AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

(Approved by A.I.C.T.E., New Delhi &Permanently Affiliated to JNTU-GV, Vizianagaram)
NAAC Accredited with A+ grade
Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist,Pin-531113

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

ACADEMIC REGULATIONS

COURSE STRUCTURE AND SYLLABUS

For PG-R24

MCA – MASTER OF COMPUTER APPLICATIONS

(Applicable for batches admitted from 2024-2025)

AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

Tamaram (V), Makavarapalem,Narsipatnam (RD), Anakapalle Dist,Pin-531113

AVANTHIINSTITUTEOFENGINEERINGANDTECHNOLOGY
(Autonomous)
(Approved by A.I.C.T.E., New Delhi &Permanently Affiliated to JNTU-GV, Vizianagaram)
NAACAccredited with A+grade
Tamaram (V), Makavarapalem,Narsipatnam (RD), Anakapalle Dist,Pin-531113.www.avanthienggcollege.ac.in,mail: principal@avanthienggcollege.ac.in

ACADEMIC REGULATIONS (R24) for
MASTER OF COMPUTER APPLICATIONS

(Applicable for the students of MCA from the AcademicYear2024-2025onwards)

1. Eligibility for Admissions

a) Admission to the above program shall be made subject to eligibility, qualification and specialization as prescribed by the institute from time to time.

b) Admissions shall be made on the basis of merit rank obtained by the candidates at ICET examination or the qualifying Entrance Test conducted by the APSCHE / Govt. of AP or on the basis of any other order of merit as approved by the JNTU-GV / Institute, subject to reservations as laid down by the Govt. from time to time.

2. Award of MCA Degree

a) A student shall be declared eligible for the award of the MCA Degree, if he pursues a course of study and completes it successfully in not less than two academic years and not more than four academic years.

b) The student shall register for all 80 credits and secure all the 80 credits.

c) The minimum instruction days in each semester are 90.

d) A Student, who fails to fulfill all the academic requirements for the award of the degree within four academic years from the year of their admission, shall forfeit his seat in MCA course.

Credit Definition:

1HourLecture(L)per week	1Credit
1HourTutorial(T)per week	1Credit
1HourPractical (P)per week	0.5Credit
2HoursPractical(Lab) perweek	1Credit

3. Attendance

a) A candidate shall be deemed to have eligibility to write end semester examinations if he has put in the minimum of 75% of attendance in aggregate of all the subjects.

b) A student is eligible to write the Institute examinations if he acquires a minimum of40% in each subject and 75% of attendance in aggregate of all the subjects.

c) Condonation of shortage of attendance up to 10% i.e., 65% and above, and below 75% may be given for a **maximum of TWO times** by the college academic committee.

d) Condonation of shortage of attendance shall be granted only on genuine and valid reasons on representations by the candidate with supporting evidence.

e) Shortageofattendancebelow65%inaggregateshall in **NO case be condoned and not eligible to write their end semester examination of that class.**

f) A candidate shall not be promoted to the next semester unless; he fulfills the attendance requirements of the previous semester.

g) A stipulated fee of Rs 1000/- shall be payable towards condonation fee for shortage of attendance. Students availing condonation on medical ground shall produce a medical certificate issued by the competitive authority.

h) A students hall not be promoted to the next semester unless he satisfies the attendance requirement of the present semester, as applicable. They may seek readmission into that semester when offered next. If any candidate fulfills the attendance requirement in the present semester, he shall not be eligible for readmission into the same class.

4. Evaluation

The performance of the candidate in each semester shall be evaluated subject-wise, with a maximum of 100 marks both for theory and practical's on the basis of continues Internal Exams and End Semester Examination.

Continuous Internal Evaluation:

Theory

(a) For theory subjects, during a semester, there shall be two mid-term examinations. Each mid term examination shall be conducted for a total duration of 120 minutes with 4 questions (without choice) each question for 10marks,and it will be reduced to 30 marks.

(b) The descriptive examination is set with 4 full questions from first two and half units (50% of the syllabus), the student has to answer all questions. In the similar lines, descriptive examination shall be conducted on the rest of the syllabus.

(c) The first mid (Mid-1) marks shall be submitted to the Institute examination section within one week after completion of first mid examination.

(d) The mid marks submitted to the Institute examination section shall be displayed in the concerned college notice boards for the benefit of the students.

(e) If any discrepancy found in the submitted Mid-1 marks, it shall be brought to the notice of institute examination section within one week from the submission.

(f) Second mid examination shall be conducted on the similar lines of Mid-1 and its Mid-2 marks shall also be submitted to Institute examination section within one week after completion of second mid examination and it shall be displayed in the notice boards. If any discrepancy found in the submitted Mid-2 marks, it shall be brought to the notice of institute examination section within one week from the submission.

(g) Internal marks can be calculated with 80% weightage for better of the two MID and 20%weightagefor another MID exam.

(h) With the above criteria, institute examination section will send mid marks of all subjects inconsolidatedformtoalltheconcernedcollegesandsameshallbedisplayedintheconcerned college notice boards. If any discrepancy found, it shall be brought to the notice of institute examination section through proper channel within one week with all proofs. Discrepancies brought after the givendeadlinewillnotbeentertainedunderanycircumstances

End Semester Theory Examination Evaluation :

Theory:

Theendsemesterexaminationsshallbeconductedbytheinstituteexaminationsectionfor70 marks consists of five questions carrying 14 marks each. Each of these questions may contain sub-questions. For each question there will be an “either” “or” choice, which means that there will be two questions from each unit and the student should answere it her of the two questions.

Laboratory Evaluation:

i. **Internal Evaluation:** The internal marks for laboratory are 30 marks and the marks shall be awarded based on the day to day work: 5 marks, Record: 5 marks and the remaining 20marksto be awarded by conducting an internal laboratory test.

ii. **External Evaluation:** For external marks for laboratory are 70 and marks shall be awarded based on the performance in the end laboratory examinations. Laboratory examination must be conducted with two Examiners, one of them being the Laboratory Class Teacher and the second examiner shall be appointed by the institute from the panel of examiners submitted

by the respective college. Laboratory examination must be conducted with a breakup marks of Procedure-20, Experimentation-25, Results-10, Viva-voce-15.

There shall be an internship / industry oriented mini project/ skill development course, one need to complete during year break (i.e., II-Sem to III-Sem) and will be evaluated for 50 marks internally at the end of III Semester by the departmental committee. For skill development course the certificate has to be verified and submitted to the Institute. A candidate has to secure a minimum 50% of marks to be declared successful.

For Socially Relevant Project using design thinking and Employability Skills the evaluation will be for 25 marks internally at the end of the Semester by the departmental committee. A candidate has to secureaminimum50% of marks to be declared successful.

A student shall be permitted to pursue up to a maximum of two elective courses under MOOCs during the programme. Students are advised to register for minimum 8 weeks or more in duration MOOCs courses. Student has to pursue and acquire a certificate for a MOOC course only from the SWAY/NPTEL through online with the approval of Head of the Department in order to earn the 2 credits. The Head of the department shall notify the list of such courses at the beginning of the semester. The details of the MOOCs courses registered by the students shall be submitted to the institute examination center as well as college examination center. The Head of the Department shall appoint a mentor for each of the MOOC subjects registered by the students to monitor the student's assignment submissions given by SWAYAM/NPTEL. The student needs to submit all the assignments given and needs to take final exam at the proctor center. The student needs to earn a certificate by passing the exam. The student will be awarded the credits given in curriculum only by submission of the certificate. In case if student does not pass subjects registered through SWAYAM/NPTEL, the same or alternative equivalent subject maybe registered again through SWAYAM/NPTEL in the next semester with the recommendation of HOD and shall be passed.

A candidate shall be deemed to have secured the minimum academic requirement in a subject if

he secures a minimum of 40% of marks in the End semester Examination and aminimumaggregate of 50% of the total marks in the End Semester Examination and Internal Evaluation taken together.

In case the candidate does not secure the minimum academic requirement in any subject (as specified in4.7) he has to reappear for the End semester Examination in that subject. A candidate shall be given one chance to re-register for each subject provided the internal marks secured by a candidate are less than 50% and has failed in the end examination. In such a case, the candidate must re-register for the subject(s) and secure the required minimum attendance. The candidate's attendance in the re-registered subject(s) shall be calculated separately to decide upon his

eligibility for writing the end examination in those subject(s).In the event of the student taking another chance, his internal marks and end examination marks obtained in the previous attempt are nullified. For re-registration the candidates have to apply to the Institute through the college by paying the requisite fees and get approval from the Institute before the start of the semester in which re-registration is required. At a given time a candidate is permitted to re-register for maximum of two subjects in addition to the subjects of regular semester.

In case the candidate secures less than the required attendance in any re registered subject (s),he/she shall not be permitted to write the End Semester Examination in that subject. He shall again re-register the subject when next offered.

Laboratory examination for MCA courses must be conducted with two Examiners, one of them being the Laboratory Class Teacher or teacher of the respective college and the second examiner shall be appointed by the institute from the panel of examiners submitted by the respective college.

A candidate shall be allowed to submit the project report only after fulfilling the attendance requirements of all the semesters. The viva-voce examination shall be conducted at the end of the coursework (4thsemester).

5 Evaluation of Project/ Dissertation Work

Everycandidateshallberequiredtosubmitathesisordissertationonatopicapproved bytheProject Review Committee.

A Project Review Committee (PRC) shall be constituted with Head of the Department and two other senior faculty members (one will be the guide).

Registration of Project Work: A candidate is permitted to register for the project work after satisfying the attendance requirement of all the subjects, both theory and practical up to III semester.

After satisfying 5.2, a candidate has to submit, in consultation with his project supervisor, the title, objective and plan of action of his project work to the Project Review Committee for approval. The student can initiate the Project work, only after obtaining the approval from the Project Review Committee (PRC) after the III semester end examinations. The duration of the project is for one semester.

If a candidate wishes to change his supervisor or topic of the project, he can do so with the approval of the Project Review Committee (PRC). However, the Project Review Committee(PRC)shallexaminewhetherornotthechangeoftopic/supervisorleadstoa major change of his initial plans of project proposal. If yes, his date of registration for the project work starts from

the date of change of Supervisor or topic as the case may be.

A candidate shall submit his status report in two stages at least with a gap of ONE month between them.

The work on the project shall be initiated at the beginning of the Sixth semester and the duration of the project is one semester. A candidate is permitted to submit Project Thesis only after the approval of PRC not earlier than 20 weeks from the date of registration of the project work. The candidate has to pass all the theory and practical subjects before submission of the Thesis. For the approval of PRC the candidate shall submit the draft copy of thesis to the Principal(through Head of the Department) and shall make an oral presentation before the PRC.

Three copies of the Project Thesis certified by the supervisor shall be submitted to the College along with **plagiarism report (<50%)**.

The thesis shall be adjudicated by one examiner selected by the Institute. For this, the Principal of the College shall submit a panel of 5 examiners, eminent in that field, with the help of the guide concerned and head of the department.

If the report of the examiner is not favorable, the candidate shall revise and resubmit the Thesis, in the time frame as decided by the PRC. If the report of the examiner is unfavorable again, the thesis shall be summarily rejected. The candidate has to re-register for the project and complete

The project within the stipulated time after taking the approval from the Institute.

