

Course Structure and Syllabus

MCA

Master of Computer Applications
(2 & 3 Year Program)



NARASARAOPETA ENGINEERING COLLEGE

(Autonomous)

Kotappakonda Road, Yellamanda (P.O), Narasaraopet- 522601, Guntur District, AP.
Approved by AICTE, New Delhi & Permanently affiliated to JNTUK, Kakinada. Code: 47.
Accredited by NBA & NAAC with "A" Grade
Phone: 08647239917 Website: www.nrtec.ac.in

ACADEMIC REGULATIONS - 2019 FOR MCA (REGULAR/LATERAL)
(Effective for the students admitted into I & II year from the Academic Year 2019-20 and onwards)**1. ELIGIBILITY FOR ADMISSION**

- (a) Admission to the above program shall be made subject to eligibility, qualification and Specialization as prescribed by the University from time to time. Admissions shall be made on the basis of merit / rank obtained by the candidates at the qualifying entrance test ICET or on the basis of any other order of merit as approved by the Government from time to time.
- (b) Admission eligibility-Under Lateral Entry, A Student with a recognised Bachelor's degree of minimum 3 years duration in BCA, B.Sc. (Information Technology/Computer Science) with mathematics as a course at 10+2 level or at Graduate level and obtained at least 50 % (45 % in the case of candidates belonging to reserved category) in the qualifying degree examinations can have an option of direct admission into 2nd year MCA (Lateral Entry scheme). Under this scheme, 20% seats of sanctioned intake will be available in each course as Supernumerary seats. Admissions to this two year MCA later entry Programme will be through ICET. The maximum period to complete MCA under Lateral Entry scheme is four consecutive academic years from the date of joining.

2. AWARD OF MCA DEGREE

A student will be declared eligible for the award of the MCA Degree, if he fulfills the following academic requirements.

- (a) Pursue a course of study for not less than three academic years and not more than six academic years.
For Lateral entry- Pursue a course of study for not less than two academic years and not more than four academic years.
- (b) The candidate registers for 142 credits and secure all 142 credits.
For Lateral entry-The candidate registers for 90 credits and secure all 90 credits.

3. STRUCTURE OF THE PROGRAM

Semester	Credits
I MCA I SEM	26
I MCA II SEM	26
II MCA III SEM	26
II MCA IV SEM	26
III MCA V SEM	26
III MCA VI SEM	12
TOTAL	142

Each course is normally assigned a certain number of credits as follows:

- 4 credits for 4 lecture periods per week
- 4 credits for 3 lecture periods and 1 tutorial per week

- 2 credits for 4 laboratory periods per week.
- 2 credits for seminar
- 10 credits for project work.

4. DISTRIBUTION AND WEIGHTAGE OF MARKS

The performance of the candidate in each semester shall be evaluated subject wise, with a maximum of 100 marks for theory / practical on the basis of internal evaluation and end semester examinations. Seminar in the 6th semester will be evaluated for 100 marks based on the internal evaluation. For project work grades(O/E/A/B/C/P/F) will be given.

4.1 THEORY

For all theory courses consisting 6 units in each subject, the assessment shall be for 40 marks through internal evaluation and 60 marks through external end semester examination of 3 hours duration.

4.1.a. INTERNAL EVALUATION

The internal evaluation will be based on two cycle tests conducted in each semester. The 40 internal marks will be awarded as 75% of the best cycle and 25% of the least cycle examinations, where each cycle of examination contain

- Descriptive test - 30 Marks
- Assignment - 10 Marks

Each descriptive test question paper contains 3 questions one from each unit covering syllabus from 3 units (first 3 units for first cycle and remaining 3 units for second cycle). The student has to answer all the 3 questions (3X10M=30M). The descriptive examination will be conducted for 1½ hour duration.

In Assignment tests 5 or 6 questions from each unit will be declared in the class room atleast one week in advance. In the test, one question will be given at random to each student and student has to answer it.

The Assignment Test – 1 will be conducted for 10 marks covering the syllabus from 1st & 2nd units. The Assignment Test - 2 will be conducted for 10 marks from 4th & 5th units.

4.1. b. EXTERNAL EVALUATION

The question paper comprises of two parts i.e. Part-A and Part-B. Part-A is compulsory and consists 6 questions covering all the units with equal marks. Each question in Part-A carries 2 marks (6X2M=12M). Part-B consists of 6 questions, one from each unit and the student has to answer 4 questions out of six (4x12M= 48M). The duration of end theory examination is 3 hours.

4.2 PRACTICALS

For practical subjects evaluation is as follows during the semester

4.2. a. INTERNAL EVALUATION

There shall be continuous evaluation during the semester for 40 internal marks. The internal marks shall be awarded as follows:

Record	- 10 Marks
Day-to-day work	- 15 Marks
Internal Lab Test	- 15 Marks

4.2. b. EXTERNAL EVALUATION

For practical subjects there shall be an external examination at the end of the semester for 60 marks in the presence of external examiner.

4.3 SEMINAR

For Seminar, a student under the supervision of a faculty member shall collect an advanced topic and review the literature then submit to the department in a report form during the 6th semester and shall make an oral presentation before the departmental review committee consisting of the supervisor and head of the Department or a senior faculty member. Seminar will be evaluated for 100 marks based on technical content and presentation. There will be no external evaluation.

A candidate shall be deemed to have secured the minimum academic requirement in the seminar, if the student secures a minimum of 50% of marks in the examination.

If a candidate fails to secure the minimum marks prescribed for successful completion, he has to re-register by paying the prescribed fee. He has to submit a fresh report during that semester and appear for evaluation by the committee.

4.4 MOOCs

In order to inculcate continuous learning habit in students, one mandatory non credit self-learning course(Massive open online course) has been introduced.

Every Student has to Register for MOOCs course offered by any reputed organization/ Agencies approved by departmental Committee.

The Student has to submit certificate issued by the organization after successful completion of the online course to the department in the 5th Semester.

Degree will not be awarded, if the student fails to submit the certificate, however the student can start pursuing the online course in 3rd or 4th Semester.

4.5 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 a minimum of 50% of the total marks in the end semester examination and internal evaluation taken together.

- 4.6** In case the candidate does not secure the minimum academic requirement in any subject (as specified in 4.5) he/she has to re-appear for the end semester examination in that subject. In the same Semester of the subsequent year, a candidate shall be given one chance to re-register for each subject provided the internal marks secured by a candidate or 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-register 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 in the previous attempt stand cancelled. For re-registration the candidates have to apply to the college by paying the requisite fee and get approval from the authorities before beginning of the semester in which re-registration is required.
- 4.7** 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 examination in that subject. He/She shall again re-register the subject when next offered.
- 4.8** Laboratory examinations for MCA courses must be conducted with two examiners, one of them being the laboratory class teacher and the second examiner shall be appointed by the authorities from the panel of examiners submitted by the respective department.
- 4.9** There shall be an audit pass (Mandatory learning Course) course with no credits. There shall be no external examination. However, attendance in the audit course shall be considered while calculating aggregate attendance . Based on the day to day evaluation of each student, a Satisfactory/Not Satisfactory grade will be awarded. If any student gets Not Satisfactory grade, he/she has to reappear for the same course in the subsequent semester. Degree will be awarded only if student gets satisfactory grade.
- 5. PROJECT WORK**
Every candidate shall be required to submit a thesis or dissertation on a topic approved by the Project Review Committee.
- 5.1** A Project Review Committee (PRC) shall be constituted with Principal, Head of the department and two other senior faculty members of the department.
- 5.2** 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 V semester.
- 5.3** Every candidate has to submit, in consultation with his project supervisor, the title, objective and plan of action of his project work for approval. The student can initiate the Project work, only after obtaining the approval from the Project Review Committee (PRC).

- 5.4** In the VI Semester, student has to do a project for 21 weeks (one full semester). The student should prepare a Project Report under the supervision of a guide from the faculty.
- 5.5** 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) shall examine whether or not the change of topic/supervisor leads to a 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.
- 5.6** Three copies of the Project Thesis certified by the supervisor shall be submitted to the College.
- 5.7** The thesis shall be adjudicated by one examiner selected by the Authorities. For this, the HOD of the department shall submit a panel of 3 examiners, eminent in that field, with the help of the guide concerned.
- 5.8** The Viva examination of the project report shall be conducted by a board consisting of the external examiner, head of the department and the supervisor. Based on the Evaluation of project work the candidate will be awarded Grade O/Grade E/Grade A/Grade B/ Grade P/Grade F in Project Viva Voce. Head of the department shall coordinate and make arrangements for the conduct of Viva Voce Examination.
- 5.9** If the candidate gets Grade F (FAIL) in project viva voce, the candidate will retake the viva voce examination after 3 Months. If he fails to get one of the above grades other than grade F, he/she will not be eligible for the award of degree. If the candidate fails to secure one of the above Grades other than Grade F again, the Project shall be summarily rejected.

6. ATTENDANCE REQUIREMENTS

- i. A student shall be eligible to appear for the end examinations if he/she acquires a minimum of 75% of attendance in aggregate of all the subjects.
- ii. Condonation of shortage of attendance in aggregate up to 10% (65% and above and below 75%) in each semester for genuine medical reasons shall be approved by a committee duly appointed by the college. A fee stipulated by the college shall be payable towards condonation of shortage of attendance. However the number of condonations is restricted to three for the entire course. For Lateral entry the number of condonations is restricted to two for the entire course.
- iii. A student who is short of attendance in a semester may seek re-admission into that semester when offered next time within 4 weeks from the date of commencement of class work.
- iv. If any candidate fulfills the attendance requirement in the present semester, he/she shall not be eligible for re-admission into the same semester.

7. COURSE PATTERN

- (i) The entire course of study is of three academic years and every year will have TWO Semesters.

For Lateral entry-The entire course of study is of two academic years and every year will have TWO Semesters.

- (ii) A student is eligible to appear for the end examination in a subject, but absent for it or has failed in the end examinations may appear for that subject in supplementary examinations, when conducted next.

- (iii) When a student is detained due to shortage of attendance, he may be re-admitted in to the same semester/year in which he has been detained.

8. METHOD FOR AWARDING OF GRADE POINTS FOR A SUBJECT

Theory/ Laboratory / Seminar/ Project (% of marks in a subject)	Corresponding Grade Points	Letter Grade
91 - 100	10	O (Outstanding)
81 - 90	9	A (Excellent)
71 - 80	8	B (Very Good)
61 - 70	7	C(Good)
50 - 60	6	D(Pass)
<50	0	F (Fail)*

9. Criteria for Award of Grades/Division

9.1. Calculation of Semester Grade Point Average (SGPA)* for Semester

The performance of each student at the end of each semester is indicated in terms of SGPA. The SGPA is calculated as given below:

$$SGPA = \frac{\sum(CR \times GP)}{\sum CR}$$

Where CR= Credits of a course

GP = Grade Points awarded for a course

*SGPA is calculated for a candidate who passed all the courses in that semester.

9.2. Calculation of Cumulative Grade Point Average (CGPA) for Entire Program

The CGPA is calculated as given below:

$$CGPA = \frac{\sum(CR \times GP)}{\sum CR}$$

Where CR= Credits of a course

GP = Grade Points awarded for a course

*CGPA is calculated for a student who passed all the subjects in previous semesters along with current semester.

*The SGPA and CGPA shall be rounded off to 2 decimal points and reported in the transcripts.

*Equivalent percentage = (CGPA – 0.75) x 10

9.3. Award of Division

After satisfying the requirements prescribed for the completion of the program, the student shall be eligible for the award of MCA Degree and shall be placed in one of the following classes:

CGPA	Class
≥7.75	First Class with Distinction (with no subject failures)
≥ 6.75	First Class (with subject failures)
≥ 5.75 & < 6.75	Second Class
< 5.75	Pass Class

10. REVALUATION/CHALLENGE EVALUATION

1. Student can submit the application for revaluation/challenge evaluation, along with the prescribed fee for Revaluation/challenge evaluation of his/her answer script(s) of theory course(s) as per the notification issued by the Controller of Examinations.
2. The Controller of Examinations shall arrange for revaluation/challenge evaluation of such answer script(s). An External examiner, other than the first examiner shall reevaluate the answer script(s).

11. MINIMUM INSTRUCTION DAYS

The minimum instruction days for each semester shall be 90 working days.

12. WITHHOLDING OF RESULTS

If the student has not paid the dues, if any, to the college or if any case of indiscipline is pending against him, the result of such student will be kept withheld. His degree will be withheld in such cases.

13. TRANSITORY REGULATIONS

Discontinued or detained candidates are eligible for readmission as and when next offered. A candidate, who is detained or discontinued in a semester, on readmission shall be required to do all the courses in the curriculum prescribed for the batch of students in which the student joins subsequently. However, exemption will be given to those candidates who have already passed such courses in the earlier semester(s) he was originally admitted into and substitute subjects are offered in place of them as decided by the Board of Studies. However, the decision of the Board of Studies will be final.

- 13.1 A student who is following JNTUK curriculum and detained due to shortage of attendance at the end of the first semester of first year shall join the autonomous batch of first year first semester. Such students shall study all the courses prescribed for the batch in which the student joins and considered on par with regular candidates of Autonomous stream and will be governed by the autonomous regulations.

13.2 A student who is following JNTUK curriculum, detained due to shortage of attendance at the end of the second semester of first year or at the subsequent semesters shall join with the autonomous batch in the appropriate semester. Such candidates shall be required to pass in all the courses in the program prescribed by the Board of Studies concerned for that batch of students from that semester onwards to be eligible for the award of degree. However, exemption will be given in the courses of the semester(s) of the batch which he had passed earlier and substitute subjects will be offered in place of them as decided by the Board of Studies. The student has to clear all his backlog subjects up to previous semester by appearing for the supplementary examinations conducted by JNTUK for the award of degree will be sum of the credits up to previous semester under JNTUK regulations and the credits prescribed for the semester in which a candidate seeks readmission and subsequent semesters under the autonomous stream. The class will be awarded based on the academic performance of a student in the autonomous pattern.

