frame into multiple data frames, apply different joins on two data frames
- Data pre-processing- formatting data to standardize it and make it consistent, normalizing data, grouping data values into bins

COURSE OUTCOMES:

Upon successful completion of this course, students will have the ability to:

CO No.	Course Outcomes	Blooms Level	Module No.	Mapping with POs
1	Apply pre-processing, transformation and cleaning techniques on data acquired from multiple sources.	Apply	1	1,2
2	Analyze the given data and apply required transformations to the data.	Analyze	1,2	2,3
3	Designing datasets for machine learning and data science projects.	Apply	1	8,9
4	Tool usage for data wrangling and visualization.	create	1,2	10,12

TEXT BOOKS:

1. Wes McKinney, “Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython”, O’Reilly, 2017
2. Wes McKinney, “Python for data analysis”, 1st Edition, O’Reilly Media, 2012.

REFERENCE BOOKS:

1. Joel Grus, “Data Science from Scratch”, O’Reilly Media Inc., 2015.
2. Cathy O'Neil, Rachel Schutt, “Doing Data Science” O'Reilly Media, 2013.