If the report of the examiner is favorable, a board consisting of the Supervisor, Head of the Department and the examiner approved by the Institute who adjudicated the thesis shall conduct Viva-Voce examination. The Board shall jointly report the candidate's work for a **maximum of100 Marks**. The Head of the Department shall coordinate and make arrangements for the conduct of Viva-Voce examination.

If the report of the Viva-Voce is **unsatisfactory (i.e., <50 marks)**, the candidate shall retake the Viva-Voce examination only after three months. If he fails to get a satisfactory report at the second Viva-Voce examination, the candidate has to re-register for the project and complete the project within the stipulated time after taking the approval from the Institution.

5 Cumulative Grade Point Average(CGPA)

As measure of the student's performance, a 10-point Absolute Grading System using the following Letter Grades and corresponding percentage of marks shall be followed: **A**fter each course is evaluated for 100 marks, the marks obtained in each course will be converted to a corresponding letter grade as given below, depending on the range in which the marks obtained

by the student fall.

Structure of Grading of Academic Performance

Marks Range(Max –100)	Level	Letter Grade	Grade Point
≥90	Superior	S	10
≥80to<89	Excellent	A	9
≥70to<79	Very Good	B	8
≥60to<69	Good	C	7
≥50to<59	Average	D	6
<50	Fail	F	0
-	Absent	AB	0

i) A student obtaining Grade "F" or Grade "Ab" in a subject shall be considered failed and will be required to reappear for that subject when it is offered the next supplementary examination.

ii) For non-credit audit courses, "Satisfactory" or "Unsatisfactory" shall be indicated instead of The letter grade and this will not be counted for the computation of SGPA /CGPA / Percentage.

Computation of Semester Grade Point Average (SGPA) and Cumulative Grade Point Average(CGPA): The Semester Grade Point Average (SGPA) is the ratio of sum of the product of the number of credits with the grade points scored by a student in all the courses taken by a student and the sum of the number of credits of all the courses under gone by a student, i.e.,

$$\mathbf{SGPA=\Sigma(C_i \times G_i)/\Sigma C_i}$$

Where, C_i is the number of credits of the i^{th} subject and G_i is the grade point scored by the student in the i^{th} course.

The Cumulative Grade Point Average(CGPA) will be computed in the same manner considering all the courses under gone by a student overall the semesters of a program,i.e.

$$\mathbf{CGPA=\Sigma(C_i \times S_i)/\Sigma C_i}$$

Where "S_i" is the SGPA of the i^{th} semester and C_i is the total number of credits up to that semester. Both SGPA and CGPA shall be rounded off to 2 decimal points and reported in the transcripts. While computing the SGPA the subjects in whom the student is awarded Zero grade points will also be included.

$$\mathbf{Equivalent\ Percentage=(CGPA-0.75)*10}$$

Grade Point: It is a numerical weight allotted to each letter grade on a10-point scale. Letter Grade: It is an index of the performance of students in a said course. Grades are denoted by the letters S, A,

B, C, D and F.

6 Award of Degree and Class

After a student has satisfied the requirements prescribed for the completion of the program and iseligiblefortheawardofM.C.A.Degreeheshallbeplacedinoneofthefollowingfour classes:

Class Awarded	CGPA to be secured	
First Class with Distinction	**≥ 7.75 (Without any supplementary appearance)**	**From the CGPA secured from 80Credits.**
First Class	**≥ 7.75 (With any supplementary appearance)** **≥ 6.75 and < 7.75 (Without any supplementary appearance)**	
Second Class	**≥ 6.75 and < 7.75 (With any supplementary appearance)** **≥ 6.0 to < 6.75 (Without any supplementary appearance)**	
Pass Class	**≥ 6.0 to < 6.75 (With any supplementary appearance)**	

The secured grade, grade points, status and credits obtained will be shown separately in the memorandum of marks.

7 With-holding of Results

If the student is involved in indiscipline/malpractices/court cases, the result of the student will be withheld.

8 Transitory Regulations

Discontinued or detained candidates are eligible for readmission (with in the duration as mentioned in above item) as and when next offered.

There admitted students will be governed by the regulations under which the candidate has been admitted.

9 Minimum Instruction Days

The minimum instruction days for each semester shall be 90 working days. There shall be no transfer from one college to another within the Constituent Colleges and Units of Jawaharlal Nehru Technological University-GV.

10 General

a) Wherever the words “he”, “him”, “his”, occur in the regulations, they include “she”, “her”, “hers”.

b) The academic regulation should be read as a whole for the purpose of any interpretation.

c) In the case of any doubt or ambiguity in the interpretation of the above rules/regulations, the decision of the Principal/ Dean-Academics of the institution is final.

d) The Institute may change or amend the academic regulations or syllabi at any time and the changes or amendments made shall be applicable to all the students with effect from the dates notified by the Institute.

MALPRACTICES RULES

DISCIPLINARY ACTION FOR/IMPROPER CONDUCT IN EXAMINATIONS

S.No	Nature of Malpractices/Improper conduct	Punishment
	If the candidate:	
1	(a) Possesses or keeps accessible in examination hall, any paper, note book, programmable calculators, Cell phones, pager, palm computers or any other form of material concerned with or related to the subject of the examination (theory or practical) in which he is appearing but has not made use of (material shall include any marks on the body of the candidate which can be used as an aid in the subject of the examination) (b) Gives assistance or guidance or receives it from any other candidate orally or by any other body language methods or communicates through cell phones with any candidate or persons in or outside the exam hall in respect of any matter.	Expulsionfromtheexaminationhallandcancellationoftheperformanceinthatsubject only of all the candidates involved. Incase of an outsider, he will be handed over to the police and a case is registered against him.
2	Has copied in the examination hall from any paper, book, programmable calculators, palm computers or any other form of material relevant to the subject of the examination (theory or practical) in which the candidate is appearing.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted to appear for the remaining examinations of the subjects of

		That Semester/year. The Hall Ticket of the candidate is to be cancelled and sent to the University.
3	Impersonates any other candidate in connection with the examination.	The candidate who has impersonated shall be expelled from examination hall. The candidate is also debarred and forfeits these at. The performance of the original candidate, who has been impersonated, shall be cancelled in all the subjects of the examination (including practical's and project work) already appeared and shall not be allowed to appear for examinations of the remaining subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all External examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with for feiture of seat. If the imposter is an outsider, he will be handed over to the police and a case is registered against him.
4	Smuggles in the Answer book or additional sheet or takes out or arranges to send out the question paper during the examination or answer book or additional sheet, during or after the examination.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all External examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with for feature of seat.
5	Uses objectionable, abusiveor offensive language in the answer paper or in letters to the examiners or writes to the examiner	Cancellation of the performance in that subject.

	Requesting him to award pass marks.	
6	Refuses to obey the orders of the Chief Superintendent/Assistant–Superintendent/ any officer on duty or misbehaves or creates disturbance of any kind in and around the examination hall or organizes a walk out or instigates others to walk out, or threatens the officer-in charge or any person on duty in or outside the examination hall of any injury to his person or to any of his relations whether by words, either spoken or written or by signs or by visible representation, as saults the officer-in-charge, or any person on duty in or outside the examination hall or any of his relations, or indulges in any other act of mis conductor mischief which result in damage to or destruction of property in the examination hall or any part of the College campus or engages in any other act which in the opinion of the officer on duty amounts to use of unfair means or misconduct or has the tendency to disrupt the orderly conduct of the examination.	In case of students of the college, they shallbeexpelledfromexaminationhallsandcancellationoftheirperformanceinthatsubjectandallothersubjectsthecandidate(s)has(have) alreadyappearedand shall not be permitted to appear for the remaining examinations of the subjects of that semester/year. The candidates also are debarred and forfeit their seats. In case of outsiders, they will be handed over to the police and a police case is registered against them.
7	Leaves the exam hall taking away answer script or intentionally tears of the script or any part thereof inside or outside the examination hall.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all External examinations. The continuation of the course by the candidate is subject to the Academic regulations in connection with for feature of seat.
8	Possess any lethal weapon or fire armin the examination hall.	Expulsion from the examination hall and cancellation of the performance in that

		subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/ year. The candidate is also debarred and forfeits the seat.
9	If student of the college, who is no a candidate for the particular examination or any person not connected with the college indulges in any malpractice or improper conduct mentioned in clause 6 to 8.	Student of the colleges expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat. Person(s)who do not belong to the College will be handed over to police and, a police case will be registered against them.
10	Comes in a drunken condition to the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/ year.
11	Copying detected on the basis of internal evidence, such as, during valuation or during special scrutiny.	Cancellation of the performance in that subject and all other subjects the candidate has appeared including practical examinations and project work of that semester/year examinations.
12	If any malpractice is detected which is not covered in the above clauses 1 to 11 shall be reported to the Institute for further action to award suitable punishment	

Malpractices identified by squad or special invigilators

1. Punishments to the candidates as per the above guidelines.
2. Punishment for institutions:(if the squad reports that the college is also involved in encouraging malpractices)
    (i) A show cause notice shall be issued to the college.
    (ii) Impose a suitable fine on the college.
    (iii) Shifting the examination center from the college to another college for aspecific period of not less than one year.

(Dr. R Prasad Rao)
Dean (Academics)&
Member Secretary(AC)

(Dr. C PV N J Mohan Rao)
Chairman
Academic Council

AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY
(Autonomous)
(Approved by A.I.C.T.E., New Delhi &Permanently Affiliated to JNTU-GV, Vizianagaram)
NAAC Accredited with A+ grade
Tamaram (V), Makavarapalem,Narsipatnam (RD), Anakapalle Dist,Pin-531113.
https://avanthienggcollege.ac.in/, E-mail: principal@avanthienggcollege.ac.in

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

Proposed Course Structure: Program– M.C.A (Master of Computer Applications)
(Applicable from the academic year 2024-2025 to 2025-2026)

I Semester- Course Structure **Regulations: R24**

S.No	Course Code	Course Title	Category	Hours per Week			Credits
				Lecture	Tutorial	Practical	
1	24F00HS01	Business Communication	HS	2	0	0	2
2	24F00HS02	Mathematical and Statistical Foundations	HS	3	0	0	3
3	24F00PC01	Computer Organization & Operating Systems	PC	3	0	0	4
4	24F00PC02	Data Structures	PC	3	0	0	3
5	24F00PC03	Object Oriented Programming with JAVA	PC	3	0	0	3
6	24F00PC04	Operating Systems and Linux Lab	PC	0	0	3	1.5
7	24F00PC05	Data Structures Lab	PC	0	0	3	1.5
8	24F00PC06	JAVA Programming Lab	PC	0	0	3	1.5
9	24F00MC01	Socially Relevant Project using Design Thinking	MC	0	0	1	0.5
Total				**15**	**1**	**10**	**20**

Category	Courses	Credits
PC: Professional Core Course	3	10
HS: Humanities, Management and Social Science	2	5
MC: Mandatory Course	1	0.5
LB: Laboratory	3	4.5
Total	9	20

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

. Proposed Course Structure: Program– M.C.A (Master of Computer Applications)
(Applicable from the academic year 2024-2025 to 2025-2026)

II Semester- Course Structure **Regulations: R24**

S.No	Course Code	Course Title	Category	Hours per Week			
				Lecture	Tutorial	Practical	Credits
1	24F00PC07	Database Management Systems	PC	3	0	0	3
2	24F00PC08	Computer Networks	PC	3	0	0	3
3	24F00PC09	Software Engineering and Design Patterns	PC	3	0	0	3
4	24F00PC10	Data Warehousing and Mining	PC	3	0	0	3
5	24F00PE01.1 24F00PE01.2 24F00PE01.3 24F00PE01.4 24F00PE01.5	**Elective-I** • No SQL Databases • Design and Analysis of Algorithms • Mobile Application Development • Artificial Intelligence • Accounting for Managers	PE	3	0	0	3
6	24F00PC11	DBMS Lab	PC	0	0	3	1.5
7	24F00PC12	Computer Networks Lab	PC	0	0	3	1.5
8	24F00PC13	Software Engineering and Design Patterns Lab	PC	0	0	3	1.5
9	24F00MC02	Employability Skills	MC	0	0	1	0.5
10	24F00MC03	Bridge Course (Python Programming To be taken through MOOCs)	MC	0	0	0	0
Total				**15**	**0**	**10**	**20**

Category	Courses	Credits
PC: Professional Core Course	4	12
PE: Professional Elective	1	3
LB: Laboratory	3	4.5
MC: Mandatory Course	2	0.5
Total	10	20

BUSINESS COMMUNICATION

I M.C.A- I SEMESTER

Course Title :**Business Communication**	Course Code: **24F00HS01**
Teaching Scheme (L:T:P): 2:0:0	Credits: 2
Type of Course: Lecture	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

Course Overview: In a Business Communications degree program, students acquire a diverse set of skills that are crucial for effective communication within a business environment. Here are some of the key skills learned, along with brief descriptions:

- Written Communication: Students learn to craft clear, concise, and persuasive written messages tailored to different audiences. This includes writing business reports, emails, press releases, marketing materials, and social media content.