MALPRACTICES RULES

DISCIPLINARY ACTION FOR / IMPROPER CONDUCT IN EXAMINATIONS

- The Principal shall refer the cases of Malpractices in Internal Assessment Test and Semester end examinations to a malpractice prevention committee constituted by him for the purpose. Such committee shall follow the approved levels of punishment. The Principal shall take necessary action against the students based on the recommendations of the committee.
- Any action by the candidate trying to get undue advantage in the performance or trying to help another, or derive the same through unfair means is punishable according to the provisions contained hereunder:

	Nature of Malpractices/ Improper conduct	Punishment
	<i>If the candidate:</i>	
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).	Expulsion from the examination hall and cancellation of the performance in that subject only.
1(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.	Expulsion from the examination hall and cancellation of the performance in that subject only of all the candidates involved. In case 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	Expulsion from the examination hall

	<p>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.</p>	<p>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 college.</p>
3.	<p>Impersonates any other candidate in connection with the examination.</p>	<p>The candidate who has impersonated shall be expelled from examination hall. The candidate is also debarred and forfeits the seat. The performance of the original candidate who has been impersonated, shall be cancelled in all the subjects of the examination (including practicals 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 college examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat. If the imposter is an outsider, he will be handed over to the police and a case is registered against him.</p>
4.	<p>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.</p>	<p>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</p>

		semester/year. The candidate is also debarred for two consecutive semesters from class work and all college examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
5.	Uses objectionable, abusive or offensive language in the answer paper or in letters to the examiners or writes to the examiner requesting him to award pass marks.	Cancellation of the performance in that subject.
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, assaults 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 misconduct or 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 shall be expelled from examination halls and cancellation of their performance in that subject and all other subjects the candidate(s) has (have) already appeared and 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 college examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
8.	Possess any lethal weapon or firearm in 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 not 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 college 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 college for further action to award suitable punishment.	

OTHER MATTERS

- Physically challenged candidates who have availed additional examination time and a scribe during their UG / PG CET examinations will be given similar concessions on production of relevant proof / documents.
- The Principal shall deal in an appropriate manner with any academic problem which is not covered under these rules and regulations, in consultation with the Heads of the departments and subsequently such actions shall be placed before the Academic Council for ratification. Any emergency modification of regulation, approved in the meetings of the Heads of the departments shall be reported to the Academic Council for ratification.

GENERAL

1. The academic council may, from time to time, revise, amend or change the regulations, schemes of examinations and / or syllabi.
2. Where ever the words ‘he’ ‘him’ ‘his’, occur in the regulations, they include ‘she’, ‘her’, ‘hers’.
3. The academic regulation should be read as a whole for the purpose of any interpretation.
4. In the case of any doubt or ambiguity in the interpretation of the above rules, the decision of the principal is final.

**MASTER OF COMPUTER APPLICATION
R19 Curriculum Structure**

Semester	No. of Courses per Semester (Theory + Lab)	Credits
I	5+3	26
II	5+3	26
III	5+3+1 (Mandatory Course)	26
IV	5+3+1 (Mandatory Course)	26
V	5+3	26
VI	Seminar + Project Work (2 + 10)	12
TOTAL		142

MCA COURSE STRUCTURE (LATERAL ENTRY)

Semester	No. of Courses per Semester (Theory + Lab)	Credits
III	5+3+1 (Mandatory Course)	26
IV	5+3+1(Mandatory Course)	26
V	5+3	26
VI	Seminar + Project Work (2 + 10)	12
	TOTAL	90

I YEAR I SEMESTER										
S.No	Course Code	Courses	Cat. Code	Int	Ext	Tot	L	T	P	C
1	19MCA1TH01	C Programming & Data Structures	PC	40	60	100	4	-	-	4
2	19MCA1TH02	Computer Organization	PC	40	60	100	4	-	-	4
3	19MCA1TH03	Mathematical Foundations of Computer Science	PC	40	60	100	3	1	-	4
4	19MCA1TH04	Probability & Statistics	BS	40	60	100	3	1	-	4
5	19MCA1TH05	Accounts and Financial Management	MS	40	60	100	4	-	-	4
6	19MCA1LB01	C Programming & Data Structures Lab	PC	40	60	100	-	-	4	2
7	19MCA1LB02	Information Technology Workshop	PC	40	60	100	-	-	4	2
8	19MCA1LB03	English Proficiency Lab	HS	40	60	100	-	-	4	2
TOTAL				320	480	800	18	2	12	26

I YEAR II SEMESTER										
S.No	Course Code	Courses	Cat. Code	Int	Ext	Tot	L	T	P	C
1	19MCA2TH01	Database Management Systems	PC	40	60	100	4	-	-	4
2	19MCA2TH02	Object Oriented Programming	PC	40	60	100	4	-	-	4
3	19MCA2TH03	Operating Systems	PC	40	60	100	4	-	-	4
4	19MCA2TH04	Operations Research	BS	40	60	100	3	1	-	4
5	19MCA2TH05	Organizational Structure and Personnel Management	MS	40	60	100	4	-	-	4
6	19MCA2LB01	Database Management Systems Lab	PC	40	60	100	-	-	4	2
7	19MCA2LB02	Object Oriented Programming Lab	PC	40	60	100	-	-	4	2
8	19MCA2LB03	Python Programming Lab	PC	40	60	100	-	-	4	2
TOTAL				320	480	800	19	1	12	26

II YEAR III SEMESTER										
S.No	Course Code	Courses	Cat. Code	Int	Ext	Tot	L	T	P	C
1	19MCA3TH01	Software Engineering	PC	40	60	100	4	-	-	4
2	19MCA3TH02	Computer Networks	PC	40	60	100	4	-	-	4
3	19MCA3TH03	Design And Analysis of Algorithms	PC	40	60	100	3	1	-	4
4	19MCA3TH04	Web Development & Digital Marketing	PC	40	60	100	4	-	-	4
5	19MCA3TH05	Linux Programming	PC	40	60	100	4	-	-	4
6	19MCA3MC01	Community Services(CS)	MC	-	-	-	-	-	-	-
7	19MCA3LB01	Web Development & Digital Marketing Lab	PC	40	60	100	-	-	4	2
8	19MCA3LB02	Linux Programming Lab	PC	40	60	100	-	-	4	2
9	19MCA3LB03	Statistical Programming with R Lab	PC	40	60	100	-	-	4	2
TOTAL				320	480	800	19	1	12	26

II YEAR IV SEMESTER										
S.No	Course Code	Courses	Cat. Code	Int	Ext	Tot	L	T	P	C
1	19MCA4TH01	Object Oriented Analysis & Design	PC	40	60	100	4	-	-	4
2	19MCA4TH02	Network Security & Cryptography	PC	40	60	100	4	-	-	4
3	19MCA4TH03	Data Warehousing and Mining	PC	40	60	100	4	-	-	4
4	19MCA4MC01	MOOCS(Mandatory Course)	MC	-	-	-	-	-	-	-
5	Professional Elective – I		PE	40	60	100	4	-	-	4
	19MCA4PE04	Software Design Methodologies								
	19MCA4PE05	Mobile Computing								
	19MCA4PE06	Blockchain Technology								
6	Professional Elective – II		PE	40	60	100	4	-	-	4
	19MCA4PE07	Internet of Things								
	19MCA4PE08	Distributed Operating Systems								
	19MCA4PE09	Network Programming								
7	19MCA4LB01	Object Oriented Analysis & Design(UML) Lab	PC	40	60	100	-	-	4	2
8	19MCA4LB02	Data Warehousing and Mining Lab	PC	40	60	100	-	-	4	2
9	19MCA4LB03	Android App Development Lab	PC	40	60	100	-	-	4	2
TOTAL				320	480	800	20	-	12	26

III YEAR V SEMESTER										
S.No	Course Code	Courses	Cat. Code	Int	Ext	Tot	L	T	P	C
1	19MCA5TH01	Software Testing Methodology	PC	40	60	100	4	-	-	4
2	19MCA5TH02	Artificial Intelligence & Machine Learning	PC	40	60	100	4	-	-	4
3	19MCA5TH03	Cloud Computing	PC	40	60	100	4	-	-	4
4	Professional Elective – III		PE	40	60	100	4	-	-	4
	19MCA5PE04	Computer Forensics								
	19MCA5PE05	Cyber Security								
	19MCA5PE06	E-Commerce								
5	Professional Elective – IV		PE	40	60	100	4	-	-	4
	19MCA5PE07	Agile Methodology & Devops								
	19MCA5PE08	Embedded & Real Time Operating Systems								
	19MCA5PE09	Big Data Analytics								
6	19MCA5LB01	Software Testing Methodology Lab	PC	40	60	100	-	-	4	2
7	19MCA5LB02	Machine Learning using python Lab	PC	40	60	100	-	-	4	2
8	19MCA5LB03	Frameworks & Devops Lab	PC	40	60	100	-	-	4	2
TOTAL				320	480	800	20	-	12	26



NARASARAOPETA ENGINEERING COLLEGE (AUTONOMOUS)

III YEAR VI SEMESTER										
S.No	Course Code	Courses	Cat. Code	Int	Ext	Tot	L	T	P	C
1	19MCA6PS01	Project Seminar	PS	100	-	-	-	-	-	2
2	19MCA6PW01	Project Work (Grade O/A/B/C/D/F)	PW	-	-	-	-	-	-	10
TOTAL				100	-	-	-	-	-	12



I MCA I SEMESTER SYLLABUS

I YEAR I SEMESTER										
S.No	Course Code	Courses	Cat. Code	Int	Ext	Tot	L	T	P	C
1	19MCA1TH01	C Programming & Data Structures	PC	40	60	100	4	-	-	4
2	19MCA1TH02	Computer Organization	PC	40	60	100	4	-	-	4
3	19MCA1TH03	Mathematical Foundations of Computer Science	PC	40	60	100	3	1	-	4
4	19MCA1TH04	Probability & Statistics	BS	40	60	100	3	1	-	4
5	19MCA1TH05	Accounts and Financial Management	MS	40	60	100	4	-	-	4
6	19MCA1LB01	C Programming & Data Structures Lab	PC	40	60	100	-	-	4	2
7	19MCA1LB02	Information Technology Workshop	PC	40	60	100	-	-	4	2
8	19MCA1LB03	English Proficiency Lab	HS	40	60	100	-	-	4	2
TOTAL				320	480	800	18	2	12	26

I MCA I SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA1TH01	4	-	-	40	60	100	4
C PROGRAMMING AND DATA STRUCTURES							

COURSE OBJECTIVES:

1. Formulating algorithmic solutions to problems and implementing algorithms.
2. To understand the various steps in Program development.
3. To understand the basic concepts in Program.
4. Comprehensive knowledge of data structures and ability to implement the same in software applications.

COURSE OUTCOMES:

- CO 1:** Demonstrate the basic knowledge of computer hardware and software.
CO 2: Ability to problem solving and logical skills in programming.
CO 3: Implement modular programming using functions.
CO 4: Interpret dynamic memory allocation using pointers and Organize data into files.

UNIT I

Introduction to Computers, HW and SW concepts, Algorithm, pseudo code, flowchart, program development steps, Introduction to various IDE's and their use in C program development, structure of C program, A Simple C program, identifiers, basic data types and sizes, Constants, variables, arithmetic, relational and logical operators, increment and decrement operators, conditional operator, bit-wise operators, assignment operators, expressions, type conversions, conditional expressions, precedence and order of evaluation. Control structures such as if, goto, labels, and switch statements.

UNIT II

Loops- while, do-while and for statements, break, continue, Arrays – concepts, declaration, definition, accessing elements, storing elements, Strings and string manipulations, 1- D arrays other than strings, 2-D character arrays – 2-D arrays other than character arrays – Multidimensional arrays.

UNIT III

Functions: basics, parameter passing, storage classes- extern, auto, register, static, scope rules, block structure, user defined functions, standard library functions, recursive functions, header files, C pre-processor. Passing 1-D arrays, 2-D arrays , and functions.

Pointers: concepts, initialization of pointer variables, pointers and function arguments, passing by address –dangling memory, Character pointers and functions, pointers to pointers,

pointers and multidimensional arrays, dynamic memory management functions, command line arguments.

UNIT IV

Derived types: structures- declaration, definition and initialization of structures, accessing structures, nested structures, arrays of structures, structures and functions, pointers to structures, self-referential structures, unions, typedef, bit-fields, Input and output – concept of a file, text files and binary files, Formatted I/o, file I/o operations

UNIT V

Data Structures: Introduction to Data Structures – Time Complexity – Space Complexity – Pattern matching – naive method – Robin Karp Algorithm - Searching – Linear and binary search methods, sorting – Bubble sort, selection sort, Insertion sort, Quick sort, merge sort.

UNIT VI

Single linked lists, doubly linked lists, circular list, representing stacks and queues in C using arrays and linked lists, infix to post fix conversion, postfix expression evaluation. Trees- Binary trees, terminology, representation, traversals, Graphs - terminology, representation, graph traversals (dfs & bfs) – Warshalls – Dijkstra – Kruskal – Prims Algorithms.