Course Objectives:

1. To acquaint the students with fundamentals of communication, help them honing oral, written and non- verbal communication skills and to transform them as effective communicators.
2. Enable students to develop new perspectives and equip themselves to meet the demands of a fast-changing world where technology and globalization and other forces have dramatically changed the practice of business communication in recent years
3. Enhance proficiency and competencies in verbal and non- verbal communication skills with a holistic long-term perspective
4. Guide the participants to manage cross cultural communication
5. Develop technical communication skills

CO#	Course Outcomes
CO1	To demonstrate the use of basic and advanced business writing skills.
CO2	To produce clear and concise written business documents.
CO3	To employ proper public speaking techniques
CO4	To develop and deliver a formal presentation.
CO5	To progress the interpersonal communications skills those are required for social and business interaction.

COURSE CONTENT (SYLLABUS)

UNIT I: **9 Hours**

Purpose and process of communication: Objectives of Communication-Process of Communication- Types of communication; noise, listening skills-Types of listening, essentials of good listening and tips- Concepts and factors of Personality development- Personality Traits and its Values and ideals. **COs-C01**

UNIT II: **12 Hours**

Managing Organizational Communication: Formal and Informal Communication- Interpersonal and Intrapersonal communication- Role of Emotion in Interpersonal Communication- Barriers to Interpersonal Communication- Exchange Theory-Gateways for Effective Interpersonal Communication **COs-C02**

UNIT III: **10 Hours**

Non-verbal communication and Body Language: Kinesics, Proxemics, Paralanguage, Haptics, handshakes, appropriate body language and mannerisms for interviews-Business etiquettes across different cultures. **COs-C03**

UNIT IV: **12 Hours**

Written communication: mechanics of writing, report writing- business correspondence-Business letter format- Meetings and managing meetings- Resume writing-Formats and Skills. **COs-C04**

UNIT V: **10 Hours**

Presentation skills: prerequisites of effective presentation, format of presentation- Assertiveness – strategies of assertive behavior -Communication skills for group discussion and interviews- Interview Techniques. **COs-C05**

Reference Books:

1) MallikaNawal:"BusinessCommunication",CengageLearning,NewDelhi,2012.
2) Edwin A. Gerloff, Jerry C. Wofford, Robert Cummins Organizational Communication: The key stone to managerial effectiveness.
3) Meenakshi Rama:"*Business Communication*", Oxford University Press, New Delhi
4) C.S.G. Krishnamacharyulu and Dr. Lalitha Rama krishnan, Business Communication, Himalaya Publishing House, Mumbai
5) Paul Turner:"*Organisational Communication*", JAICO Publishing House, New Delhi.
6) Sathya Swaroop Debasish, BhagabanDas""*Business Communication*", PHI Private Limited, New Delhi, 2009.
7) R.K.Madhukar:"Business Communication",VikasPublishingHouse,NewDelhi,2012
8) Kelly M Quintanilla, Shawn T.Wahl:"Business and Professional Communication", SAGE, New Delhi,2012.
9) Sangita Mehta, Neety Kaushish:"Business Communication", University Science Press, New Delhi,2010.
10) Anjali Ghanekar: "Business Communication Skills", Everest Publishing House, New Delhi, 2011.

Web References:

1) http://dx.doi.org/10.31521/modecon.v28(2021)-21.
2) http://dx.doi.org/10.1177/108056998905200113.
3) http://dx.doi.org/10.54783/ijsoc.v6i3.1197.

Mathematical and Statistical Foundations

I M.C.A- I SEMESTER

Course Title: **Mathematical and Statistical Foundations**	Course Code: **24F00HS02**
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

Course Objectives:

1. Understand the mathematical fundamental that is prerequisites for variety of courses like Data mining, Network protocols, analysis of Web traffic, Computer security, Software engineering, Computer architecture, operating systems, distributed systems bioinformatics, Machine learning.
2. Develop the understand in gof the mathematical and logical basis to many modern techniques in computer science technology like machine learning, programming language design, and concurrency.
3. Study various sampling and classification problems.

CO#	Course Outcomes
C01	Apply the basic rules and theorems of probability theory such as Baye's Theorem; determine probabilities that help to solve engineering problems and to determine the expectation and variance of a random variable from its distribution.
C02	Able to perform and analyze of sampling, means, proportions, variances and estimates the maximum likelihood based on population parameters.
C03	Learn how to formulate and test hypotheses about sample means, variances and proportions and to draw conclusions based on the results of statistical tests.
C04	Design various ciphers using number theory.
C05	Apply graph theory for real-time problems like network routing problem.

COURSE CONTENT (SYLLABUS)

UNIT- I: **10 Hours**

Basic Probability and Random Variables: Random Experiments, Sample Spaces Events, the Concept of Probability the Axioms of Probability, Some Important Theorems on Probability Assignment of Probabilities, Conditional Probability Theorems on Conditional Probability, Independent Events, Bayes Theorem or Rule. Random Variables, Discrete Probability Distributions, Distribution Functions for Random Variables, Distribution Functions for Discrete Random Variables, Continuous Random Variables. **COs-CO1**

Self-Learning Topic**:** To understand risk and return on investment.

UNIT-II: **10 Hours**

Sampling and Estimation Theory: Population and Sample, Statistical Inference Sampling With and Without Replacement Random Samples, Random Numbers Population Parameters Sample Statistics Sampling Distributions, Frequency Distributions, Relative Frequency Distributions, Un biased Estimates and Efficient Estimates Point Estimates and Interval Estimates. Maximum Likelihood Estimates. **COs-CO2**

Self-Learning Topic: Computation of Mean, Variance, and Moments for Grouped Data**.**

UNIT-III: **10 Hours**

Tests of Hypothesis and Significance: Statistical Decisions Statistical Hypotheses. Null Hypotheses Tests of Hypotheses and Significance Type I and Type II Errors Level of Significance Tests Involving the Normal Distribution One-Tailed and Two-Tailed Tests, The Chi-Square Test for Goodness of Fit Contingency Tables Yates' Correction for Continuity Coefficient of Contingency. **COs-CO3**

Self-Learning Topic**:** Power of a Test Quality Control Charts Fitting Theoretical Distributions to Sample Frequency Distributions

UNIT-IV: **10 Hours**

Algebraic Structures and Number Theory: Algebraic Systems, Examples, General Properties, Semi Groups and Monoids, Homomorphism of Semi Groups and Monoids, Group, Subgroup, Abelian Group, Homomorphism, Isomorphism. Properties of Integers, Division Theorem, The Greatest Common Divisor, Euclidean Algorithm. **COs-CO4**

Self-Learning Topic**:** Modular Arithmetic (Fermat's Theorem and Euler's Theorem)

UNIT-V: **10 Hours**

Graph Theory: Basic Concepts of Graphs, Sub graphs, Matrix Representation of Graphs: Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs, Multi graphs, Planar Graphs, Euler's Formula, Spanning Trees, Algorithms for Spanning Trees (Problems Only and Theorems witho*ut* Proofs). **COs-CO5**

Self-Learning Topic**:** Graph Coloring and Covering, Chromatic Number.

Text Books:

1) Foundation Mathematics for Computer Science, 1stEdition, John Vince, Springer, 2015.
2) Probability & Statistics, 3rd Edition, Murray R.Spiegel, John J. Schiller and R.Alu Srinivasan, Schaum's Outline Series, Tata McGraw-Hill Publishers, 2018.
3) Probability and Statistics with Reliability, 2nd Edition, K. Trivedi, Wiley,2011
4) Discrete Mathematics and its Applications with Combinatorics and GraphTheory,7th
5) Edition, H.Rosen, Tata McGrawHill,2003

Reference Books:

1. Probability and Computing: Randomized Algorithms and Probabilistic Analysis,1st Edition, M. Mitzenmacher and E.Upfal, 2005.
2. Applied Combinatorics, 6thEdition, AlanTucker,Wiley,2012

Computer Organizations & Operating Systems

I M.C.A- I SEMESTER

Course Title : **Computer Organizations & Operating Systems**	Course Code: **24F00PC01**
Teaching Scheme (L:T:P): 3:1:0	Credits: 4
Type of Course: Lecture + Tutorial	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

COURSE OBJECTIVES:

1. Conceptualize the basics of organization a land architectural issues of a digital computer.
2. Learn the function of each element of a memory hierarchy.
3. Study various data transfer techniques in digital computer.

CO#	Course Outcomes
CO1	Understand the basic organization of computer and different instruction formats and addressing modes
CO2	Analyze the concept of pipelining, segment register and pin diagram of CPU
CO3	Understand and analyze various issues related to memory hierarchy
CO4	Evaluate various modes of data transfer between CPU and I/O devices
CO5	Examine various inter connection structures of multi processors

COURSE CONTENT (SYLLABUS)

UNIT- I: Introduction: **15 Hours**

Basic Structure Of Computers: Computer Types, Functional units, Basic Operational concepts, Bus structures, Software, Performance, multiprocessor and multi computers, Historical perspective.

Machine Instructions and Programs: Numbers, Arithmetic Operations, and c Characters, Memory locations and addresses, Memory operations, Instructions and Instruction sequencing, Addressing Modes. **COs-CO1**

UNIT- II:

Processing Unit Fundamental Concepts: **15 Hours**

Register Transfers Language, Types of Registers, Register transfer Performing an Arithmetic or Logic Operation, Fetching a Word from Memory, Execution of Complete Instruction, Hardwired Control

Micro Programmed Control: Micro instructions, Microprogram Sequencing. **COs-CO2**

UNIT -III: Introduction to Operating System Concept: 15 Hours

Types of Operating Systems, Operating Systems Concepts, Operating System Operations. Operating Systems Structures- Operating System Services, User Operating-System Interface, Introduction to System calls, Types of System Calls

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 COs-CO3

UNIT -IV: Process Synchronization: 12 Hours

The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitor

Principles of deadlock: System Model, Dead lock characterization, Deadlock handling, Deadlock Prevention, Detection and Avoidance. **COs-CO4**

UNIT -V: 15 Hours

Memory Management: Swapping, Contiguous Memory Allocation, Paging, structure of the Page Table, Segmentation Virtual Memory Management- Demand Paging, Page-Replacement Algorithms

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, Disk Scheduling Algorithms First-Come-First-Served (FCFS), Shortest Seek Time First (SSTF), SCAN, and C-SCAN. **COs-CO5**

Text Books:

1. Computer Organization, Carl Hamacher, ZvonksVranesic, SafeaZaky, 5th edition, McGrawHill.
2. Operating System concepts,7th edition, Abraham Siliberschatz, Galvin, JohnWiley & Sons, Inc
3. Advanced Programming in the Unix environment by W. Richard Stevens

Reference Books:

1. Computer Architecture and Organization, John P. Hayes, 3rdEdition, McGrawHill
2. Computer Organization and Architecture, William Stallings 6thEdition, Pearson / PHI
3. Operating Systems, 6th Edition, William Stallings, PHI/ Pearson
4. Unix and Shell Programming by B. M. Harwani, OXFORD University Press.

Web References:

1 https://www.geeksforgeeks.org/computer-organization-and-architecture-tutorials/?ref=lbp

2 https://www.jntufastupdates.com/jntuk-r20-2-1-operating-systems-material/

DATA STRUCTURES
I M.C.A- I SEMESTER

Course Title :**Data Structures**	Course Code: **24F00PC02**
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

Course Overview :
In this course, you'll learn how to optimize your data analysis using data structures — and how to improve performance on common tasks like searching and sorting.

Course Objectives: The objective of this course is to explore basic data structures such as stacks and queues, introduce a variety of data structures such as hash tables, search trees, tries, heaps, graphs, sorting and pattern matching algorithms.

CO#	Course Outcomes
CO1	Implement basic programs by using C concepts.
CO2	Select the data structures that efficiently model the information in a problem
CO3	Assess efficiency trade-offs among different at a structure implementations or combinations
CO4	Implement and know the application of algorithms for sorting and pattern matching.
CO5	Understand and describe the different sorting algorithms, including insertion sort, selection sort, bubble sort, quick sort, and merge sort, and explain their respective principles and use cases

COURSE CONTENT (SYLLABUS)

UNIT-I: Introduction to the 'C' Programming **15 Hours**

Introduction: Character set, Variables and Identifiers, Built-in Data Types, Input/output statements, Variable Definition, Arithmetic operators and Expressions, Constants and Literals, Simple assignment statement, Basic input/output statement, Type Casting and Type def Simple 'C' programs

Conditional Statements and Loops: Decision making within a program, Conditions, Relational Operators, Logical Connectives, if statement, if-else statement, Loops: while loop, do while, for loop, Nested loops, Infinite loops, Switch statement, Break statement, Goto statement.

Self-Learning Topics: Escape Sequences

Arrays, Strings and Functions:

Arrays: One dimensional array: Array manipulation; Searching, Insertion, Deletion of an element from an array; Finding the largest/smallest element in an array; two dimensional arrays with examples.

Strings: Concepts, String Types, String Input / Output functions, String manipulation functions, Null terminated strings as array of characters, Standard library string functions.

Functions: Top-down approach of problem solving, Modular programming and functions, Standard Library of C functions, Prototype of a function: Formal parameter list, Return Type, Function call, Block structure, passing arguments to a Function: call by reference, call by value, Recursive Functions, arrays as function arguments. **COs-C01**
Self-Learning Topics: String Pattern Matching

UNIT-II: Storage Classes & Structures and Unions 15 Hours
Storage Classes: Scope and extent, Storage Classes in a single source file: auto, extern and static, register, Storage Classes in multiple source files: extern and static
Structures and Unions: Structure variables, initialization, structure assignment, nested structure, structures and functions, structures, and arrays: arrays of structures, structures containing arrays, unions, Enumeration.
Self-Learning Topics: How do you pass a structure to a function
Pointers & File Processing
Pointers: Address operators, pointer type declaration, pointer assignment, pointer initialization, pointer arithmetic, functions and pointers, Arrays and Pointers, pointer arrays, pointers and structures, dynamic memory allocation.
File Processing: Concept of a file, streams, text files and binary files, Differences between text and binary files, State of a file, Opening and Closing files, file input/output functions (standard library input/output functions for files), file status functions (error handling), Positioning functions.
COs-C02
Self-Learning Topics: Binary Files and operations on Binary files

UNIT-III: 15 Hours
Data structure: Definition, types of data structures, Recursion Definition, Linear and binary recursion. Preliminaries of algorithms, analysis and complexity .Linear list – singly linked list, Double linked list and circular linked list - implementation, insertion, deletion and searching operations on linear list. **COs-C03**

UNIT-IV: 15 Hours
Stacks-Operations, array and linked representations of stacks, stack applications, Queues-operations, array and linked representations. Hash Table Representation: hash functions, collision resolution- separate chaining, open addressing-linear probing, quadratic probing, double hashing and rehashing, extendible hashing. **COs-C04**

UNIT-V: 15 Hours
Sorting Techniques: Insertion sort, selection sort, exchange-bubble sort, quick sort and merge sort Algorithms. Trees: Binary Trees, terminology, representation and traversals- pre, post & in order traversals. Search Trees: Binary Search Trees, Definition, Implementation, Operations- Searching, Insertion and Deletion. **COs-C05**

Text Books:

1. Let Us C: Authentic Guide to C Programming Language, 17th edition, Yashavant Kanetkar, BPB Publications.
2. Data Structures Using C. 2nd Edition, Reema Thareja, Oxford
3. Data Structures and Algorithm Analysis in C, 2nd edition, Mark AllenWeiss

Reference Books:

1. Data Structures: APseudo code Approach with C, 2ndEdition, R. F.Gilberg and B.A. Forouzan, Cengage Learning.
2. Programming in ANSI C, 5th edition, E. Balaguruswamy, TMH

Web References:

1. http://www.hackerrank.com/domains/datastructures
2. http://www.github.com/topics/data_structures_c
3. http://nptel.ac.in/courses
4. http://www.cslibrary.stanford.edu

OBJECT ORIENTED PROGRAMMING WITH JAVA
I M.C.A- I SEMESTER

Course Title**: Object Oriented Programming With Java**	Course Code: **24F00PC03**
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

COURSE OVERVIEW: Object-oriented programming aims to implement real-world entities like inheritance, hiding, polymorphism etc. in programming. OOP stands for Object-Oriented Programming.
Procedural programming is about writing procedures or methods that perform operations on the data.

COURSE OBJECTIVES:

1. To understand the basic concepts of object oriented programming concepts.
2. To introduce the principles of inheritance and polymorphism and demonstrate how they are related to the design of abstract classes
3. To understand the implementation of packages and interfaces
4. To introduce the concept of multithreading and exception handling
5. To learn and understand the design of Graphical User Interface using swing controls

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Describe the uses OOP concepts
CO2	Apply OOP concepts to solve real world problems
CO3	Distinguish the concept of packages and interfaces
CO4	Demonstrate the exception handing, multithread applications with synchronization
CO5	Design the GUI based applications using AWT and Swings Establishing Java Database Connection

COURSE CONTENT (SYLLABUS)

UNIT-I: **15 Hours**

Object-Oriented Programming: Introduction, Object Oriented paradigm, Basic Concepts of OOP, Benefits of OOP, Applications of OOP
Java Basics: Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments, Programming Style. Data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and costing, simple java program
Classes and objects: concepts of classes, objects, constructors methods, access control, this

keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, string handling. **COs-C01**

UNIT-II: **15 Hours**

Inheritance: Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance- specialization, specification, construction, extension, limitation, combination, Types of Inheritance, benefits of inheritance costs of inheritance. Member access rules, super uses, using final with inheritance, polymorphism, abstract classes.

Packages and Interfaces: Defining, Creating and Accessing a package, Importing packages, differences between classes and interfaces, defining an interface, Implementing interface, applying interfaces variables in interface and extending interfaces. **COs-C02**

UNIT-III: **15 Hours**

Exception handling and Multithreading: Concepts of exception handling, benefits of exception handling, exception hierarchy, usage of try, catch, throws and finally, built in exceptions, creating own exception sub classes. Differences between multi threading and multitasking, thread life cycle, creating threads, synchronizing threads, daemon threads, thread groups.

COs-C03

UNIT-IV: **15 Hours**

Event Handling: Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes, inner classes. The AWT class hierarchy , user- interface components- labels, button, canvas, scrollbars, text components, check box, check box groups, choices, list panes- scroll pane, dialogs, menu bar, graphics, layout manager- layout manager types- boarder, grid, flow, card and grid bag. **COs-C04**

UNIT-V: **15 Hours**

Swings: Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing- JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons-The JButton class, Check boxes, Radio Buttons, Combo boxes, Tabbed panes, Scroll panes, Trees and Tables.

JDBC Fundamentals: JDBC Overview, JDBC Components, JDBC Driver Registration, Establishing Connections. **COs-C05**

Text Books:

1) Java-The complete reference,7/e, Herbert Schildt, TMH
2) JAVA: How to program, 8/e, Dietal , Dietal,PHI
3) Introduction of programming with JAVA, S.Dean,TMH
4) Introduction to Java programming, 6/e, Y.Daniel Liang, Pearson

Reference Books:

1) Core Java 2, Vol 1(Vol 2) Fundamentals(Advanced), 7/e, Cay.S.Horstmann, Gary Cornell, Pearson
2) Big Java2, 3/e, Cay.S. Horstmann, Wiley
3) Object Oriented Programming through Java, P.Radha Krishna, University Press
4) JAVA& Object Orientation an Introduction, 2/e, John Hunt, Springer

5) Introduction to JAVA Programming, 7/e, Y. Daniel Liang, Pearson. , TMH

Web References:

1. https://www.codecademy.com/courses/learn-java/lessons/hello-world-java/exercises/introduction-to-java
2. https://www.w3schools.com/java/java_variables.asp
   https://www.geeksforgeeks.org/java/

OPERATING SYSTEMS and LINUX LAB

I M.C.A- I SEMESTER

Course Title : **Operating Systems and Linux Lab**	Course Code: **24F00PC04**
Teaching Scheme (L:T:P): 0:0:3	Credits: 1.5
Type of Course: Practicals	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

Course Overview:

- Introduction to Linux. Introduction to Linux Operating System. Linux Full Form. what are …
- Getting Started with Linux. How to install and Run Linux Software in Windows 10 using …
- Basic Linux Commands: Is cp touch. man. cal. pwd mv ln grep wc mkdir rm cat. echo df cd …
- Linux File System. Linux File Hierarchy Structure. Linux Directory Structure.

Course Objectives:

This Course will enable students to implement CPU scheduling algorithms, Disk scheduling algorithms, Execute different types of Linux commands and Write shell scripts.

CO#	Course Outcomes
CO1	Implement various CPU scheduling algorithms and compare results
CO2	Implement various disk scheduling algorithms and compare results
CO3	Implement page replace algorithms
CO4	Implement various memory management techniques.
CO5	Execute basic Linux commands

Developing the following programs:

Operating Systems Lab

1. 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()

2. Simulate The Following
   a) Multiprogramming with A Fixed Number Of Tasks (MFT)
   b) Multiprogramming with A Variable Number Of Tasks (MVT)
3. Write a program to implement first fit, best fit and worst fit algorithm for memory management.
4. Simulate Bankers Algorithm for Dead Lock Avoidance
5. Simulate Bankers Algorithm for Dead Lock Prevention.