TEXT BOOKS

1. Computer science, A structured programming approach using C, B.A.Forouzan and R.F. Gilberg, Third edition, Thomson.

REFERENCE BOOKS

1. Fundamentals of Data Structures in C , Horowitz, Sahni, Anderson-Freed, 2nd ed, Universities Press, 2008.
2. Classic Data Structures, Samanta, 2nd ed, PHI, 2009.
3. The C Programming Language, B.W. Kernighan, Dennis M.Ritchie, PHI/ Pearson.
4. C Programming with problem solving, J.A. Jones & K. Harrow, Dreamtech Press
5. DataStructures Using C , A.S.Tanenbaum, Y. Langsam, and M.J. Augenstein, Pearson.
6. Programming in C , Stephen G. Kochan, III Edition, Pearson .
7. Data Structures and Program Design in C, R.Kruse, Tondo, Leung, Shashi M, 2nd Edition, Pearson.
8. Data Structures and Algorithms, Aho, Hopcroft, Ullman, Pearson ,2006
9. C and Data Structures, Ashok N. Kamthane, Pearson.
10. C Programming and Data Structures, E Balaguruswamy, TMH, 2008.

I MCA I SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA1TH02	4	-	-	40	60	100	4
COMPUTER ORGANIZATION							

COURSE OBJECTIVE:

Comprehensive knowledge of computer system including the analysis and design of components of the system.

COURSE OUTCOMES:

After completion of this course student will be able to

- CO 1:** The basic components of a computer, including CPU, memories, and input/output, and their organization representation of data, addressing modes, instructions sets.
- CO 2:** Discriminate different register transfer micro operations, Principles of hardwired and micro programmed control.
- CO 3:** Demonstrate various fixed and floating point arithmetic operations, performing the Arithmetic operations of binary number systems and formulating the arithmetic functions and solve scientific problems by means of a numerical analysis method.
- CO 4:** Extrapolate memory organization and input, output organizations.
- CO 5:** Generalize pipe line and vector processing, multi processors and its applications.

UNIT I

BASIC STRUCTURE OF COMPUTERS: Computer Types, Functional Units, Basic Operational concepts, Bus Structures.

NUMBER SYSTEM AND COMPUTER ARITHMETIC:

Signed, Unsigned numbers, Addition and Subtraction, Multiplication, Division, Floating point representation, Logic operations, Gray Code, BCD Codes, Error Detecting Codes, Boolean Algebra, Simplification of Boolean Expressions, K-Maps.

UNIT II

COMBINATION AND SEQUENTIAL CIRCUITS:

Decoders, Encoders, Multiplexers, Half and Full Adders, Sequential Circuits, Flip flops, Registers.

UNIT III

BASIC COMPUTER ORGANIZATION AND DESIGN: Instruction codes, Computer registers, Computer instructions, Timing and control, Instruction cycle, Memory – Reference Instruction, Input – Output and Interrupt.

CENTRAL PROCESSING UNIT: General Register Organization, Stack organization, Instruction formats, Addressing modes, Data Transfer and manipulation, Program control.

UNIT IV

REGISTER TRANSFER LANGUAGE AND MICRO OPERATIONS: Register transfer language, Register transfer, Bus and Memory transfers, Arithmetic Micro operations, Logical Micro operations, Shift Micro operations, Arithmetic logic shift unit.

UNIT V

MICRO PROGRAMMED CONTROL: Control memory, Address sequencing, Micro program example, design of control unit.

UNIT VI

THE MEMORY SYSTEM: Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory.

INPUT-OUTPUT ORGANIZATION: Peripheral devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupts, Direct Memory Access.

TEXT BOOKS

1. Computer System Architecture, M.Moris Mano, 3rd Edition, Pearson/PHI
2. Digital Logic and Computer Organization, Rajaraman, Radhakrishnan, PHI, 2006
3. Digital Logic Design, Moriss Mano, PHI

REFERENCES

1. Computer Organization, 5 th ed., Hamacher, Vranesic and Zaky, TMH, 2002.
2. Computer System Organization & Architecture, John D. Carpinelli, Pearson, 2008.
3. Computer System Organization, Naresh Jotwani, TMH, 2009.
4. Computer Organization & Architecture: Designing for Performance, 7th edition.

I MCA I SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA1TH03	3	1	-	40	60	100	4
MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE							

COURSE OBJECTIVE:

This course will discuss fundamental concepts and tools in discrete mathematics with emphasis on their applications to computer science.

COURSE OUTCOMES:

- CO 1:** Student will be able to demonstrate skills in solving mathematical problems.
- CO 2:** Student will be able to comprehend mathematical principles and logic.
- CO 3:** Student will be able to demonstrate knowledge of mathematical modelling.
- CO 4:** Student will be able to formulate logic expressions for a variety of applications.

UNIT I

Mathematical Logic: Statements and notations, Connectives, Well-formed formulas, Truth Tables, tautology, equivalence implication, Normal forms, Theory of inference for the statement calculus, Rules of inference, Consistency of premises and indirect method of proof, Automatic Theorem Proving.

Predicate calculus: Predicates, statement functions, variables and quantifiers, predicate formulas, free & bound variables, universe of discourse, inference theory of predicate calculus

UNIT II

Set theory & Relations: Introduction, Relations and ordering, Properties of binary Relations, Equivalence, Compatibility Relations, Partial ordering, Hasse diagram.

Functions: composition of functions, Inverse Function, Recursive Functions, Lattice and its Properties, Pigeon hole Principles and its application.

UNIT III

Algebraic structures: Algebraic systems, Examples and general properties, Semi groups and monoids, groups, sub groups, Definitions, Examples, homomorphism, Isomorphism and related problems.

UNIT IV

Elementary Combinatorics: Basis of counting, Enumeration of Combinations & Permutations, Enumerating of Combinations & Permutations with repetitions and constrained

repetitions, Binomial Coefficients, Binomial, Multinomial theorems, principles of Inclusion – Exclusion.

UNIT V

Recurrence Relations: Generating Function of Sequences, Calculating Coefficient of generating functions, Recurrence relations, Solving recurrence relation by substitution and Generating functions, The method of Characteristic roots, Solution of Inhomogeneous Recurrence Relation.

UNIT VI

Graph Theory: Representation of Graph, Spanning Trees, BFS, DFS, Prim's, Kruskals Algorithm, Binary trees, Planar Graphs.

Graph Theory and Applications: Basic Concepts, Isomorphism and Sub graphs, Multi graphs and Euler circuits, Hamiltonian graphs, Chromatic Numbers.

TEXT BOOKS

1. Discrete Mathematical Structures with Applications to computer science J.P Tremblery, R. Manohar, TMH.
2. Discrete Mathematical structures for computer Scientists & Mathematicians “ L. Molt, A. Andel, T.P.Baker, PHI.

REFERENCE TEXT BOOKS

1. Discrete Mathematics, J K Sharma, 2nd ed., Macmillan, 2005.
2. Discrete Mathematics for computer science, Bogart, Stein and Drysdale, Springer, 2005.
3. Discrete Mathematics and its Applications, Rosen, 5th ed, T M Graw-Hill ed, 2006.
4. Discrete Mathematics for Computer Science, Gary Haggard, John Schlipf, Sue Whitesides, Cengage., 2006.
5. Discrete Mathematical Structures, Jayant Ganguly, Sanguine, 2007.
6. Discrete Mathematics and Combinatorics, Sengadir, Pearson, 2009.
7. Discrete Mathematical Structures, Kolman, Busby, Ross, 6th ed., PHI, 2009.
8. Discrete Mathematics with Combinatorics and Graph Theory, Santha, Cengage Learning, 2009.
9. Mathematical Foundations of Computer Science, Rajendra Prasad, Rama Rao et al., USP, 2009.
10. Discrete structures and Graph theory, GSS Bhisma Rao.
11. Mathematical Foundations of Computer Science, Dr.D.S. Chandrasekhar.

I MCA I SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA1TH04	3	1	-	40	60	100	4
PROBABILITY AND STATISTICAL APPLICATIONS							

COURSE OBJECTIVE:

To know the fundamental concepts in Probability which is regarding to sample points in a Sample space, Large Samples, Small Samples and a queuing model.

COURSE OUTCOMES:

At the end of the course student will be able to

CO 1: Examine, analyse and compare probability distributions.

CO 2: Construct confidence intervals for population parameters.

CO 3: Formulate null, alternative hypothesis and test its validity based on random sample.

CO 4: Apply queuing models to find the optimum service rate.

UNIT I

Probability Theory: Motivation, Probability models, Sample space, Events, Probability models; Probability Axioms, Union, Intersection and compliments of events; independent events, Conditional Probability; Baye's theorem.

UNIT II

Random variables: Random variables, Functions of Random variables; Probability mass function, Probability distributive function; cumulative distributive function; Discrete Probability Distributions, Binomial, Poisson, Continuous Probability distribution: Normal

UNIT III

Sampling Distributions: Populations and samples-Sampling distribution of mean (known and unknown), Proportions, Sums and differences. Point and Interval estimators for means and proportions, Bayesian Estimation.

UNIT IV

Test of significance(Large Samples): Introduction, Type I and Type II errors, one tail, two tail tests-Test concerning one mean and proportion, Two means, Proportions and their differences using Z-Test

UNIT V

Test of significance (Small Samples): Student t test, F-test and Chi-square test-ANOVA-for one way and two way classifications

UNIT VI

Curve Fitting: Method of least squares-straight line-parabola-exponential-power curves

TEXT BOOKS

1. Probability, Statistics and Random Process: Dr. K. Murugesan & P. Gurusamy by Anuradha Agencies, Deepthi Publications.
2. Probability, Statistics and Random Process: T. Veerarajan, TMH, India.

REFERENCE BOOKS

1. Probability & Statistics for Engineers: Miller and Freund, PHI
2. “Fundamentals of Mathematical Statistics”, 9th Edition, S.C. Gupta and V.K. Kapoor, Sultan Chand & Sons Educational Publishers, 2007.

I MCA I SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA1TH05	4	-	-	40	60	100	4
ACCOUNTS AND FINANCIAL MANAGEMENT							

COURSE OBJECTIVE:

To provide familiarity with the accounting concepts which will help in preparation of various accounting records.

COURSE OUTCOMES:

The student is able to

- CO 1:** Outline the basic knowledge of accounting Concepts.
- CO 2:** Apply the concepts and Accounting Principles.
- CO 3:** List the Basic Concepts of Financial Management and. Ratio Analysis.
- CO 4:** Assess the working capital requirement of a firm.
- CO 5:** Apply different marginal costing concepts under various situations.
- CO 6:** Measure the variances between the expected costs and actual costs.

UNIT I

Introduction to Accounting- Single Entry System- Objectives- Generally Accepted Accounting Principles- Systems of Accounting- Single Entry- Double Entry- Introduction to Basic Books of Accounts- Journal- Ledger.

UNIT II

Preparation of Trail Balance – Final Accounts with Simple Adjustments – Users of Accounting Information – Role of Accountant in Modern Business Organizations.

UNIT III

Introductions to Financial Management - Meaning – Scope – Role – Objectives – Time Value of Money – Over Capitalization – Under Capitalization – Profit Maximization – Wealth Maximization – EPS Maximization – Ratio Analysis – Different Types of Ratios – Advantages – Disadvantages.

UNIT IV

Costing – Nature – Importance – Basic Principles – Elements of Cost – Absorption Costing Vs Marginal Costing – Financial Accounting Vs Cost Accounting Vs Management Accounting – Marginal Costing and Break Even Analysis – Nature – Importance – Scope – CBP Analysis – Simple Problems.

UNIT V

Standard Costing and Budgeting – Nature – Scope – Computation Analysis – Material Variances – Labour Variances – Budgeting – Definition – Types of Budgets – Cash Budget – Flexible Budget – Sales Budget – Simple Problems.

UNIT VI

Introduction to Computerized Accounting System – Coding – Logic and Codes – Master Files – Transaction Files - Introduction to Documents Used for Data Collection – Process of Document Files and Outputs Obtained.

REFERENCE BOOKS

1. Accounting for Managers, G. Prasad, Himalaya Publications.
2. Accounting for Management, Vijay Kumar, Tata McGraw Hill.
3. Financial Accounting, S. N. Maheswari and S. K. Maheswari, Vikas Publications.
4. Cost and Management Accountancy, M.N. Arora, Himalaya Publishing House.
5. Financial Analysis and Accounting, P. Premchand Babu and M. Madan Mohan, Himalaya Publishing House.
6. Essentials of Financial Accounting, Ashish. K and Ballacharya, Prentice Hall of India.

I MCA I SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA1LB01	-	-	4	40	60	100	2
C PROGRAMMING AND DATA STRUCTURES LAB							

COURSE OBJECTIVE:

1. To learn/strengthen a programming language like C, To learn problem solving techniques
2. To introduce the student to simple linear and nonlinear data structures such as lists, stacks, queues etc.,

COURSE OUTCOMES:

- CO1:** Study, analyse and understand logical structure of computer programming and different constructs to develop programs in C Language.
- CO2:** Know how to write, compile and debug programs in C Language.
- CO3:** Understand and analyse data types, typecasting and operator precedence.
- CO4:** Analyse the use of conditional and looping statements to solve problems associated with conditions and repetitions.

Exercise 1

- a) Write a C program to find the sum of individual digits of a positive integer.
- b) A Fibonacci Sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.
- c) Write a C program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.
- d) Write a program which checks a given integer is Fibonacci number or not.

Exercise 2

- a) Write a C program to calculate the following Sum: $Sum = 1 - x^2/2! + x^4/4! - x^6/6! + x^8/8! - x^{10}/10!$
- b) Write a C program to find the roots of a quadratic equation.
- c) Write a C program to implement Newton Raphson method for a quadratic equation
- d) Write a C program to implement Newton Raphson method for a general purpose algebraic equation

Exercise 3

- a) Write C programs that use both recursive and non-recursive functions
 - i) To find the factorial of a given integer.
 - ii) To find the GCD (greatest common divisor) of two given integers.
 - iii) To solve Towers of Hanoi problem.
 - iv) Write program to calculate probability of head/tail by generating random numbers using random() function.