Linux Lab

1. Study of Unix/Linux general purpose utility commands
2. Study of Bash shell, Bourne shell and C shell in Unix/Linux operating system.
3. C program to emulate the UNIX ls –l command
   a. Write a Shell program to check whether given number is prime or not.

4. Write a shell script which will display Fibonacci series up to the given range.
5. Write a shell script to check whether the given number is Armstrong or not.
6. Write a shell script to the calculate the value of
7. Write a shell script to accept student number, name, marks in 5 subjects.
   Find total, average and grade using the following rules:
   i. Avg>=80 then grade A
   ii. Avg<80&&Avg>=70 then grade B
   iii. Avg<70&&Avg>=60 then grade C
   iv. Avg<60&&Avg>=50 then grade D
   v. Avg<50&&Avg>=40 then grade E
8. Write a shell script to find minimum and maximum elements in the given list of elements.
9. Write a shell program to check whether the given string is palindrome or not.
10. Write a shell program to print sum, avg of students marks list
11. Write a shell script to check whether the given input number is prime or not.

Data Structures Lab
I M.C.A- I SEMESTER

Course Title : **Data Structures Lab**	Course Code: **24F00PC05**
Teaching Scheme (L:T:P): 0:0:3	Credits: 1.5
Type of Course: Practicals	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

COURSE OBJECTIVES:

1. Design and implement various data structures.
2. Implement operations like searching, insertion, and deletion, traversing mechanism
3. Develop applications using data structure algorithms.

At the end of the course, student will be able to

CO#	Course Outcomes
CO1	Implement various basic data structures and its operations
CO2	Apply sorting and searching algorithms to given numbers
CO3	Implement various tree operations.
CO4	Implement various graphs algorithms.
CO5	Develop applications using various data structures.

List of Experiments

Experiment 1:

a) Write a program in C to display the n terms of even natural number and their sum.
b) Write a program in C to display the n terms of harmonic series and their sum. 1 + 1/2 + 1/3 + 1/4+ 1/5 ... 1/n terms.
c) Write a C program to check whether a given number is an Armstrong number or not.
d) Write a C program to calculate the factorial of a given number.

Experiment 2:

a) Write a program in C for multiplication of two square Matrices.
b) Write a program in C to find transpose of a given matrix.

Experiment 3:

a) Write a program in C to check whether a number is a prime number or not using the function.
b) Write recursive program which computes the n^{th} Fibonacci number, for appropriate values of n.
c) Write a program in C to add numbers using call by reference.

Experiment 4:

a) Write a program in C to append multiple lines at the end of a text file.
b) Write a program in C to copy a file in another name.

Experiment 5:

a) Write recursive program for the following
b) Write recursive and non recursive C program for calculation of Factorial of an integer.
c) Write recursive and non recursive C program for calculation of GCD (n, m)
d) Write recursive and non recursive C program for Towers of Hanoi: N disks are to be transferred from peg S to peg D with Peg I as the intermediate peg.

Experiment 6:

a) Write C program that use both recursive and non recursive functions to perform Linear search for a Key value in a given list.
b) Write C program that use both recursive and non recursive functions to perform Binary search for a Key value in a given list.

Experiment 7:

a) Write C program that implement stack (its operations) using arrays.
b) Write C program that implement stack (its operations) using Linked list.

Experiment 8:

a) Write a C program that uses Stack operations to convert infix expression into postfix expression.
b) Write C program that implement Queue (its operations) using arrays.
c) Write C program that implement Queue (its operations) using linked lists.

Experiment 9:

a) Write a C program that uses functions to create a singly linked list and perform various operations on it.

Experiment 10:

a) Write a C program to store a polynomial expression in memory using linked list and perform polynomial addition.

Experiment 11:

a) Write a recursive C program for traversing a binary tree in preorder, inorder and post order.
b) Write a non recursive C program for traversing a binary tree in preorder, inorder and post order.

Experiment 12:

a) Implementation of Hash table using double hashing as collision resolution function.

Experiment 13:

a) Implementation of Binary Search trees- Insertion and deletion.

Experiment 14:

a) Write C program that implement Bubble sort, to sort a given list of integers in ascending order.

b) Write C program that implement Quick sort, to sort a given list of integers in ascending order.
c) Write C program that implement merge sort, to sort a given list of integers in ascending order.

Exercise Problems:

1. Write a program to declare variables of different data types (int, float, char, double) and initialize them. Print the values of these variables.
2. Create a C program that reads two integers from the user and performs addition, subtraction, multiplication, and division. Display the results.
3. Write a program that demonstrates type casting between different data types. For example, cast a float to an int and display the result.
4. Implement a C program that takes user input for a name and age, and prints a personalized message.
5. Write a program that uses if-else statements to check if a given number is positive, negative, or zero.
6. Create a program that prints the first 10 natural numbers using a for loop.
7. Write a program that prints a multiplication table (1 to 10) using nested loops.
8. Implement a simple calculator using a switch statement that can perform addition, subtraction, multiplication, and division based on user input.
9. Write a program that initializes an array of 5 integers, performs insertion of a new element, searches for an element, and deletes an element.
10. Implement a program that finds the largest and smallest elements in an array of integers.
11. Write a program that reverses a given string and prints the reversed string.
12. Create a program that uses standard library functions like strlen, strcmp, and strcpy to perform string operations.
13. Implement a function that calculates the factorial of a number using recursion.
14. Write a function that accepts two integers and returns their sum. Use this function in a main program to display the result.
15. Demonstrate the difference between call by value and call by reference using appropriate functions.
16. Write a program that demonstrates the use of auto, static, extern, and register storage classes.
17. Define a structure to store student information (name, roll number, and marks). Write a program to input and display student details using this structure.
18. Create a structure for an employee with nested structures for address details. Write a program to input and display employee information.
19. Write a program that demonstrates the use of unions by storing and displaying different types of data (e.g., int, float, char) using a single memory location.
20. Implement a singly linked list with operations to insert, delete, and search for nodes.
21. Create a doubly linked list and implement insertion and deletion operations from both ends.
22. Write a program that creates a circular linked list and performs insertion and deletion operations.
23. Implement a recursive function to calculate the factorial of a number.
24. Write a recursive function to generate the nth Fibonacci number.

25. Implement a stack using an array and perform push, pop, and display operations.
26. Write a program to implement a stack using a linked list.
27. Create a queue using an array and implement enqueue, dequeue, and display operations.
28. Implement a queue using a linked list with basic operations.
29. Implement a hash table using separate chaining for collision resolution.
30. Write a program that demonstrates hash table implementation with open addressing (linear probing).
31. Implement the following sorting algorithms in C: insertion sort, selection sort, bubble sort, quick sort, and merge sort.
32. Write a program that compares the efficiency of different sorting algorithms using randomly generated data.
33. Implement a binary tree with operations for insertion, deletion, and traversal (pre-order, in-order, post-order).
34. Create a binary search tree (BST) and perform search, insertion, and deletion operations.
35. Write a program that demonstrates different tree traversal methods (pre-order, in-order, post-order) on a binary tree.

Text Books:

1) Let Us C: Authentic Guide to C Programming Language, 17th ed., Yashavant Kanetkar, BPB Publications.
2) Data Structures Using C. 2nd Edition, Reema Thareja, Oxford
3) Data Structures and Algorithm Analysis in C, 2nd ed, Mark Allen Weiss

Reference Books:

1) Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B.A. Forouzan, Cengage Learning.
2) Programming in ANSI C, 5th ed, E. Balaguruswamy, TMH

JAVA PROGRAMMING LAB

I M.C.A- I SEMESTER

Course Title : **Java Programming Lab**	Course Code: 24F00PC06
Teaching Scheme (L:T:P): 0:0:3	Credits: 1.5
Type of Course: Practicals	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

COURSE OBJECTIVES:

1. To understand how to design, implement, test, debug, and document programs that use basic data types and computation, simple I/O, conditional and control structures, string handling and functions.
2. To understand the importance of Classes & objects along with constructors, Arrays and Vectors.
3. Discuss the principles of inheritance, interface and packages and demonstrate though problem analysis assignments how they relate to the design of methods, abstract classes and interfaces and packages.
4. To understand importance of Multi-threading & different exception handling mechanisms.
5. To learn experience of designing, implementing, testing, and debugging graphical user interfaces in Java using applet and AWT that respond to different user events.
6. To understand Java Swings for designing GUI applications based on MVC architecture.

CO#	Course Outcomes
CO1	Apply OOP concepts to solve real-world problems
CO2	Implement different forms of inheritance
CO3	Create packages and to reuse them.
CO4	Implement multi threaded programs using synchronization concepts
CO5	Create user defined exceptions and Design GUI applications using AWT and SWINGS.

List of Experiments:

1. Write a Java Program that uses both recursive and non recursive functions to print the nth value of the Fibonacci sequence.
2. Write a Java Program that prompts the user for an integer and then prints out all the prime numbers up to that Integer.
3. Write a Java Program that checks whether a given string is a palindrome or not. Ex. MALAYALAM is a palindrome.
4. Write a Java Program for sorting a given list of names in ascending order.
5. Write a Java Program that illustrates how runtime polymorphism is achieved.
6. Write a Java Program to create and demonstrate packages.
7. Write a Java Program, using String Tokenize class, which reads a line of integers and then displays each integer and the sum of all integers.

8. Develop a Java application that demonstrates comprehensive file handling operations. The application should perform the following tasks:
   - Create a new file
   - Read the contents of the file and display it on the console.
   - Append additional text to the existing content of the file.
   - Copy the contents of the original file to a new file.
   - Delete the original file
9. Write a Java Program that displays the number of characters, lines and words in a text/text file.
10. Design a Java Swing form with two text fields, a "Submit" button, and a label. When the "Submit" button is clicked, the text from the two text fields should be concatenated and displayed in the label. Implement the event handling for the button click.
11. Write a Java Program for handling mouse events.
12. Write a Java Program demonstrating the life cycle of a thread.
13. Write a Java Program that lets users create Pie charts. Design your own user interface (with Swings & AWT).
14. Write a Java Program to implement a Queue, using user defined Exception Handling (also make use of throw, throws).
15. Write a Java program to establish a connection to a database and verify the connection.

SOCIALLY RELEVANT PROJECT USING DESIGN THINKING
I M.C.A- I SEMESTER

Course Title : **SOCIALLY RELEVANT PROJECT USING DESIGN THINKING**	Course Code: **24F00MC01**
Teaching Scheme (L:T:P): 0:0:1	Credits: 0.5
Type of Course: SOC	
Continuous Internal Evaluation: 0 Marks	Semester End Exam: 25 Marks
Pre requisites:	

COURSE OBJECTIVES:

- Build mind sets & foundations essential for designers
- Learn about the Human-Centered Design methodology and understand the in real-world applications
- Use Design Thinking for problem solving methodology for investigating ill defined problems.

  Under go several design challenges and work to wards the final design challenge

At the end of the course, student will be able to

CO#	Course Outcomes
CO1	Project Stream1:Electronics,Robotics,IOT and Sensors
CO2	ProjectStream2:Computer Science and IT Applications
CO3	ProjectStream3:Mechanical and Electrical tools
CO4	ProjectStream4:Eco-friendly solutions for waste management, infrastructure, safety, alternative energy sources, Agriculture, Environmental science and other fields of engineering.

How to Pursue The Project Work?

- The first part will be learning-based-masking students to embrace the methodology by exploring all the phases of design thinking through the wallet/ bag challenge and podcasts.
- The second part will be more discussion-based and will focus on building some necessary skills as designers and learning about complementary material for human-centered design.
- The class will then divide into teams and they will be working with one another for about 2 – 3 weeks. These teams and design challenges will be the basis for the final project and final presentation to be presented.
- The teams start with **Design Challenge** and go through all the phases more in depth from coming up with the right question to empathizing to ideating to prototyping and to testing.
- Outside of class, students will also be gathering the requirements, identifying the challenges, usability, importance etc
- At the end, Students are required to submit the final reports, and will be evaluated by the faculty.

Tasks to be done:

Task 1:Every one is a Designer

- Understand class objectives &harness the designer mindset Task 2: The Wallet/Bag Challenge and Podcast
- Gain a quick introduction to the design thinking methodology
- Go through all stages of the methodology through as impled design challenge
- Podcast: Observe, Listen and Engage with the surrounding environment and identify a design challenge.

Task 3:Teams&Problems

- Start Design Challenge and learn about teams &problems through this
- Foster team collaboration, find inspiration from the environment and learn how to identify problems

Task 4: Empathizing

- Continue Design Challenge and learn empathy
- Learn techniques on how to empathize with users
- Go to the field and interview people in their environments
- Submit ActivityCard Task 5: Ideating
- Continue Design Challenge and learn how to brain storm effectively
- Encourage exploration and foster spaces for brain storming
- Submit ActivityCard Task 6: Prototyping
- Continue Design Challenge and learn how to create effective prototypes
- Build tangible models and use them as communication tools
- Start giving constructive feedback to classmates and teammates
- Submit ActivityCard Task 7: Testing
- Finish Design Challenge and iterate prototypes and ideas through user feedback
- Evolve ideas and prototypes through user feedback and constructive criticism
- Get peer feedback on individual and group performance
- Submit ActivityCard Task 8:
- Final Report Submission and Presentation

Note: The colleges may arrange for Guest Speakers from Various Design Fields: Graphic Design, Industrial Design, Architecture, Product Design, Organizational Design, etc to enrich the students with Design Thinking Concept.

References:

1. TomKelly,TheArtofInnovation:LessonsinCreativityFromIDEO,America'sLeading Design Firm (Profile Books, 2002)
2. Tim Brown, Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation (Harper Business, 2009)
3. Jeanne Liedtka, Randy Salzman, and DaisyAzer,Design Thinking for the GreaterGood: Innovation in the Social Sector (Columbia Business School Publishing, 2017)

Other Useful Design Thinking Frameworks and Methodologies:

- Human-Centered Design Toolkit(IDEO);https://www.ideo.com/post/design-kit
- Design Thinking Boot Camp Bootleg (Stanford D-School); https://dschool.stanford.edu/resources/the-bootcamp-bootleg
- Collective Action Toolkit (frogdesign); https://www.frogdesign.com/wpcontent/uploads/2016/03/CAT_2.0_English.pdf
- Design Thinking for Educators(IDEO);https://designthinkingforeducators.com/

DATA BASE MANAGEMENT SYSTEMS
I M.C.A- II SEMESTER

Course Title: **DATA BASE MANAGEMENT SYSTEMS**	
Teaching Scheme (L:T:P): 3:0:0	Course Code: **24F00PC07**
Type of Course: Lecture	Credits: 3
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

COURSE OVERVIEW:

DBMSs facilitate data storage, retrieval, and management, providing essential support for decision-making and operational processes in organizations.

They enable scalability, security, and integrity of data, crucial for modern applications and business environments.

COURSE OBJECTIVES:

1. Explain the concept of databases, database management systems, database structures and how they work.
2. Make use of Entity-Relationship Modeling and Relational Modeling for creating simple databases from the real world scenarios.
3. Write relational algebra and structured query language (SQL) statements.
4. Normalize a database using Normalization Rules.
5. Discuss the issues associated with Transaction Management and Recovery, Tree Structured and Hash-Based Indexing.

At the end of the course, the students will be able to:

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Illustrate the concept of databases, database management systems, database languages, database structures and their work
CO2	Apply ER modeling and Relational modeling for designing simple databases.
CO3	Summarize the concepts related to relational model and SQL and Write database queries using relational algebra and structured query language.
CO4	Design and develop databases from the real world by applying the concepts of Normalization.
CO5	Outline the issues associated with Transaction Management and Recovery, Tree Structured and Hash-Based Indexing

COURSE CONTENT (SYLLABUS)

UNIT-I: **15 Hours**

Databases and Database Users: Introduction, Characteristics of the Database Approach, Actors on the Scene, Workers behind the scene, Advantages of the using the DBMS Approach.

Database System Concepts and Architecture: Data Models, Schemas and Instances, Three Schema architecture and Data Independence, Database Languages and Interfaces, Centralized and Client/Server Architecture for DBMS, Classification of Database Management Systems.

COs-C01

UNIT-II: **15 Hours**

Introduction to Database Design: Database Design and ER Diagrams, Entities, Attributes and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design with the ER Model, Conceptual Design for Large Enterprises

Relational Model: Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity Constraints, Querying Relational Data, Logical Database Design: ER to Relational, Introduction to Views, Destroying/Altering Tables and Views. **COs-C02**

UNIT-III: **15 Hours**

Relational Algebra: Selection and Projection, Set Operations, Renaming, Joins, Division, More Examples of Algebra Queries.

Relational Calculus : Types of Relational Calculus, Functional Dependency, Inference Rule

SQL: Queries, Constraints, Triggers: The Form of a Basic SQL Query, UNION, INTERSECT and EXCEPT, Nested Queries, Aggregate Operators, Null Values, Complex Integrity Constraints in SQL, Triggers and Active Databases, Designing Active Databases.

COs-C03

UNIT-IV: **15 Hours**

Introduction to Normalization Using Functional and Multivalued Dependencies: Informal Design Guidelines for Relation Schema, Functional Dependencies, Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form. **COs-C04**

UNIT-V: **15 Hours**

Transaction Management and Concurrency Control: Transaction Concept, A Simple Transaction Model, Storage Structure, ACID Properties, Serializability, Transaction Isolation Levels, Concurrency Control, Lock-Based Protocols, Validation-Based Protocols. **COs-C05**

Note: For Practical Examples Please Go Through Reference 1

Text Books:

1) Database Management Systems, 3/e, Raghurama Krishnan, Johannes Gehrke, Mc Graw-Hill
2) Database System Concepts, 6/e, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Mc Graw-Hill
3) Database Systems, 6/e Ramez Elmasri, Shamkant B. Navathe, Pearson

Reference Books:

1) Database Systems, 9/e, Carlos Coronel, Steven Morris, Peter Rob, Cengage
2) Introduction to Database Systems, 8/e, C J Date, Pearson

Web references:

1. https://www.geeksforgeeks.org/dbms/
2. https://www.javatpoint.com/dbms-tutorial

COMPUTER NETWORKS
I M.C.A- II SEMESTER

Course Title: **COMPUTER NETWORKS**	Course Code: **24F00PC08**
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

COURSE OVERVIEW: Computer networks enable communication, resource sharing, and collaboration, making them essential for personal, educational, and business environments. They form the backbone of the modern digital world, supporting everything from social media to cloud services.

COURSE OBJECTIVES:

1. Understands the fundamental concepts of computer networking and OSI Reference model.
2. Familiarize the student with the basic taxonomy and terminology of the computer networking area.
3. Learn and understand the advanced networking concepts, preparing the student for entry advanced courses in computer networking.
4. Develop and gain expertise in some specific areas of networking such as the design and maintenance of individual networks.

At the end of the course, the students will be able to:

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Explain the network architecture, TCP/IP and OSI reference models
CO2	Identify and understand various techniques and modes of transmission
CO3	Describe routing and congestion in network layer with routing algorithms and classify IPV4 addressing scheme
CO4	Discuss the elements and protocols of transport layer
CO5	Develop network security and define various protocols such as FTP, HTTP, Telnet, DNS

COURSE CONTENT (SYLLABUS)

UNIT-I: **10 Hours**

Introduction: Network Topologies WAN, LAN, MAN. Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models. **Physical Layer –**Introduction to physical layer-Data and Signals, Periodic analog signals, digital signals, transmission impairment, ,Data rate limits, performance -Introduction to Guided Media-Twisted-pair cable, Coaxial cable and Fiber optic cable and Unguided media: Wireless-Radio waves, microwaves, infrared. **COs-C01**

UNIT-II: **15 Hours**

The Data Link Layer - Services Provided to the Network Layer – Framing – Error Control – Flow Control, Error Detection and Correction – Error-Correcting Codes – Error Detecting Codes. **Elementary Data Link Protocols-** A Utopian Simplex Protocol-A Simplex Stop and Wait

Protocol for an Error free channel-A Simplex Stop and Wait Protocol for a Noisy Channel, Sliding Window Protocols-A One Bit Sliding Window Protocol-A Protocol Using Go-Back-N- A Protocol Using Selective Repeat. **COs-C02**

UNIT-III: **15 Hours**

The Network Layer Design Issues – Store and Forward Packet Switching-Services Provided to the Transport layer- Implementation of Connectionless Service-Implementation of Connection Oriented Service- Comparison of Virtual Circuit and Datagram Networks, Routing Algorithms-The Optimality principle-Shortest path, Flooding, Distance vector, Link state, Hierarchical. Congestion Control algorithms-General principles of congestion control, Congestion prevention polices, Approaches to Congestion Control-Traffic Aware Routing- Admission Control-Traffic Throttling-Load Shedding. **COs-C03**

UNIT-IV: **10 Hours**

Internet Working: How networks differ- How networks can be connected- Tunneling, internetwork routing-, Fragmentation, network layer in the internet – IP protocols-IP Version 4 protocol-, IP addresses-, Subnets-IP Version 6-The main IPV6 header- Internet control protocols-ICMP-ARP- DHCP. **COs-C04**

UNIT-V: **15 Hours**

The Transport Layer: Transport layer protocols: Introduction-services- port number-User data gram protocol-User datagram-UDP services-UDP applications-Transmission control protocol: TCP services- TCP features- Segment- A TCP connection- windows in TCP- flow control-Error control. **Application Layer –-** World Wide Web: HTTP , FTP-Two connections-control connection-Data connection-security of FTP-Electronic mail-Architecture- web based mail- email security- TELENET-local versus remote Logging. **Domain Name System:** Name Space, DNS in Internet, - Resolution-Caching- Resource Records- DNS messages- Registrars-security of DNS Name Servers, DNS Security. **COs-C05**

Text Books:

1) Computer Networks: Andrew S Tanenbaum David J. Wetherall, 5/e, Pearson
2) Data communications and networking: Behrouz Forouzan, 5/e, McGraw Hill

Reference Books:

1) Computer Networks – A System Approach, Peterson, Bruce Davie,2/e , Harcourt Asia
2) Compute communications and networking technologies, Gallo, Hancock, Cengage
3) An Engineering approach to compute networking, Kesha, Pearson

Web References:

1) https://www.geeksforgeeks.org/computer-network-tutorials/
2) https://www.tutorialspoint.com/computer_fundamentals/computer_networking.htm

Software Engineering and Design patterns
I M.C.A- II SEMESTER

Course Title: Software Engineering and Design Patterns	Course Code: **24F00PC09**
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

Course Overview: This overview provides a foundational understanding of what a Software Engineering and Design Patterns course typically includes, preparing students for real-world software development challenges.