Exercise 4

- a) The total distance travelled by vehicle in 't' seconds is given by distance = $ut + 1/2at^2$ where 'u' and 'a' are the initial velocity (m/sec.) and acceleration (m/sec²). Write C program to find the distance travelled at regular intervals of time given the values of 'u' and 'a'. The

program should provide the flexibility to the user to select his own time intervals and repeat the calculations for different values of 'u' and 'a'.

- b) Write a C program, which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +, -, *, /, % and use Switch Statement)

Exercise 5

- a) Write a C program to find both the largest and smallest number in a list of integers.
- b) Write a C program that uses functions to perform the following:
 - i) Addition of Two Matrices
 - ii) Multiplication of Two Matrices
 - iii) Checking symmetricity of a square matrix.
 - iv) Calculating transpose of a matrix in-place manner.

Exercise 6

- a) Write a C program that uses functions to perform the following operations:
 - i) To insert a sub-string in to given main string from a given position.
 - ii) To delete n Characters from a given position in a given string.
- b) Write a C program to determine if the given string is a palindrome or not

Exercise 7

- a) Write a C program that displays the position/ index in the string S where the string T begins, or -1 if S doesn't contain T.
- b) Write a C program to count the lines, words and characters in a given text.

Exercise 8

- a) Write a C program to generate Pascal's triangle.
- b) Write a C program to construct a pyramid of numbers.

Exercise 9

Write a C program to read in two numbers, x and n, and then compute the sum of this geometric progression: $1+x+x^2+x^3+\dots+x^n$

For example: if n is 3 and x is 5, then the program computes $1+5+25+125$. Print x, n, the sum
Perform error checking. For example, the formula does not make sense for negative exponents – if n is less than 0. Have your program print an error message if $n < 0$, then go back and read in the next pair of numbers of without computing the sum. Are any values of x also illegal ? If so, test for them too.

Exercise 10

- a) 2's complement of a number is obtained by scanning it from right to left and complementing all the bits after the first appearance of a 1. Thus 2's complement of 11100 is 00100. Write a C program to find the 2's complement of a binary number.
- b) Write a C program to convert a Roman numeral to its decimal equivalent.

Exercise 11

Write a C program that uses functions to perform the following operations using Structure:

- i) Reading a complex number
- ii) Writing a complex number
- iii) Addition of two complex numbers
- iv) Multiplication of two complex numbers

Exercise 12

- a) Write a C program which copies one file to another.
- b) Write a C program to reverse the first n characters in a file.
(Note: The file name and n are specified on the command line.)

Exercise 13

- a) Write a C program that uses functions to perform the following operations on singly linked list.:

- i) Creation ii) Insertion iii) Deletion iv) Traversal
- b) Adding two large integers which are represented in linked list fashion.

Exercise 14

Write a C program that uses functions to perform the following operations on doubly linked list.

- i) Creation ii) Insertion iii) Deletion iv) Traversal in both ways

Exercise 15

- a.) Write C programs that implement stack (its operations) using
 - i) Arrays ii) Pointers iii) linked list.

Exercise 16

- a. Write C programs that implement Queue (its operations) using
 - i) Arrays ii) Pointers iii) linked lists.

Exercise 17

Write a C program that uses Stack operations to perform the following:

- i) Converting infix expression into postfix expression ii) Evaluating the postfix expression

Exercise 18

- a. Write a C program that uses functions to perform the following:
 - i) Creating a Binary Tree of integers
 - ii) Traversing the above binary tree in preorder, inorder and postorder.
- b. Program to check balance property of a tree.
- c. Program to check for its strictness.

Exercise 19

Write C programs that use both recursive and non-recursive functions to perform the following searching operations for a Key value in a given list of integers : i) Linear search ii) Binary search

Exercise 20

Write C programs that implement the following sorting methods to sort a given list of integers in ascending order: i) Bubble sort ii) Quick sort

Exercise 21

- a. Write C programs that implement the following sorting methods to sort a given list of Integers in ascending order: i) Insertion sort ii) Bubble sort
- b. Recursive implementation of sorting algorithms.

Exercise 22

Write C programs to implement the Lagrange interpolation and Newton- Gregory forward interpolation.

Exercise 23

- a. Program to calculate mean and standard deviation of a population.
- b. Write C programs to implement the linear regression and polynomial regression algorithms.

Exercise 24

- a. Write C programs to implement Trapezoidal and Simpson methods.
- b. Program for Calculating pi value.

REFERENCE BOOKS

1. Digital Fundamentals, Floyd, Jain, 8th ed , Pearson
2. Digital Logic and Computer Organization, Rajaraman, Radhakrishnan, PHI, 2006

I MCA I SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA1LB02	-	-	4	40	60	100	2
INFORMATION TECHNOLOGY WORKSHOP							

COURSE OBJECTIVE:

The objective of this course is to provide an insight into basic features of Computer Systems and their Applications in day to day professional work.

COURSE OUTCOME:

CO 1: The student will be able to perform documentation.

CO 2: The student will be able to create and use spread sheets with macros, charts, filters etc.

CO 3: The student will be able to create presentations with animations & auto slide transition

CO 4: The student will be able to configure Wi-Fi of laptop/tab/mobile to access internet.

Exercise - I

Introduction to Computer Concepts – Elements of computer – Characteristics of a Computer – Classification of Computers – Basic Computer Architecture – Input-output Devices, Types of software – Software: its nature and qualities — Windows and Linux Operating System Functions .

Exercise - II

Hardware, Software, Operating Systems, Printer, Scanner, Pen drive, DVD, Web cam, LAN File Sharing, Device Sharing, Blue Tooth, WI-FI, Internet, Web site, Domain, Search-Engine, Windows and Linux commands.

Exercise- III

Electronic mail, Outlook Configuration, Skype, Hangouts, Facebook, Twitter,(Social Networks), Cloud Storage(Google drive, One drive), Mobile, Tablets, Laptops, Mobile Apps.

Exercise - IV

MS Office- Applications of MS Word in Business Correspondence: letters, tables, mail merge, labels.

Exercise - V

Applications of MS Excel: Graphs and Charts – Calculation of various financial functions - Ms Access: Tables and Queries

Exercise - VI

MS Power Point: Introduction – Toolbar, their Icons and Commands – Navigating in Power point - Creation of slides, animation, and templates - Designing Presentations – Slide show controls – Making notes on Pages and Handouts – Printing Presentations – Customizing Presentations - Auto content Wizard.

REFERENCE BOOKS

1. Sanjay Saxena and Prathpreet Chopra, Computer Applications in Management, Vikas, New Delhi.
2. Aksoy, Introduction to Information Technology, Cengage, ND.
3. Parameswaran: Computer Application in Business – S Chand, New Delhi.
4. Management Information Systems by Mahadeo Jaiswal, Monika Mittal, Oxford University Press.
5. PS Gill, Database Management Systems, IK Int Pub House, New Delhi
6. Management Information Systems by D.P. Goyal, MacMillan Publishers.
7. The Compact guide to Microsoft office, Mansfield Rom, BPB Publications, Delhi.
8. Sudalaimuthu&Anthony Raj, Computer Applications in Business, Himalaya, Mumbai

I MCA I SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA1LB03	-	-	4	40	60	100	4
ENGLISH PROFICIENCY LAB							

COURSE OBJECTIVES:

1. To train the students to use language effectively in professional situations like group discussions, public speaking, presentations and interviews.
2. To make the students understand the importance of body language.
3. To develop positive attitude and soft skills to improve their employability quotient.
4. To expose the students to variety of a self-instructional, learner friendly, electronic media and stimulate intellectual faculties/resources

COURSE OUTCOMES:

Learners are able to

- CO 1:** Know the importance of Non-verbal communication and interpret nonverbal symbols
- CO 2:** Give presentations and attend job interviews confidently.
- CO 3:** Participate in group discussions applying critical thinking
- CO 4:** Face job interviews confidently for employability.

Unit-1: Body Language

Unit-2: Visume & Presentation skills

Unit-3: Group Discussion

Unit-4: Team Building and Presentation skills

Unit-5: Time Management and Goal setting

Unit-6: Interview Skills and Telephonic Interviews

TEXT BOOK:

Strengthen your Communication Skills by Maruthi Publications, 2013

REFERENCE BOOKS:

1. Personality Development and Soft Skills (Oxford University Press, New Delhi)
2. M Ashraf Rizvi, Effective Technical Communication skills, McGraw-Hill, 2005
3. Barun K Mitra, Personality Development and Soft Skills, Oxford University Press, 2011
4. Konar N, Communication Skills for Professionals, PHI Learning Private Limited, 2011



I MCA II SEMESTER SYLLABUS

I YEAR II SEMESTER										
S.No	Course Code	Courses	Cat. Code	Int	Ext	Tot	L	T	P	C
1	19MCA2TH01	Database Management Systems	PC	40	60	100	4	-	-	4
2	19MCA2TH02	Object Oriented Programming	PC	40	60	100	4	-	-	4
3	19MCA2TH03	Operating Systems	PC	40	60	100	4	-	-	4
4	19MCA2TH04	Operations Research	BS	40	60	100	3	1	-	4
5	19MCA2TH05	Organizational Structure and Personnel Management	MS	40	60	100	4	-	-	4
6	19MCA2LB01	Database Management Systems Lab	PC	40	60	100	-	-	4	2
7	19MCA2LB02	Object Oriented Programming Lab	PC	40	60	100	-	-	4	2
8	19MCA2LB03	Python Programming Lab	PC	40	60	100	-	-	4	2
TOTAL				320	480	800	19	1	12	26

I MCA II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA2TH01	4	-	-	40	60	100	4
DATABASE MANAGEMENT SYSTEMS							

COURSE OBJECTIVE:

The objective of the course is to enable students to understand and use a relational database system. Introduction to Databases, Conceptual design using ERD, Functional dependencies and Normalization, Relational Algebra are covered in detail. Students learn how to design and create a good database and use various SQL operations. The course concludes with an overview of transaction management and introduction to advanced and non-relational databases.

COURSE OUTCOMES:

- CO 1:** Able to master the basic concepts and understand the applications of database systems.
CO 2: Able to construct an Entity-Relationship (E-R) model from specifications and to transform to relational model.
CO 3: Able to construct unary/binary/set/aggregate queries in Relational Algebra.
CO 4: Understand and apply database normalization principles.

UNIT I

Database System Applications, Purpose of Database Systems, View of Data – Data Abstraction, Instances and Schemas, Data Models – the ER Model, Relational Model, Other Models – Database Languages – DDL, DML, Database Access from Applications Programs, Transaction Management, Data Storage and Querying, Database Architecture, Database Users and Administrators, History of Data base Systems. Introduction to Data base design, ER diagrams, Beyond ER Design, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of 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 data base Design, Introduction to Views – Destroying /altering Tables and Views.

UNIT II

Relational Algebra and Calculus: Relational Algebra – Selection and Projection, Set operations, Renaming, Joins, Division, Examples of Algebra Queries, Relational calculus – Tuple relational Calculus – Domain relational calculus – Expressive Power of Algebra and calculus.

Form of Basic SQL Query – Examples of Basic SQL Queries, Introduction to Nested Queries, Correlated Nested Queries, Set – Comparison Operators, Aggregate Operators, NULL values – Comparison using Null values – Logical connectives – AND, OR and NOT –

Impact on SQL Constructs, Outer Joins, Disallowing NULL values, Complex Integrity Constraints in SQL Triggers and Active Data bases.

UNIT III

Introduction to Schema Refinement – Problems Caused by redundancy, Decompositions – Problem related to decomposition, Functional Dependencies - Reasoning about FDS, Normal Forms – FIRST, SECOND,THIRD Normal forms – BCNF –Properties of Decompositions-Loss less-join Decomposition, Dependency preserving Decomposition, Schema Refinement in Data base Design – Multi valued Dependencies – FOURTH Normal Form, Join Dependencies, FIFTH Normal form, Inclusion Dependencies.

UNIT IV

Overview of Transaction Management: The ACID Properties, Transactions and Schedules, Concurrent Execution of Transactions – Lock Based Concurrency Control, Deadlocks – Performance of Locking – Transaction Support in SQL.

Concurrency Control: Serializability, and recoverability – Introduction to Lock Management – Lock Conversions, Dealing with Dead Locks, Specialized Locking Techniques – Concurrency Control without Locking.

Crash recovery: Introduction to Crash recovery, Introduction to ARIES, the Log, Other Recovery related Structures, the Write-Ahead Log Protocol, Check pointing, recovering from a System Crash, Media recovery

UNIT V

Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing – Clustered Indexes, Primary and Secondary Indexes, Index data Structures – Hash Based Indexing, Tree based Indexing, Comparison of File Organizations.

Storing data: Disks and Files: -The Memory Hierarchy – Redundant Arrays of Independent Disks.

UNIT VI

Tree Structured Indexing: Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM) B+ Trees: A Dynamic Index Structure, Search, Insert, Delete.

Hash Based Indexing: Static Hashing, Extendable hashing, Linear Hashing, Extendible vs. Linear Hashing.

TEXT BOOKS

1. Data base Management Systems, Raghu Ramakrishnan, Johannes Gehrke, TMH, 3 rd Edition, 2003.
2. Data base System Concepts, A.Silberschatz,H.F. Korth, S.Sudarshan,McGraw hill, VI edition,2006.
3. Fundamentals of Database Systems 5th edition., Ramez Elmasri, Shamkant B.Navathe,Pearson Education,2008.

REFERENCE BOOKS

1. Database Management System Oracle SQL and PL/SQL,P.K.DasGupta,PHI.
2. Database System Concepts,Peter Rob & Carlos Coronel,Cengage Learning,2008.
3. Database Systems, A Practical approach to Design Implementation and Management Fourth edition, Thomas Connolly, Carolyn Begg,Pearson education.
4. Database Principles, Programming, and Performance, P.O'Neil, E.O'Neil, 2 nd ed.,ELSEVIER
5. Fundamentals of Relational Database Management Systems,S.Sumathi,S.Esakkirajan, Springer.
6. Introduction to Database Management,M.L.Gillenson and others,Wiley Student Edition.
7. Database Development and Management,Lee Chao,Auerbach publications,Taylor & Francis Group.
8. Introduction to Database Systems,C.J.Date,Pearson Education.

WEB REFERENCES:

1. nptel.ac.in/courses/106106093
2. nptel.ac.in/courses/106104135

I MCA II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA2TH02	4	-	-	40	60	100	4
OBJECT ORIENTED PROGRAMMING							

COURSE OBJECTIVE:

1. To understand the difference between Procedure Oriented Programming and Object-Oriented Programming.
2. Understand the importance of Exception Handling and create robust programs in Java which can withstand runtime errors.
3. To learn how to create Graphical User Interface applications in Java.
4. To learn how to create lightweight multithreaded applications with synchronization.
5. To learn how to develop web applications using java Applets.

COURSE OUTCOMES:

- CO 1:** To be able to analyse the real world problems in an Object Oriented way.
- CO 2:** Apply Encapsulation, Inheritance and Polymorphism features of Java appropriately to solve problems.
- CO 3:** To be able to create Java console, GUI and Web applications in Java.
- CO 4:** To be able to create lightweight multithreaded applications with synchronization.

UNIT I

Basics of Object Oriented Programming(OOP): Problems with Procedure Oriented Programming and Need for OO paradigm. A way of viewing world-Agents, responsibility, messages, methods, classes and instances, class hierarchies (Inheritance), method binding, overriding and exceptions, summary of oops concepts, coping with complexity, abstraction mechanisms.

UNIT II

Java Basics: Data types, variables, scope and lifetime of variables, arrays, operators, expressions, control statements, type conversion and casting, 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.

UNIT III

Packages, Interfaces and Inheritance: Defining, creating and accessing a package, understanding CLASSPATH, importing packages. Differences between classes and interface. Defining an interface, applying interfaces, variables in interface and extending interfaces. Hierarchical abstraction, base class objects, Subclass, sub type, substitutability, forms of inheritance, Specialization, specification, construction extension, limitation, combination, benefits of inheritance, costs of inheritance. Member access rules, Super uses, using final with inheritance, polymorphism, abstract classes.

UNIT IV

Exception handling and Multithreading: Concept of exception handling, benefits of exception handling, Termination or presumptive models, exception hierarchy, Usage of try, catch, throw, throws and finally, built in exceptions, creating own exception sub classes. Differences between multithreading and multitasking, thread life cycle, creating threads, synchronizing threads, daemon threads, thread groups.

UNIT V

Event Handling, AWT and Swings: 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, checkbox, checkbox groups, choice, lists, panels-scroll pane, dialogs, menu bar, graphics, layout manager - layout manager types - border, grid, flow, card and grid bag.

UNIT VI

Applets and Swings: Concepts of applets, differences between applets and applications, life cycle of an applet, types of applets, creating applets, passing parameter to applets. Limitation of AWT, MVC architecture, components, containers, exploring swing- JApplet, JFrame and Component, Icons and Labels, text fields, buttons - The JButton class, Check boxes, Radio buttons, Combo boxes, Tabbed Panes, Scroll Panes, Trees, and Tables.

TEXT BOOKS

1. Java-The complete reference,7/e, Herbert schildt, TMH.

REFERENCE BOOKS

1. JAVA:How to program, 8/e, Dietal , Dietal,PHI.
2. Introduction of programming with JAVA,S.Dean,TMH.
3. Introduction to Java programming, 6/e, Y.Daniel Liang, Pearson.
4. Core Java 2, Vol 1(Vol 2) Fundamentals(Advanced), 7/e, Cay.S.Horstmann,Gary Cornell,Pearson.
5. Big Java2,3/e, Cay.S. Horstmann,Wiley.
6. Object Oriented Programming through Java, P.Radha Krishna, University Press.
7. JAVA&Object Orientation an Introduction, 2/e, John Hunt, Springer.
8. Introduction to JAVA Programming, 7/e, Y.Daniel Liang, Pearson.
9. AVA Programming and OO Application Development , Johnson, Cengage Learning.
10. First Encounter with JAVA, S.P.Bhuta, SPD
11. JAVA for Professionals , B.M.Harwani, SPD.
12. Program with JAVA, Mahesh Bhave, Palekan, Pearson.
13. Programming with JAVA, 3/e, E.Balaguruswamy, TMH.

WEB REFERENCES:

1. <http://www.w3schools.in/java/>
2. <https://docs.oracle.com/javase/tutorial/>
3. <https://docs.oracle.com/javase/tutorial/java/>

I MCA II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA2TH03	4	-	-	40	60	100	4
OPERATING SYSTEMS							

COURSE OBJECTIVE:

Operating systems are an essential part of any computer system. It provides a clear description of the concepts that underlie operating systems.

COURSE OUTCOMES:

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

- CO 1:** Classify different types of systems like distributed system, special purpose system and the working principle of system calls.
- CO 2:** Compute the waiting time and turnaround time of different CPU scheduling algorithms and comparison among all the CPU scheduling algorithms.
- CO 3:** Examine the page faults for different Page Replacement algorithms and comparison among all the Page Replacement algorithms, and methods to handle deadlocks.
- CO 4:** Differentiate among different File System implementation and directory implementation.
- CO 5:** Assess different types of disk scheduling algorithms, distinguish different protection and security mechanisms.

UNIT I

INTRODUCTION: Operating system structures – simple batch, Multi programmed, Time shared, Personal computer, Parallel, Distributed systems, Real time systems, system components, Operating System services, System calls, Virtual machines, System Design and Implementation.

UNIT II

PROCESS SCHEDULING:

Process Concept: Overview – Process Scheduling – Operations on Processes – co-operating processes, Inter process Communication – Examples of IPC Systems – Communication in Client and Server Systems. **Threads:** Overview, Multi-threading models, Threaded Libraries, Java Threads, Threading issues, case studies of UNIX, LINUX, Windows.

UNIT III

CPU SCHEDULING: Basic concepts, Scheduling criteria, Scheduling algorithms: FCFS scheduling, SJF, SRTF, Priority Scheduling, Round Robin scheduling, Multilevel queue scheduling, Multilevel feedback queue scheduling, Multiprocessor scheduling.

DEADLOCKS: System Model, Deadlock Characterization, Methods for handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

UNIT IV

PROCESS MANAGEMENT AND SYNCHRONIZATION: The critical - section problem, Peterson's solution, Synchronization hardware, Semaphores, Classic problems of Synchronization, Monitors, Synchronization examples.

UNIT V

MEMORY MANAGEMENT AND VIRTUAL MEMORY: Logical versus Physical Address space, Swapping, Contiguous Allocation, paging, Segmentation, Segmentation with Paging, Demand Paging, Copy-on-Write, Page replacement algorithms, Allocation of Frames, Thrashing.

FILE SYSTEM INTERFACE AND IMPLEMENTATION: File Concept, Access Methods, Directory Structure, File System Mounting, File Sharing, Protection, File System Structure, Allocation Methods, Free Space Management, Directory Management, Directory Implementation, Efficiency and Performance, Recovery, Log structured File System.

UNIT VI

MASS STORAGE STRUCTURE: Disk structure, Disk Scheduling, Disk and Swap Space Management, RAID Structure, Stable Storage structure.

PROTECTION : Goals of protection, Principles of protection, Domain of protection, Implementation of Access Matrix, Revocation of Access Rights, Capability Based Systems.

SECURITY: Introduction part of Security problem, Program Threats, System and Network Threats, Cryptography.

TEXT BOOKS

1. Operating System Concepts, 7/e, Abraham Silberschatz, Peter Galvin, Greg Gagne.

REFERENCE BOOKS

1. Operating Systems, 6/e , William Stallings, PHI/ pearson.
2. Operating Systems Design and Implementation, 3/e, Tanenbaum, WoodHull.

WEB REFERENCES:

1. nptel.ac.in/courses/106108101
2. nptel.ac.in/courses/106106144

I MCA II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA2TH04	3	1	-	40	60	100	4
OPERATIONS RESEARCH							

COURSE OBJECTIVE:

To solve a problem by using scientific approach, OT techniques. To learn decision making in a competitive situation. To learn planning and scheduling large projects in a smooth manner by determining critical factors and co-ordinating various parts of overall job.

UNIT I

The origin of OR, the nature of OR, the impact of OR, Definition, Characteristics and Phrases, scientific method. Types of models, general methods for solving, operations research models.

UNIT II

Linear Programming: Introduction, linear programming, General statement of Linear Programming problems, Solution to linear programming problems- Graphic Method, Some special cases. Simplex method, Solution of Maximization problems, Solution to minimization problems. Big-M Method.

UNIT III

Transportation problem: Formulation, optimal solution, unbalanced transportation, assignment problem: formulation, optimal solution, variations problem, degeneracy i.e. non square (MXN) matrix, restrictions.

UNIT IV

Queuing Theory: Queue description, characteristics of a queuing model, study state solutions of M/M/1: Alpha model, M/M/1; N Model.

UNIT V

Theory of Games: Introduction, minmax(maximum), criterion and optimal strategy solution of games with saddle points, rectangular without saddle points. Sequencing: Introduction, optimal solution for processing each of n jobs through three machines, travelling salesman problem(i.e.) shortest acyclic route models.

UNIT VI

Project Management: PERT and CPM, difference between PERT and CPM, PERT/CPM network components and precedence relations, Time Estimates for activities.

TEXT BOOKS

1. Operations Research, S.D. Sharma, Ramnath, & Kedarnath co, Meerut.
2. Operations Research, An introduction, 8/e, Taha, Pearson.

REFERENCE BOOKS

1. Operations Research, P.K. Gupta, D.S. Hira, S. Chand.
1. Operations Research, R.D. Asrhedkar, R.V. Kulkarni.
2. Operations Research, Problems & solutions, 3/e, JK Sharma, Macmillan.
3. Operations Research, 8/e, Hillier, Liberman, TMH.
4. Operations Research, 2/e, Panneerselvam.

I MCA II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA2TH05	4	-	-	40	60	100	4
ORGANIZATIONAL STRUCTURE AND PERSONNEL MANAGEMENT							

COURSE OBJECTIVE:

To equip the student with the basic structure of organisation and to provide knowledge relating to recruitment, selection, training and motivation of employees in the organisation.

COURSE OUTCOMES:

After completion of this course, the student is able to

- CO 1:** List out the Management concepts and functions.
- CO 2:** Discuss the theories of organization and importance of communication
- CO 3:** Explain different functions of HR manager.
- CO 4:** Discuss the concept of strategic management and develop vision, mission.
- CO 5:** Analyse the various functions under the concept of staffing.
- CO 6:** Importance of controlling principles and practices,

UNIT I

Introduction to management- concepts, nature, definition, Importance of management- Principles of management-levels of management-evolution of management- Management Vs Administration- managerial roles and functions.

UNIT II

Organisational design- nature, importance of organising, types of organisational structures, theories of organisation, formal and informal organisations, difficulties due to informal organisation.

UNIT III

Human Resource management- objectives, functions of hrm, duties and responsibilities of hr managers, recruitment, sources of recruitment, selection procedure, tests and interview techniques, training and development- methods of training, transfer, promotion and its policies.

UNIT IV

Strategic management- Introduction- study of strategic management- corporate planning- environmental scanning- SWOT analysis- challenges in LPG

UNIT V

Motivation- need, importance, theories of motivation,-Principles of motivation- types of motivation- communication- concept-need-importance-process- barriers of communication- types and methods of communication-essentials of effective communication.

UNIT VI

Contemporary Issues- HRIS, BPO, HR Outsourcing, Exit interviews, HRD, JIT, TQM, benchmarking, value chain analysis, cross culture in HRM.

REFERENCE BOOKS

1. Organization Structure and personal Management, 2/e, Subbarao.P, HPH.
2. Personal and Human Resource Management, Recenzo, Robins, PHI.
3. Business Communications and soft skills, kuberudu B, and Krishna K.s, Excel publications.
4. Management process and Organizational Behaviour, karam pal, I.k.int.
5. Management process and Organizational Behaviour, karam pal, I.K int.
6. Human Resource Management Jyothoi, Oxford.
- 7 Organizations and Management, Agarwal, TMH.
8. Fundamentals of HRM, David A. Decenzo, Stephen R. Robins, Wiley India.at es
9. Organizational Structure and Human Resurce management, Varaprasad, SciTech.
10. Human Resource Management, Chabra. T.N, Dhanpat Rai.
11. Personal Management and Human Resources, Venkat Ratnam, TMH,

I MCA II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA2LB01	-	-	4	40	60	100	2
DATABASE MANAGEMENT SYSTEM LAB							

COURSE OUTCOMES:

After Completion of this course student must be able to

CO1 : Understand, analyse and apply SQL commands like DDL, DML, DCL to perform different Database operations

CO2 : Understand and practice PL/SQL block, control statements and cursors.

CO3 : Develop PL/SQL programs using functions & procedures.

CO4 : Develop PL/SQL programs using packages and Triggers

1. Execute a single line and group functions for a table.
2. Execute DCL and TCL Commands.
3. Create and manipulate various DB objects for a table.
4. Create views, partitions and locks for a particular DB.
5. Write PL/SQL procedure for an application using exception handling.
6. Write PL/SQL procedure for an application using cursors.
7. Write a DBMS program to prepare reports for an application using functions.
8. Write a PL/SQL block for transaction operations of a typical application using triggers.
9. Write a PL/SQL block for transaction operations of a typical application using package.
10. Design and develop an application using any front end and back end tool (make use of ER diagram and DFD).
11. Create table for various relation Implement the query in sql for
a) insertion b) retrieval c) updation d) deletion
12. Creating Views
13. Writing Assertion
14. Writing Triggers
15. Implementing operation on relation using PL/SQL
16. Creating Forms
17. Generating Reports
18. Typical Applications – Banking, Electricity Billing, Library Operation, Pay roll, Insurance, Inventory etc.