Course Objectives:

1. To understand the nature of software development and software life cycle models
2. To understand methods of capturing, specifying, visualizing and analyzing software requirements.
3. Understand the concept of Design patterns and its importance.
4. Understand the behavioral knowledge of the problem and solutions.
5. Relate the Creational, Structural, behavioral Design patterns.
6. Apply the suitable design patterns to refine the basic design forgiven context.

At the end of the course, student will be able to

CO#	Course Outcomes
CO1	Define various software application domains and remember different process model used in software development.
CO2	Explain needs for software specifications also they can classify different types of software requirements and their gathering techniques.
CO3	Convert there requirements model into the design model and demonstrate use of software and user interface design principles.
CO4	Illustrate the appropriate design patterns to solve object-oriented design problems.
CO5	Evaluate the design solutions by using behavioral patterns and Apply structural patterns to solve design problems.

COURSE CONTENT (SYLLABUS)

UNIT-I: Introduction to Software Engineering **14 Hours**

The evolving role of software, Changing Nature of Software, Software myths. **The software problem:** Cost, schedule and quality, Scale and change. **Software Process:** Process and project, component software process, SDLC, **Software development process models:** Waterfall model, prototyping, iterative development, relational unified process, time boxing model, Extreme programming and agile process, Sprial Model **COs-CO1**

UNIT-II: **12 Hours**

Software requirement analysis and specification: Value of good SRS, requirement process, requirement specification, functional specifications with use-cases, **Planning a software project:** Effort estimation, project schedule and staffing, quality planning, risk management planning, project monitoring plan, detailed scheduling. **COs-CO2**

UNIT-III: Software Architecture **16 Hours**

Role of software architecture, architecture views, components and connector view, documenting architecture design, evaluating architectures, **Design:** Design concepts, function-oriented design, object oriented design, detailed design,

Software Testing: Introduction, verification and validation, White box and Black box techniques **COs-CO3**

UNIT-IV: Introduction: **12 Hours**

History and Origin of Patterns, Design Patterns in MVC, Describing Design Patterns, How Design Patterns Solve Design Problems, selecting a Design Pattern, Using a Design Pattern

DesignPatterns-1:Creational , Abstract Factory-Builder, Factory – Method, Prototype-singleton.

UNIT-V:

Design Patterns-2:Structural Patterns: Adapter, Bridge, Composite, Decorator, Façade, Flyweight, Proxy

Design Patterns-3: Behavioural Patterns, Chain of Responsibility, Command-Interpreter, Iterator-Mediator, Memento, Observer, State, Strategy, Template Method, Visitor **COs-CO4**

Text Books:

1. Software Engineering: A Practitioner's Approach, Roger S. Pressman, 10thed,Mc Graw Hill.
2. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, Design patterns: Elements of Reusable object-oriented software, Addison-Wesley, 1995.
3. James W Cooper, Java Design Patterns-A Tutorial, Addison-Wesley

Reference Books:

1. Software Engineering,8/e, Sommerville, Pearson
2. Software Engineering principles and practice, WS Jawadekar, TMH
3. Craig Larman, Applying UML and Patterns: An Introduction to object- Oriented Analysis and Design and iterative development,3rd Edition, Pearson,2005.
4. Thomas J Mowbray and Raphael Malveau, CORBA and Design Patterns, John Wiley, 1997.
5. William J Brown, Anti-Patterns: Refactoring Software, Architectures and Projects in Crisis, John Wiley, 1998.

Web References:

1. https://www.computer.org/education/bodies-of-knowledge/software-engineering
2. https://www.geeksforgeeks.org/software-engineering/?ref=lbpMartin Fowler’s Blog

Data Warehousing and Data Mining

I M.C.A- II SEMESTER

Course Title: **Data Warehousing and Data Mining**	Course Code: **24F00PC10**
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

Course Overview: Data warehousing and data mining are complementary disciplines that play crucial roles in modern data management and analytics.

Together, they empower organizations to harness their data effectively, uncover valuable insights, and drive better business outcomes.

COURSE OBJECTIVES:

1. Be familiar with mathematical foundations of data mining tools..
2. Understand and implement classical models and algorithms in data warehouses and data mining
3. Characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering.
4. Develop skills in electing the appropriate data mining algorithm for solving practical problems.

At the end of the course, the students will be able to:

CO#	Course Outcomes
CO1	Understand the basics of types of data, quality of data, suitable techniques required for preprocessing and measures required to perform data analysis
CO2	Describe the need of classification, identify suitable technique(s) to perform classification , model building and evaluation
CO3	Identify the requirements and usage of association rule mining on categorical and continuous data.
CO4	Evaluate clustering results and handle high-dimensional data challenges.
CO5	Describe the requirements and the need of web mining

COURSE CONTENT (SYLLABUS)

UNIT-I: **15 Hours**

Introduction to Data mining and Data preparation

Introduction to Data mining, Definition of Data and Data mining, types of Data, Data Quality, Data Processing, Measures of Similarity and Dissimilarity, Exploring Data: Data Set, Data Set Characteristics, Summary Statistics, Visualization, Data Warehouse, OLAP and multi-dimensional data analysis. **COs-C01**

UNIT-II: **15 Hours**

Classification: Basic Concepts, Decision Trees and model evaluation: General approach for solving a classification problem, Decision Tree induction, Regression, Type of Regression **Model over fitting:** due to presence of noise, due to lack of representation samples, Evaluating the performance of classifier. Nearest Neighborhood classifier, Bayesian Classifier, Support vector Machines: Linear SVM, Separable and Non Separable case. **COs-C02**

UNIT-III: **15 Hours**

Association Analysis: Problem Definition, Frequent Item-set generation, rule generation, compact representation of frequent item sets, FP-Growth Algorithms. Handling Categorical, Continuous attributes, Concept hierarchy, Sequential, Sub graph patterns. **COs-C03**

UNIT-IV: **15 Hours**

Clustering Techniques: Clustering Overview, Types of clustering, K-means, Agglomerative Hierarchical clustering, DBSCAN, **Cluster evaluation**: overview, Unsupervised Cluster Evaluation using cohesion and separation, using proximity matrix, Scalable Clustering algorithm. **COs-C04**

UNIT-V: **18 Hours**

Web data mining: Introduction, Web terminology and characteristics, Web content mining, Web usage mining, web structure mining, **Search Engines**: Characteristics, Functionality, Architecture, Ranking of Web pages, Enterprise search. **COs-C05**

Board of Studies : Master of Computer Applications
Approved in BOS No: 21, August, 2024
Approved in ACM No: 01

Text Books:

1) Introduction to Data Mining, Tan, Steinbach and Vipin Kumar, PearsonEducation,2016
2) Data Mining: Concepts and Techniques, 2nd Edition, JiaweiHan and Micheline Kamber, ELSEVIER

Reference Books:

1) Data Mining: The Textbook, Springer, May2015, Charu C.Aggarwal

Web References:

1) NPTEL:https://nptel.ac.in/courses/106/105/106105174/
2) https://www.saedsayad.com/data_mining.htm

DESIGN AND ANALYSIS OF ALGORITHMS
I M.C.A- II SEMESTER

Course Title: **DESIGN AND ANALYSIS OF ALGORITHMS**	Course Code: **24F00PC11**
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

COURSE OVERVIEW: Design and Analysis of Algorithms is a fundamental aspect of computer science that involves creating efficient solutions to computational problems and evaluating their performance.
DSA focuses on designing algorithms that effectively address specific challenges and analyzing their efficiency in terms of **time** and **space complexity**

COURSE OBJECTIVES:

1. To provide an introduction to formalisms to understand, analyze and denote time complexities of algorithms
2. To introduce the different algorithmic approaches for problem solving through numerous example problems
3. To provide some theoretical grounding in terms of finding the lower bounds of algorithms and the NP-completeness

At the end of the course, the students will be able to:

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Describe asymptotic notation used for denoting performance of algorithms.
CO2	Analyze the performance of a given algorithm and denote its time complexity using the asymptotic notation for recursive and non-recursive algorithms.
CO3	List and describe various algorithmic approaches.
CO4	Solve problems using divide and conquer, greedy, dynamic programming, backtracking and branch and bound algorithmic approaches.
CO5	Apply graph search algorithm store al world problems.

COURSE CONTENT (SYLLABUS)

UNIT-I: **10 Hours**

Introduction: Algorithm, Pseudo code for expressing algorithms, performance Analysis-Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, probabilistic analysis, Amortized analysis. Disjoint Sets-disjoint set operations, union and find algorithms, spanning trees, connected components and bi- connected components.
COs-C01

UNIT-II: **15 Hours**

Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort, Stassen's matrix multiplication, Convex shell. Greedy method: General method, applications-Job sequencing with deadlines, 0/1 knapsack problem, Minimum cost spanning trees, Single source shortest path problem.
COs-C02

UNIT-III: **15 Hours**

Dynamic Programming: General method, applications-Matrix chain multiplication, Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.. COs-C03

UNIT-IV: **10 Hours**

Backtracking: General method, applications-n-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles. COs-C04

UNIT V: **10 Hours**

Branch and Bound: General method, applications - Travelling sales person problem, 0/1 knapsack problem- LC Branch and Bound solution, FIFO Branch and Bound solution. NP -Hardand NP-Complete problems: Basic concepts, non deterministic algorithms, NP - Hard and NP Complete classes, Cook's theorem. COs-C05

Text Books:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahniand Rajase kharam, Universities Press
2. The Algorithm Design Manual, 2ndedition, Steven S.Skiena, Springer
3. Introduction to Algorithms ,second edition, T.H. Cormen, C.E .Leiserson, R.L. Rivestand C.Stein, PHI Pvt. Ltd

Reference Books:

1) Introduction to the Design and Analysis of Algorithms, Anany Levitin, PEA
2) Design and Analysis of Algorithms, Pearson Education, Parag Himanshu Dave, Himansu Balachandra Dave
3) Introduction to Design and Analysis of Algorithms A strategic approach, R.C.T .Lee, S.S.Tseng, R.C.Chang and T.Tsai, Mc Graw Hill.
4) Design and Analysis of algorithms, Pearson education, Aho, Ullman and Hopcroft

Web References:

1) https://www.Design and Analysis of Algorithms - GeeksforGeeks
2) https://www. DAA Tutorial | Design and Analysis of Algorithms Tutorial - javatpoint

Database Management System Lab

I M.C.A- II SEMESTER

Course Title: **Database Management System Lab**	Course Code: **24F00PC12**
Teaching Scheme (L:T:P): 0:0:3	Credits: 1.5
Type of Course: Practicals	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

Course overview:

A database is an organized collection of data stored in a computer system and usually controlled by a database management system (DBMS).

The data in common databases is modeled in tables, making querying and processing efficient. Structured query language (SQL) is commonly used for data querying and writing.

COURSE OBJECTIVES:

1. This Course will enable students to
2. Populate and query a database using SQL DDL/DML Commands
3. Declare and enforce integrity constraints on a database
4. Writing Queries using advanced concepts of SQL
5. Programming PL/SQL including procedures, functions, cursors and triggers.