I MCA II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA2LB02	-	-	4	40	60	100	2
OBJECT ORIENTED PROGRAMMING LAB							

COURSE OBJECTIVE:

The course provides user interface and application development program implementation using core java principles.

COURSE OUTCOMES:

After completion of this course, the students would be able to

CO1 : Demonstrate his or her ability to write, compile & execute basic java program and also the use of OOP concept i.e. data abstraction & data hiding, encapsulation, inheritance, polymorphism.

CO2 : Create and use interfaces, threads.

CO3 : Handle exceptions and write applets.

CO4 : Knowledge of GUI for developing java components.

Note: 1) Use JDK 1.5 or above on any platform e.g. Windows or Unix.

2) Student is expected to complete any 16 programs.

1. The Fibonacci sequence is defined by the following rule. The first 2 values in the sequence are 1,1.every subsequent value is the sum of the 2 values preceding it. Write A Java Program (WJJP) that uses both recursive and non-recursive functions to print the nth value of the Fibonacci sequence.
2. WJJP to demonstrate wrapper classes and to fix the precision.
3. WJJP that prompts the user for an integer and then prints out all the prime numbers upto that Integer.
4. WJJP that checks whether a given string is a palindrome or not.
Ex. MALAYALAM is a palindrome.
5. WJJP for sorting a given list of names in ascending order.
6. WJJP to check the compatibility for multiplication , if compatible multiply two matrices and find its transpose.
7. WJJP that illustrates how runtime polymorphism is achieved.
8. WJJP to create and demonstrate packages.

9. WJJP, using String Tokenizer class, which reads a line of integers and then displays each integer and the sum of all integers.
10. WJJP that reads on file name form the user then displays information about whether the file exists, whether the file is readable/writable, the type of file and the length of the file in bytes and display the content of the using FileInputStream class.
11. WJJP that displays the number of characters, lines and words in a text/text file.
12. Write an Applet that displays the content of a file.
13. WJJP that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +-*?% operations. Add a text field to display the result.
14. WJJP for handling mouse events.
15. WJJP demonstrating the life cycle of a thread.
16. WJJP that correctly implements Producer-Consumer problem using the concept of Inter Thread Communication.
17. WJJP that lets users create Pie charts. Design your own user interface(with Swings & AWT).
18. WJJP that allows user to draw lines, rectangles and ovals.
19. WJJP that implements a simple client/server application. The client sends data to a server.
The server receives the data, uses it to produce a result and then sends the result back to the client. The client displays the result on the console. For ex: The data send form the client is the radius of a circle and the result produced by the server is the area of the circle.
20. WJJP to generate a set of random numbers between two numbers x1 and x2, and $x1 > 0$.
21. WJJP to create an abstract class named shape, that contains an empty method named numberOfSides(). Provide three classes named Trapezoid, Triangle and Hexagon, such that each one of the classes contains only the method numberOfSides(), that contains the number of sides in the given geometrical figure.
22. WJJP to implement a Queue, using user defined Exception Handling (also make use of throw, throws).
23. WJJP that creates 3 threads by extending Thread class. First thread displays “Good Morning” Every 1 sec, the second thread displays “Hello” every 2 seconds and the third displays “Welcome” every 3 seconds. (Repeat the same by implementing Runnable).
24. Create an inheritance hierarchy of Rodent, Mouse, Gerbil, Hamster etc. In the base class Provide methods that are common to all Rodents and override these in the derived classes to perform different behaviors, depending on the specific type of Rodent. Create an array of Rodent, fill it with different specific types of Rodents and call your base class methods.

TEXT BOOKS:

1. The Complete Reference Java, 8/e, Herbert Schildt, TMH, 2011 (UNITS : I, IV, V & VI)
2. Programming in JAVA, 2/e, Sachin Malhotra, Saurabh Choudhary, Oxford, 2014, (UNITS: I, II & III)

REFERENCE BOOKS:

1. JAVA Programming, K. Rajkumar, 1/e, Pearson, 2013.
2. Core JAVA, Black Book, Nageswara Rao, Wiley, 1/e, Dream Tech, 2012.

WEB REFERENCES:

<https://www.tutorialspoint.com/java/index.htm>

I MCA II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA2LB03	-	-	4	40	60	100	2
PYTHON PROGRAMMING LAB							

COURSE OBJECTIVES:

1. To introduce students the open source software.
2. To understand why Python is a useful scripting language for developers.
3. To learn how to design and program Python applications.

COURSE OUTCOMES:

After successful completion of this course, the students will be able to

- CO 1:** Develop python applications using strings, functions and Files.
- CO 2:** Demonstrate the use of Python lists and dictionaries.
- CO 3:** Make use of Object Oriented Programing concepts to develop python programs.
- CO 4:** Experiment with GUI based python programs.

LIST OF EXPERIMENTS

1. BASICS

- a) Running instructions in Interactive interpreter and a Python Script.
- b) Write a program to purposefully raise Indentation Error and correct it.

2. OPERATIONS

- a) Write a program to compute distance between two points taking input from the user (Pythagorean Theorem)
- b) Write a program add.py that takes 2 numbers as command line arguments and prints its sum.

3. CONTROL FLOW

- a) Write a Program for checking whether the given number is a even number or not.
- b) Using a for loop, write a program that prints out the decimal equivalents of 1/2, 1/3, 1/4, ,1/10
- c) Write a program using a for loop that loops over a sequence. What is sequence?
- d) Write a program using a while loop that asks the user for a number, and prints a countdown from that number to zero.

4. CONTROL FLOW - CONTINUED

a) Find the sum of all the primes below two million.

Each new term in the Fibonacci sequence is generated by adding the previous two terms.

By starting with 1 and 2, the first 10 terms will be: 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

b) By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

5. DS

a) Write a program to count the numbers of characters in the string and store them in a

dictionary data structure

b) Write a program to use split and join methods in the string and trace a birthday with a dictionary data structure.

6. DS - CONTINUED

a) Write a program combine lists that combines these lists into a dictionary.

b) Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?

7. FILES

a) Write a program to print each line of a file in reverse order.

b) Write a program to compute the number of characters, words and lines in a file.

8. FUNCTIONS

a) Write a function ball collides that takes two balls as parameters and computes if they are colliding. Your function should return a Boolean representing whether or not the balls are colliding.

Hint: Represent a ball on a plane as a tuple of (x, y, r), r being the radius

If (distance between two balls centers) \leq (sum of their radii) then (they are colliding)

b) Find mean, median, mode for the given set of numbers in a list.

9. FUNCTIONS - CONTINUED

a) Write a function nearly_equal to test whether two strings are nearly equal. Two strings a and b are nearly equal when a can be generated by a single mutation on b.

b) Write a function dups to find all duplicates in the list.

c) Write a function unique to find all the unique elements of a list.

10. FUNCTIONS - PROBLEM SOLVING

a) Write a function cumulative product to compute cumulative product of a list of numbers.

b) Write a function reverse to reverse a list. Without using the reverse function.

c) Write function to compute gcd, lcm of two numbers. Each function shouldn't exceed one line.

11. MULTI-D LISTS

- a) Write a program that defines a matrix and prints
- b) Write a program to perform addition of two square matrices
- c) Write a program to perform multiplication of two square matrices

12. MODULES

- a) Install packages requests, flask and explore them. using (pip)
- b) Write a script that imports requests and fetch content from the page. Eg. (Wiki)
- c) Write a simple script that serves a simple HTTPResponse and a simple HTML Page

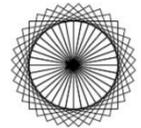
13. OOP

- a) Class variables and instance variable
 - i) Robot
 - ii) ATM Machine

14. GUI, GRAPHICS

Write a GUI for an Expression Calculator using tk

Write a program to implement the following figures using turtle



15. Testing

Write a test-case to check the function even numbers which return True on passing a list of all even numbers

Write a test-case to check the function reverse string which returns the reversed string

16. Advanced

Build any one classical data

Write a program to solve knapsack problem.

- c) Write a program using django web application framework.

WEB REFERENCES:

1. <https://www.edx.org/course/subject/computer-science/python>
2. <https://www.edureka.co/python>



II MCA III SEMESTER SYLLABUS

II YEAR III SEMESTER										
S.No	Course Code	Courses	Cat. Code	Int	Ext	Tot	L	T	P	C
1	19MCA3TH01	Software Engineering	PC	40	60	100	4	-	-	4
2	19MCA3TH02	Computer Networks	PC	40	60	100	4	-	-	4
3	19MCA3TH03	Design And Analysis of Algorithms	PC	40	60	100	3	1	-	4
4	19MCA3TH04	Web Development & Digital Marketing	PC	40	60	100	4	-	-	4
5	19MCA3TH05	Linux Programming	PC	40	60	100	4	-	-	4
6	19MCA3MC01	Community Services(CS)	MC	-	-	-	-	-	-	-
7	19MCA3LB01	Web Development & Digital Marketing Lab	PC	40	60	100	-	-	4	2
8	19MCA3LB02	Linux Programming Lab	PC	40	60	100	-	-	4	2
9	19MCA3LB03	Statistical Programming with R Lab	PC	40	60	100	-	-	4	2
TOTAL				320	480	800	19	1	12	26

II MCA III SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA3TH01	4	-	-	40	60	100	4
SOFTWARE ENGINEERING							

COURSE OBJECTIVES:

This course is intended to provide the students with an overall view over Software Engineering discipline and with insight into the processes of software development.

COURSE OUTCOMES:

After completion of this course the students should be able to

CO 1: Identify generic models of software development process.

CO 2: Understand fundamental concepts of Requirements Engineering and Analysis Modeling.

CO 3: Compare different design techniques and their implementation.

CO 4: List out various testing and maintenance measures.

UNIT I

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, Software myths.

A Generic View of Process: Software Engineering- A Layered Technology, A Process Framework, The Capability Maturity Model Integration (CMMI), Process Patterns, Process Assessment, Personal and Team Process Models.

UNIT II

Process Models: The Waterfall Model, Incremental Process Models, Evolutionary Process Models, the Unified Process, Agile Models-Extreme Programming.

Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

Requirements engineering processes: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

UNIT III

Design Engineering: Design Process and Design Quality, Design Concepts, the Design Model.

Creating an architectural Design: Software Architecture, Data Design, Architectural Styles and Patterns, Architectural Design.

UNIT IV

System models: Context Models, Behavioral models, Data models, Object models, structured methods.

Object-Oriented Design: Objects and object classes, An Object-Oriented design process, Design evolution.

UNIT V

Testing Strategies: A Strategic Approach to Software Testing, Test Strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging.

Product Metrics: Software Quality, Metrics for Analysis Model, Metrics for Design Model, Metrics for Source Code, Metrics for Testing, Metrics for Maintenance.

UNIT VI

Metrics for Process and Projects: Software Measurement, Metrics for Software Quality.

Risk management: Reactive vs Proactive Risk Strategies, Software Risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM Plan.

Quality Management: ISO 9000 quality standards, Six Sigma.

TEXT BOOKS

1. Software Engineering, A Practitioner's Approach- Roger S. Pressman, 7th edition. McGraw-Hill International Edition, 2005(Unit-1, 3, 5, 6)
2. Software Engineering- Somerville, 7th edition, Pearson education.(Unit-2,4)
3. Software Engineering, A Precise approach, Pankaj Jalote, Wiley.

REFERENCE BOOKS

1. Software Engineering- K.K. Agarwal & Yogesh Singh, New Age International Publishers
2. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiely.
3. Systems Analysis and Design- Shely Cashman Rosenblatt, Thomson Publications.
4. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.

WEB REFERENCES

1. nptel.ac.in/courses/106101061

II MCA III SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA3TH02	4	-	-	40	60	100	4
COMPUTER NETWORKS							

COURSE OBJECTIVES:

The course provides knowledge on the fundamental concepts of the Computer Networks and problem solving techniques on Networks.

COURSE OUTCOMES:

After completion of this course, the students should be able to

- CO 1:** Interpret the basic computer network technology.
- CO 2:** Identify the different types of network topologies and protocols.
- CO 3:** Explain the layers of the OSI and TCP/IP model and the functions of each layer.
- CO 4:** Design and implement data link or network layer protocols within a simulated networking environment

UNIT I

Network Hardware Reference Model: Transmission Media, Narrowband ISDN, Broadband ISDN, ATM.

The Data Link Layer: Error detection and correction, Elementary Data Link Protocols, Sliding window protocols.

UNIT II

Channel Allocation Methods: TDM, FDM, ALOHA, Carrier sense Multiple access protocols, Collision Free protocols, IEEE standard BO2 for LANS, Ethernet, Token Bus, Token ring, Bridges.

UNIT III

Network Layer Routing Algorithms: Shortest path, Flow based Distance vector, Link state, Hierarchical, Broadcast routing, Congestion Control algorithms, General principles of congestion control, Congestion prevention polices.

UNIT IV

Internet Working: Tunneling, internetworking, Fragmentation, Network Layer in the internet – IP protocols, IP address, Subnets, Internet control protocols, OSPF, BGP, Internet multicasting, Mobile IP.

Network Layer in the ATM Networks-cell formats, connection setup, routing and switching, service categories and quality of service, ATM LANs.

UNIT V

The Transport Layer: Elements of transport protocols – addressing, establishing a connection, releasing connection, flow control and buffering and crash recovery, End to End protocols: UDP, TCP Protocol and Remote Procedure Call.

UNIT VI

Application Layer: Network Security, Cryptographic Algorithms: DES, RSA. Security Mechanisms: Authentication Protocols, Firewalls, DNS, Protocols-SMTP, MIME, HTTP, SNMP.

TEXT BOOKS

1. Computer Networks Andrew, Tanenbaum, 4/e, Pearson.
2. Data and computer communications, Stallings, 8/e, PHI.

REFERENCE BOOKS

1. Data communications and networking, Forouzan, 4/e, TMH.
2. An Engineering approach to compute networking, Kesha, Pearson.
3. Communication networks, 2/e, Leon-Garcia, TMH.
4. Computer networks, C R Sarma, Jaico, Held, 7/e, Pearson.

WEB REFERENCES

1. https://onlinecourses.nptel.ac.in/noc18_cs38
2. <https://in.udacity.com/course/computer-networking--ud436>
3. <https://www.class-central.com/subject/computer-networking>
4. <https://www.youtube.com/watch?v=3DZLIafbqtQ&list=PL32DBC269EF768F74>

II MCA III SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA3TH03	3	1	-	40	60	100	4
DESIGN AND ANALYSIS OF ALGORITHMS							

COURSE OBJECTIVES:

Learn the student's good principles of algorithm design, elementary analysis of algorithms, and fundamental data structures. The emphasis is on choosing appropriate data structures and designing correct and efficient algorithms to operate on these data structures.

COURSE OUTCOMES:

After completion this course, the student should be able to:

- CO 1:** Analyze worst-case running times of algorithms using asymptotic analysis.
- CO 2:** Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it.
- CO 3:** Describe the greedy, dynamic-programming paradigm and explain when an algorithmic design situation calls for it.
- CO 4:** Explain the major graph algorithms and their analyses. Employ graphs to model engineering problems, when appropriate.

UNIT I

Introduction: Algorithm Specification, Performance Analysis -Space complexity, Time complexity, Asymptotic Notations (Big-oh notation, Omega notation, Theta notation, little-oh notation, little- Omega notation).

Disjoint Sets: Disjoint set operations, union and find algorithms.

UNIT II

Divide and Conquer: General method, Applications: Binary search, Merge sort, Quick sort, Strassen's matrix multiplication.

UNIT III

Greedy method: General method, Applications: 0/1 Knapsack problem, Job sequencing with deadlines, Minimum cost spanning trees, Single source shortest paths.

UNIT IV

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

UNIT V

Backtracking: The General method, Applications: n-queen problem, Sum of subsets, Graph coloring, Hamiltonian cycles.

UNIT VI

Branch and Bound: The method, Applications: 0/1 knapsack problem, Travelling sales person, LC Branch and Bound Solution. NP-Hard and NP-complete problems: Basic concepts, non-deterministic algorithms, cook's theorem.

TEXT BOOKS

1. Fundamentals of Computer Algorithms, Second Edition – Ellis Horowitz, Satraj Sahni and Rajasekaram, Universities Press.

REFERENCE BOOKS

1. Design and Analysis of Algorithms, First Edition – S. Sridhar, Oxford.
2. Design and Analysis of Algorithms, Second Edition – Parag Himanshu Dave, Himanshu Bhalchandra Dave, Pearson Education.
3. Design and Analysis of Computer Algorithms – Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, Pearson Education.
4. Introduction to the Design and Analysis of Algorithms, Third Edition – Anany Levitin, Pearson Education.

WEB REFERENCES

1. <http://www.personal.kent.edu/~rmuhamma/Algorithms/algorithm.html>
2. <https://www.coursera.org/learn/algorithms-divide-conquer>
3. <http://nptel.ac.in/courses/106101060/>
4. <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-design-and-analysis-of-algorithms-spring-2015/index.htm>

II MCA III SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA3TH04	4	-	-	40	60	100	4
WEB DEVELOPMENT & DIGITAL MARKETING							

COURSE OBJECTIVES:

The main objective of the course is to expose the students to different web technologies and prepare him to design, develop and maintain a web site.

COURSE OUTCOMES:

After completion of this course, the students should be able to

- CO 1:** Acquaint with necessary fundamental skills for creating web applications.
- CO 2:** Install and manage web server software and study server side scripting languages like servlets and JSP.
- CO 3:** Acquire the concepts of java beans.
- CO 4:** Understand database access and use it in JSP and servlets.

UNIT I

Review of HTML 5: HTML Common Tags, List, Tables, images, forms, Frames, SVG, CSS3: Images, Rounded Corners, Shadows, Animations.

Client side Scripting: Introduction to JavaScript: JavaScript language, declaring variables, scope of variables, functions, Event handlers (onclick, onsubmit etc.), Document Object Model, Form validation, Simple AJAX application.

UNIT II

Introduction to Servlets: Lifecycle of a Servlet, JSDK, the Servlet API, Types of Servlet (HTTP and Generic), The javax.servlet Package, Reading Servlet parameters, Reading Initialization Parameters, The javax.servlet.HTTP package, Handling, Http Request & Responses, Using Cookies, Session Tracking, Security Issues.

UNIT III

Introduction to JSP: The Problem with Servlets, The Anatomy of a JSP Page, Differences between Servlets and JSP, JSP Processing, JSP Application Design with MVC, Setting Up the JSP Environment: Installing the Java Software Development Kit, Tomcat Server & Testing Tomcat.

UNIT IV

JSP Application Development: Generating Dynamic Content, Using Scripting Elements, Implicit JSP Objects, Conditional Processing – Displaying Values, Using an Expression to Set an Attribute, Declaring Variables and Methods, Error Handling and Debugging, Sharing Data between JSP Pages, Requests and Users.

UNIT V

Java Beans: Introduction to Java Beans, Advantages of Java Beans, Introspection, Using Bound Properties, Bean Info Interface, Constrained Properties, Persistence, Customizers, Java Beans API, Deploying JAVA Beans in a JSP Page.

UNIT VI

Social Media Marketing:

An Introduction to Digital Marketing, Start with the Customer and Work Backward, What Are the 3i Principles.

Search Engine Optimization: An Introduction, Search Engine Result Pages: Positioning, Search Behavior: Stage 1: Goals. Stage 2: On-Page Optimization, Stage 3: Off-Page Optimization, Stage 4: Analyze

TEXT BOOKS

1. Internet and World Wide Web: How to program,6/e, Dietel, Dietel, Pearson(2008).
2. The Complete Reference Java2, 8/e, Patrick Naughton, Herbert Schildt, TMH.
3. Java Server Faces, Hans Bergstan, O'reilly.
4. The Art of Digital Marketing: The Definitive Guide to Creating Strategic, Targeted, and Measurable Online Campaigns- Ian Dodson-John Wiley & Sons

REFERENCE BOOKS

1. Web Programming, building internet applications, 2/e, Chris Bates, Wiley Dreamtech
2. Programming world wide web, Sebesta, PEA
3. Web Tehnologies, 2/e, Godbole, kahate, TMH
4. An Introduction to Web Design , Programming ,Wang,Thomson

WEB REFERENCES

1. <https://www.coursera.org/learn/html-css-javascript-for-web-developers>.
2. <https://www.coursera.org/learn/web-applications-php>.
3. <https://www.udemy.com/javawebtut/>.
4. <https://www.coursera.org/learn/desenvolvimento-agil-com-java-avancado/lecture/LUXm0/criando-java-servlets>.

II MCA III SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA3TH05	4	-	-	40	60	100	4
LINUX PROGRAMMING							

COURSE OBJECTIVES:

In this course student will learn about

1. Various utilities and filters in Unix operating system.
2. Programming in networking environment.
3. Various operating system functionalities including process management, file management, networking etc.
4. Inter Process Communication.
5. Network Programming through sockets.

COURSE OUTCOMES:

At the end of this course student will be able to

- CO 1:** Utilize various utilities in Unix operating system.
CO 2: Implement various OS functionalities using system calls.
CO 3: Implement Inter Process Communication.
CO 4: Implement network programming using sockets.
CO 5: Implement Remote Procedure Call

UNIT-I

Review of Unix Utilities and Shell Programming: -File handling utilities, security by file permissions, process utilities, disk utilities, networking commands, backup utilities, text processing utilities,

UNIT-II

Working with the Bourne shell-, What is a shell, shell responsibilities, pipes and input redirection, output redirection, here documents, the shell as a programming language, shell meta characters, shell variables, shell commands, the environment, control structures, shellscript examples.

UNIT-III

Unix Files: Unix file structure, directories, files and devices, System calls, library functions, low level file access, usage of open, create, read, write, close, lseek, stat, fstat, octl, umask, dup,dup2. The standard I/O (fopen, fclose, fflush, fseek, fgetc, getc, getchar, fputc, putc, putchar,fgets, gets), formatted I/O, stream errors, streams and file descriptors, file and directory maintenance (chmod, chown, unlink, link, symlink, mkdir, rmdir, chdir, getcwd), Directory handling system calls (opendir, readdir, closedir, rewinddir, seekdir, telldir)

UNIT-IV

Unix Process: Threads and Signals: What is process, process structure, starting new process, waiting for a process, zombie process, process control, process identifiers, system call interface for process management, -fork, vfork, exit, wait, waitpid, exec, system, Threads, - Thread creation, waiting for a thread to terminate, thread synchronization, condition variables, cancelling a thread, threads vs. processes, Signals-, Signal functions, unreliable signals, interrupted system calls, kill and raise functions, alarm, pause functions, abort, sleep functions.

UNIT-IV

Data Management: Management Memory (simple memory allocation, freeing memory) file and record locking (creating lock files, locking regions, use of read/ write locking, competing locks, other commands, deadlocks). Interprocess Communication: Introduction to IPC, IPC between processes on a single computer system, IPC between processes on different systems, pipes, FIFOs, streams and messages, namespaces, introduction to three types of IPC (system-V)-message queues, semaphores and shared memory

Message Queues-: IPC, permission issues, Access permission modes, message structure, working message queues, Unix system-V messages, Unix kernel support for messages, Unix APIs for messages, client/server example.

UNIT-VI

Semaphores: -Unix system-V semaphores, Unix kernel support for semaphores, Unix APIs for semaphores, file locking with semaphores. Shared Memory: -Unix system-V shared memory, working with a shared memory segment, Unix kernel support for shared memory, Unix APIs for shared memory, semaphore and shared memory example.

TEXTBOOKS:

- 1 Unix and shell Programming, N B Venkateswarlu, Reem
2. Unix Concepts and Applications, 3/e, Sumitabha Das, TMH

REFERENCEBOOKS:

1. Unix and shell Programming, Sumitabha Das, TMH
2. A Beginner's Guide to Unix, N.P.Gopalan, B.Sivaselva, PHI
3. Unix Shell Programming, Stephen G.Kochan, Patrick Wood, 3/e, Pearson
4. Unix Programming, Kumar Saurabh, Wiley,India
5. Unix Shell Programming, Lowell Jay Arthus & Ted Burns,3/e, GalGotia
6. Nix Concepts and Applications, Das, 4/e, TMH

WEB REFERENCES

1. <https://www.learnshell.org>
2. <https://www.udemy.com/shellprogramming/>
3. <https://www.edureka.co/unix>

II MCA III SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA3LB01	-	-	4	40	60	100	2
WEB DEVELOPMENT & DIGITAL MARKETING LAB							

COURSE OBJECTIVES:

The main objective of the lab course is to expose the students to different programming aspects related to web designing using different technologies.

1. Understand web page site planning, designing, and maintenance.
2. Develop web sites which are secure and dynamic in nature and writing scripts which get executed on server as well.
3. Study the actual advanced Web methodologies, specifications and techniques.
4. Acquire the skills necessary to design, implement and deploy complex Web sites and applications.
5. Understands the concepts of PHP and AJAX to develop web pages & website.

COURSE OUTCOMES:

After Completion of this course, the student should be able to:

CO 1: Get practical exposure on HTML, XHTML, CSS, JavaScript, XML and other web technologies.

CO 2: Get practical exposure to develop XML Technologies such as XML Schemas, XSLT.

CO 3: Get practical exposure to develop Server-Side Programming using Servlets and JSP's.

CO 4: Develop a web page & web site using AJAX and PHP.

Week-1

Design the following static web pages required for an online book store web site.

HOME PAGE

The static home page must contain three frames.

Top frame:

Logo and the college name and links to Home page, Login page, Registration page, Catalogue page and Cart page (the description of these pages will be given below).

Left frame:

At least four links for navigation, which will display the catalogue of respective links.

For e.g.: When you click the link "CSE" the catalogue for CSE Books should be displayed in the Right frame.

Right frame:

The pages to the links in the left frame must be loaded here. Initially this page contains description of the web site.

Week-2:

1) Login Page:

This page looks like below:

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	Login :			
	2) CATALOGUE PAGE:			
	The catalogue page should contain the details of all the books available in the web site in a table.			
	The details should contain the following:			
	1. Snap shot of Cover Page.		2. Author Name.	
	3. Publisher.		4. Price.	
5. Add to cart button.				

Note: Week 2 contains the remaining pages and their description.

3) CART PAGE: The cart page contains the details about the books which are added to the cart. The cart page should look like this:

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	Book name	Price	Quantity	Amount
	Java 2	\$35.5	2	\$70
	XML bible	\$40.5	1	\$40.5
	Total amount			\$130.5

4) REGISTRATION PAGE:

Create a “*registration form*” with the following fields

- 1) Name (Text field)
- 2) Password (password field)
- 3) E-mail id (text field)
- 4) Phone number (text field)
- 5) Sex (radio button)
- 6) Date of birth (3 select boxes)
- 7) Languages known (check boxes – English, Telugu, Hindi, Tamil)
- 8) Address (text area)

WEEK 3:

VALIDATION:

Write *JavaScript* to validate the following fields of the above registration page.

1. Name (Name should contains alphabets and the length should not be less than 6 characters).

2. Password (Password should not be less than 6 characters length).
3. E-mail id (should not contain any invalid and must follow the standard pattern name@domain.com)
4. Phone number (Phone number should contain 10 digits only).