CO#	Course Outcomes
CO1	Utilize SQL to execute queries for creating database and performing data manipulation operations
CO2	Examine integrity constraints to build efficient databases
CO3	Apply Queries using Advanced Concepts of SQL

List of Experiments:

1. Execute all DDL, DML and DCL commands on sample tables.
2. Implementation of different types of operators and built-in functions with suitable examples
3. Implementation of different types of joins with suitable examples
4. Create views, partitions, Sequence, Indexes and locks for a particular DB
5. Implement different types of constraints on relations.
6. Implementation of sub queries and nested queries.
7. Implement Queries on Group By & Having Clauses, ALIAS, Sequence By, Order By
8. Control Structure
    a. Write a PL/SQL block for Addition of Two Numbers
    b. Write a PL/SQL block for IF, IF and else condition
    c. Write a PL/SQL block for implementation of loops
    d. Write a PL/SQL block for greatest of three numbers using IF and ELSEIF
9. Exception Handling- Implement the following with respect to exception handling. Raising Exceptions, User Defined Exceptions, Pre-Defined Exceptions.

10. Write PL/SQL block for an application using exception handling Procedures
    a. Write a PL/SQL Procedure using Positional Parameters
    b. Write a PL/SQL Procedure using notational parameters
    c. Write a PL/SQL Procedures for cursor implementation (explicit and implicit cursors)
11. **Functions:**
    a. Write a PL/SQL block to implement factorial using functions
    b. Develop a PL/SQL function get_bonus that takes an employee ID and returns a bonus amount based on the employee's salary: 10% for salaries above $50,000, otherwise 5%. Test the function with various salaries.
12. Write a DBMS program to prepare Pl/SQL reports for an application using functions.
13. **Triggers:**
    a. Write a Trigger to pop-up the DML operations
    b. Write a Trigger to check the age valid or not Using Message Alert.
    c. Create a Trigger to Raise appropriate error code and error message.
    d. Create a Trigger on a table so that it will update another table while inserting values
14. Write PL/SQL block for an application using cursors and all types of triggers.
15. Write a PL/SQL block for transaction operations of a typical application using package

Text Books:

1) Oracle: The Complete Reference by oracle Press.
2) Nilesh Shah, " Database Systems using Oracle ", PHI, 2007
3) Rick F Vander Lans, "Introduction to SQL ", Fourth Edition , Pearson Education, 2007

Computer Networks Lab
I M.C.A- II SEMESTER

Course Title: **Computer Networks Lab**	Course Code: **24F00PC13**
Teaching Scheme (L:T:P): 0:0:3	Credits: 1.5
Type of Course: Practicals	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

Course overview:

In computer networking, instead of phones, we have computers and instead of phone lines, we use cables, Wi-Fi, or other methods to connect them.

When computers are connected to a network, they can share information and resources, like files, printers, and internet connections.

This allows them to communicate with each other quickly and easily, just like friends talking on their phones.

Course Objectives:

These objectives aim to equip students with practical skills in implementing and understanding network protocols, IPC mechanisms, and client-server communications.

CO#	Course Outcomes
CO1	Demonstrate the ability to implement and analyze various network protocols.
CO2	Apply and evaluate key algorithms such as Dijkstra's for shortest path calculation and distance vector routing algorithms
CO3	Implement and utilize different IPC mechanisms, including pipes, FIFOs, message queues, and shared memory, to facilitate effective communication and synchronization between processes in a multi-tasking environment.
CO4	Design, develop, and test TCP and UDP client-server applications
CO5	Apply error detection techniques

LIST OF EXPERIMENTS

PART – A

1. Implement the data link layer farming methods such as character stuffing and bit stuffing.
2. Implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.
3. Implement Dijkstra's algorithm to compute the Shortest path through a graph.
4. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table art each node using distance vector routing algorithm
5. Take an example subnet of hosts. Obtain broadcast tree for it.

PART – B

1. Implement the following forms of IPC.
   a) Pipes b) FIFO
2. Implement file transfer using Message Queue form of IPC
3. Write a programme to create an integer variable using shared memory concept and increment the variable
4. Simultaneously by two processes. Use semaphores to avoid race conditions
5. Design TCP iterative Client and server application to reverse the given input sentence
6. Design TCP client and server application to transfer file
7. Design a TCP concurrent server to convert a given text into upper case using multiplexing system call “select”
8. Design a TCP concurrent server to echo given set of sentences using poll functions
9. Design UDP Client and server application to reverse the given input sentence
10. Design UDP Client server to transfer a file
11. Design using poll client server application to multiplex TCP and UDP requests for converting a given text into upper case.

Software Engineering and Design Patterns Lab

I M.C.A- II SEMESTER

Course Title: **Software Engineering and Design Patterns Lab**	Course Code: **24F00PC14**
Teaching Scheme (L:T:P): 0:0:3	Credits: 1.5
Type of Course: Practicals	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

Course Overview: This overview provides a foundational understanding of what a Software Engineering and Design Patterns course typically includes, preparing students for real-world software development challenges.

COURSE OBJECTIVES:

1. To develop problem-solving skills by defining real-world problems and creating detailed documentation including flowcharts, SRS sheets, and risk management plans.
2. To apply effort estimation techniques such as COCOMO and Function Points (FP) to assess project effort and manage project timelines using PERT or CPM.
3. To design and model software systems using various software engineering tools and techniques, including E-R diagrams, DFDs, CF diagrams, and structured charts.
4. To design effective test cases and perform thorough testing based on requirements and design specifications.
5. To understand and apply version control and change control practices for managing software configuration items.

By the end of this lab course, students should be able to:

CO#	Course Outcomes
CO1	Clearly articulate problem statements for real-world systems and develop flowcharts to illustrate problem-solving approaches and Create a comprehensive Software Requirements Specification (SRS) document for the chosen system.
CO2	Use the COCOMO model to estimate the effort required for a software project and Perform Function Point (FP) estimation to assess the project's complexity and required effort and Develop a project timeline and scheduling plan using PERT or CPM methods
CO3	Identify and analyze potential risks associated with a project and prepare a Risk Management, Monitoring, and Mitigation (RMMM) plan and Draw and interpret E-R diagrams, Data Flow Diagrams (DFD), Control Flow Diagrams (CFD), and structured charts to model the system's architecture and data flow.
CO4	Create test cases based on system requirements and design to ensure the software meets the specified criteria and Prepare version control and change control documentation to effectively manage software configuration items throughout the development lifecycle.
CO5	Design and implement the Abstract Factory, Builder, Façade, Bridge, and Decorator design patterns using UML diagrams and Java code and Demonstrate the practical application of these patterns to solve design problems and Design and implement the Chain of Responsibility pattern to manage request handling in a system, such as processing print commands in a word processing application.

List of Experiments

1. Write down the problem statement for a suggested system of relevance. Develop Flow-Charts to understand basic problem solving technique
2. Do requirement analysis and develop Software Requirement Specification Sheet (SRS) for suggested system.
3. Using COCOMO model estimate effort.
4. Perform Estimation of effort using FP Estimation for chosen system
5. Analyze the Risk related to the project and prepare RMMM plan.
6. Develop Time-line chart and project table using PERT or CPM project scheduling methods.
7. Draw E-R diagrams, DFD, CFD and structured charts for the project.
8. Design of Test cases based on requirements and design.
9. Prepare FTR
10. Prepare Version control and change control for software configuration items.
11. Using UML/JAVA, design Abstract Factory design pattern
12. Using UML/JAVA, design Builder design pattern
13. Using UML/JAVA, design Façade design pattern
14. Using UML/JAVA, design Bridge design pattern
15. Using UML/JAVA, design Decorator design pattern
16. User gives a print command from a word document. Design to represent this chain of responsibility design pattern

Exercise Problems

1. Using UML/JAVA, design Factory Method design pattern.
2. Using UML/JAVA, design Prototype
3. Using UML/JAVA, design Singleton
4. Using UML/JAVA , design Adapter design pattern
5. Using UML/JAVA , design Composite design pattern
6. Using UML/JAVA, design Flyweight, Proxy design pattern

EMPLOYABILITY SKILLS
I M.C.A- II SEMESTER

Course Title: **EMPLOYABILITY SKILLS**	Course Code: **R24MCA2109**
Teaching Scheme (L:T:P): 0:0:1	Credits: 0.5
Type of Course: Practicals	
Continuous Internal Evaluation: 0 Marks	Semester End Exam: 25 Marks
Pre requisites:	

Course Overview: Employability skills refer to the latter—they are workplace or transferable skills that help make you an impactful employee.

You can also think of them as skills that employers tend to value.

Many employability skills are "higher cognitive skills" that can't be carried out in automated tasks because they require creativity, critical thinking, and other important social and emotional abilities.

COURSE OBJECTIVES:

The main of this course is

- To learn how to make effective presentations and impressive interviews
- To learn skills for discussing and resolving problems on the work site
- To assess and improve personal grooming.
- To promote safety awareness including rules and procedures on the work site
- To develop and practice self management skills for the work site

By the end of this lab course, students should be able to:

CO#	Course Outcomes
CO1	By the end of this course, the student
CO2	• Recite the soft skills
CO3	• Make presentations effectively with appropriate ody language
CO4	• Be composed with positive attitude
CO5	• Apply their core competencies to succeed in professional and personal life

List of Experiments:

A list of vital employability skills from the stand point of engineering students with discussion how to potentially develop such skills through campus life.

1) Soft Skills: An Introduction–Definition and Significance of Soft Skills; Process, Importance and Measurement of Soft Skill Development.
2) Self-Discovery: Discovering the Self; Setting Goals; Beliefs, Values, Attitude, Virtue.
3) Positivity and Motivation: Developing Positive Thinking and Attitude; Driving out Negativity; Meaning and Theories of Motivation; Enhancing Motivation Levels.
4) Time Management–Concept, Essentials, Tips.
5) Personality Development– Meaning, Nature, Features, Stages, Models; Learning Skills; Adaptability Skills.
6) Decision-Making and Problem-Solving Skills: Meaning, Types and Models, Group and Ethical Decision-Making, Problems and Dilemmas in application of these skills.
7) Conflict Management: Conflict- Definition, Nature, Types and Causes; Methods of Conflict Resoultion.
8) Stress Management: Stress - Definition, Nature, Types, Symptoms and Causes; Stress Analysis Models and Impact of Stress; Measurement and Management of Stress

9) Leadership and Assertiveness Skills: A Good Leader; Leaders and Managers; Leadership Theories; Types of Leaders; Leadership Behaviour ; Assertivness Skills.

Note: The student shall be instructed to Record a 2 min video and add to profile before and after taking the course. Students are to be involved in Role Play, Team dynamics, Group Discussion and out comes are to be recorded.

Reference Books:

1) B. arun, K.Mitra, Personality Development and SoftSkills,Oxford University Press,2011.
2) S.P.Dhanavel,English and SoftSkills,OrientBlackswan,2010.
3) R.S.Aggarwal, A Modern Approach to Verbal& Non-Verbal Reasoning, S.Chand& Company Ltd., 2018.
4) Raman, Meenakshi & Sharma ,Sangeeta, Technical Communication Principles and Practice, Oxford University Press, 2011.
5) Managing Soft Skills for Personality Development – edited by B.N.Ghosh, Mc Graw Hill India, 2012.
6) English and Soft Skills–S.P.Dhanavel, Orient Blackswan India,2 010.