Week-4:

Design a web page using CSS (Cascading Style Sheets) which includes the following:

Use different font, styles: In the style definition you define how each selector should work (font,color etc.). Then, in the body of your pages, you refer to these selectors to activate the styles.

For example:

```
BODY { background-image:url(myimage.gif); }
<HTML>
<HEAD>
<style type="text/css">
B.headline { color:red;font-size:22px;font-family:arial;text-decoration:underline }
</style>
</HEAD>
<BODY>
<b>This is normal bold</b><br>
Selector { cursor:value }
```

For example:

```
<html>
<head>
<style type="text/css">
.xlink { cursor:crosshair }
.hlink { cursor:help }
</style>
</head>
<body>
<b>
<a href="mypage.htm" class="xlink">CROSS LINK</a>
<br>
<a href="mypage.htm" class="hlink">HELP LINK</a>
</b>
</body>
</html>
<b class="headline">This is headline style bold</b>
</HTML>
```

- 2) Set a background image for both the page and single elements on the page.
You can define the background image for the page like this:
- 3) Control the repetition of the image with the background-repeat property.
As background-repeat: repeat Tiles the image until the entire page is filled, just like an ordinary background image in plain HTML.

- 4) Define styles for links as A:link
 A:visited
 A:active
 A:hover

Example:

```
<style type="text/css"> A:link {text-decoration: none}
A:visited {text-decoration: none} A:active {text-
decoration: none}
A:hover {text-decoration: underline; color: red;} </style>
```

- 5) Workwith layers: For example:

LAYER 1 ON TOP:

```
<div style="position:relative; font-size:50px; z-index:2;">LAYER 1</div> <div
style="position:relative; top:-50; left:5; color:red; font-size:80px; zindex: 1">LAYER
2</div>
```

LAYER 2 ON TOP:

```
<div style="position:relative; font-size:50px; z-index:3;">LAYER 1
</div> <div style="position:relative; top:-50; left:5; color:red; font-
size:80px; zindex: 4">LAYER 2</div>
```

- 6) Add a customized cursor:

Selector {cursor:value}

For Example:

```
<html>
<head>
<style type="text/css">
.xlink{cursor:crosshair}
.hlink{cursor:help}
</style>
<body>
<b>
<a href="mypage.htm"class="xlink">CROSS LINK</a>
<br>
<a href="mypage.htm"class="hlink">HELP LINK</a>
</b>
</body>
</html>
```

Week 5:

Write an XML file which will display the Book information which includes the following:

- | | |
|----------------------|-------------------|
| 1) Title of the book | 2) Author Name |
| 3) ISBN number | 4) Publisher name |
| 5) Edition | 6) Price |

Write a Document Type Definition (DTD) to validate the above XML file. Display the XML file as follows.

The contents should be displayed in a table. The header of the table should be in color GREY. And the Author names column should be displayed in one color and should be capitalized and in bold. Use your own colors for remaining columns. Use XML schemas XSL and CSS for the above purpose.

Note: Give at least for 4 books. It should be valid syntactically.

Week-6:**VISUAL BEANS:**

Create a simple visual bean with a area filled with a color. The shape of the area depends on the property shape. If it is set to true then the shape of the area is Square and it is Circle, if it is false. The color of the area should be changed dynamically for every mouse click. The color should also be changed if we change the color in the “property window“.

Week-7:

- 1) Install TOMCAT web server and APACHE.
While installation assign port number 4040 to TOMCAT and 8080 to APACHE. Make sure that these ports are available i.e., no other process is using this port.
- 2) Access the above developed static web pages for books web site, using these servers by putting the web pages developed in week-1 and week-2 in the document root.
Access the pages by using the urls :
<http://localhost:4040/rama/books.html> (for tomcat)
<http://localhost:8080/books.html> (for Apache)

Week-8:**User Authentication:**

Assume four users user1, user2, user3 and user4 having the passwords pwd1,pwd2,pwd3 and pwd4 respectively. Write a servlet for doing the following.

1. Create a Cookie and add these four user id's and passwords to this Cookie.
2. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies.

If he is a valid user(i.e., user-name and password match) you should welcome him by name(user-name) else you should display “ You are not an authenticated user “.

Use init-parameters to do this. Store the user-names and passwords in the webinf.xml and access them in the servlet by using the getInitParameters() method.

Week 9:

Install a database(Mysql or Oracle). Create a table which should contain at least the following fields: name, password, email-id, phone number(these should hold the data from the registration form). Practice ‘JDBC’ connectivity.

Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them. Experiment with various SQL queries.

Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page (week2).

Week-10:

Write a JSP which does the following job:

Insert the details of the 3 or 4 users who register with the web site (week9) by using registration form. Authenticate the user when he submits the login form using the user name and password from the database (similar to week8 instead of cookies).

Week-11:

HTTP is a stateless protocol. Session is required to maintain the state. The user may add some items to cart from the catalog page. He can check the cart page for the selected items. He may visit the catalogue again and select some more items. Here our interest is the selected items should be added to the old cart rather than a new cart. Multiple users can do the same thing at a time(i.e., from different systems in the LAN using the ip-address instead of localhost). This can be achieved through the use of sessions. Every user will have his own session which will be created after his successful login to the website. When the user logs out his session should get invalidated (by using the method `session.invalidate()`). Modify your catalogue and cart JSP pages to achieve the above mentioned functionality using sessions.

Week-12:

Digital Marketing Applications

TEXT BOOKS

1. Internet and World Wide Web: How to program,6/e, Dietel, Dietel, Pearson(2008).
2. The Complete Reference Java2, 8/e, Patrick Naughton, Herbert Schildt, TMH.
3. Java Server Faces, Hans Bergstan, O'reilly.

WEB REFERENCES

1. <https://www.coursera.org/learn/html-css-javascript-for-web-developers>.
2. <https://www.coursera.org/learn/web-applications-php>.
3. <https://www.udemy.com/javawebtut/>.
4. <https://www.coursera.org/learn/desenvolvimento-agil-com-java-avancado/lecture/LUXm0/criando-java-servlets>.

II MCA III SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA3LB02	-	-	4	40	60	100	2
LINUX PROGRAMMING LAB							

COURSE OBJECTIVES:

This subject provides professional skills and knowledge for working in a UNIX environment. UNIX is a widely popular operating system for computing platforms ranging from tablet PCs to server farms. In this subject, students acquire the fundamentals of UNIX and how to perform system programming in the UNIX shell. These skills are essential for professional profiles as diverse as IT system administrators, system developers, data engineers, software engineers, network administrators, and IT managers at large.

COURSE OUTCOMES:

After completion of this course, student should be able to:

- CO 1:** Utilize the UNIX environment effectively to perform a range of system-level tasks.
- CO 2:** Analyze and apply shell scripts of medium complexity to solve system-level problems and form an appropriate skill set.
- CO 3:** Construct and apply regular expressions in shell to process text, search files and Validate formats.
- CO 4:** Acquire the knowledge with the Unix environment and Unix Administration

Note: Student needs exposure to all programs, but expected to complete at least 25-30.

1. Unix Utilities – Introduction to Unix File System, vi editor, File Handling Utilities, Security by File Permissions, Process Utilities, Disk Utilities, Networking Commands, Text Processing Utilities, Backup Utilities, and General Purpose Utilities.
2. Write a shell script to find the simple interest.
3. Write a shell script to display student grade based on 3 subjects.
4. Write a shell script to perform arithmetic operations using case construct.
5. Write a shell script to find the factorial of a given number.
6. Write a shell script to display prime numbers.
7. Write a shell script to find the addition of two numbers using functions.
8. Write a shell script to count number of characters, words, and lines in a given text using wc command.
9. Write a shell script whether the directory is exist/not.
10. Write a shell script whether the file is exist/not.
11. Write a shell script to display the good morning, good afternoon, good evening and good night depending on the users log on time.
12. Write a shell script which deletes all lines containing the word "UNIX" in the files supplied as arguments to this shell script.
13. Write a shell script which displays a list of all files in the current directory to which you have Read, Write and Execute Permissions.

14. Write a menu-driven program which has the following options: i) append data into the file ii) Search a word in an existing file iii) Display content of the file iv) Display the users v) Display the current directory name vi) Display the current login.
15. Program using system calls: create, open, read, write, close, stat, fstat, lseek, dup, dup2.
16. Program using Standard I/O: fopen, fclose, getchar, putchar, getc, fgetc, putc, fputc, fgets, fputs, fseek.
17. Program using File and Directory maintenance: chmod, chown, unlink, link, symlink, mkdir, rmdir, chdir, getcwd.
18. Program using Directory handling system calls: opendir, closedir, readdir, rewinddir, seekdir.
19. Write a „C“ Program that demonstrate the fork() system call.
20. Write a „C“ Program that demonstrate vfork() system call.
21. Write a „C“ Program to implement zombie process.
22. Write a „C“ Program to implement orphan process.
23. Write a „C“ Program that demonstrates SIGSTOP, SIGQUIT, and SIGINT.
24. Write a „C“ Program that demonstrates SIGSTOP, SIGCONT, and SIGKILL.
25. Write a „C“ Program to illustrate sleep functions.
26. Write a „C“ Program to illustrate the pipe function.
27. Write a „C“ Program to implement unnamed pipe by using fork() system call.
28. Write a „C“ Program to illustrate fcntl function.
29. Write a „C“ Program to illustrate message queue.
30. Write a program demonstrating mutual exclusion principle using semaphore/file locks/mutex.
31. Write a „C“ Program to implement client-server application by using shared memory.

TEXT BOOKS

1. Unix Concepts and Applications - Sumitabha Das -2nd Edition - Tata McGraw Hill, 2001.
2. Advanced Programming in the Unix Environment – W.Richard Stevens (2004)
3. Unix Network Programming – W.Richard Stevens(2001).

REFERENCE BOOKS

1. Learning Perl - Randal Schwartz , Tom Phoenix , brian d foy- oreilly, 5th Edition 2008
2. Unix Shell Programming Y.Kanetkar-BPB Pub.,2003.

WEB REFERENCES

1. <https://www.learnshell.org>
2. <https://www.udemy.com/shellprogramming/>
3. <https://www.edureka.co/unix>

II MCA III SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA3LB03	-	-	4	40	60	100	2
STATISTICAL PROGRAMMING WITH R Lab							

1. Write a program to illustrate basic Arithmetic in R
2. Write a program to illustrate Variable assignment in R
3. Write a program to illustrate data types in R
4. Write a program to illustrate creating and naming a vector in R
5. Write a program to illustrate create a matrix and naming matrix in R
6. Write a program to illustrate Add column and Add a Row in Matrix in R
7. Write a program to illustrate Selection of elements in Matrixes in R
8. Write a program to illustrate Performing Arithmetic of Matrices
9. Write a program to illustrate Factors in R
10. Case study of why you need use a Factor in R
11. Write a program to illustrate Ordered Factors in R
12. Write a program to illustrate Data Frame Selection of elements in a Data frame
13. Write a program to illustrate Sorting a Data frame
14. Write a program to illustrate List ? Why would you need a List
15. Write a program to illustrate Adding more elements into a List
16. Write a program to illustrate if-else-else if in R
17. Write a Program to illustrate While and For loops in R
18. Write a program to illustrate Compare and Matrices and Compare vectors
19. Write a program to illustrate Logical & and Logical | operators in R.
20. Write a program to illustrate Function inside function in R
21. Write a program to illustrate to create graphs and usage of plot() function in R
22. Write a program to illustrate Customising and Saving to Graphs in R.
23. Write a program to illustrate some built in Mathematical Functions.
24. Write a program to calculate mean, mode, SD, variance.
25. Write a program to implement linear regression.



II MCA IV SEMESTER SYLLABUS

II YEAR IV SEMESTER										
S.No	Course Code	Courses	Cat. Code	Int	Ext	Tot	L	T	P	C
1	19MCA4TH01	Object Oriented Analysis & Design	PC	40	60	100	4	-	-	4
2	19MCA4TH02	Network Security & Cryptography	PC	40	60	100	4	-	-	4
3	19MCA4TH03	Data Warehousing and Mining	PC	40	60	100	4	-	-	4
4	19MCA4MC01	MOOCS(Mandatory Course)	MC	-	-	-	-	-	-	-
5	Professional Elective – I		PE	40	60	100	4	-	-	4
	19MCA4PE04	Software Design Methodologies								
	19MCA4PE05	Mobile Computing								
	19MCA4PE06	Blockchain Technology								
6	Professional Elective – II		PE	40	60	100	4	-	-	4
	19MCA4PE07	Internet of Things								
	19MCA4PE08	Distributed Operating Systems								
	19MCA4PE09	Network Programming								
7	19MCA4LB01	Object Oriented Analysis & Design(UML) Lab	PC	40	60	100	-	-	4	2
8	19MCA4LB02	Data Warehousing and Mining	PC	40	60	100	-	-	4	2
9	19MCA4LB03	Android App Development Lab	PC	40	60	100	-	-	4	2
TOTAL				320	480	800	20	-	12	26

II MCA IV SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA4TH01	4	-	-	40	60	100	4
OBJECT-ORIENTED ANALYSIS AND DESIGN							

COURSE OBJECTIVES:

The students become familiar with all phases of Object-Oriented Analysis and Design (OOAD). List out the Object Design Principles and understand how to apply them towards implementation.

COURSE OUTCOMES:

After completion this course, the student should be able to:

- CO1:** Analyze software requirements and document those using Use Cases.
- CO2:** Utilize the notation of the UML diagrams such as Use Case, Class, Sequence, Activity, State chart, Deployment and Package Diagrams.
- CO3:** Acquire the knowledge of Library Files, Executable Files, Processors and Devices for a system.
- CO4:** Build the Forward and Reverse Engineering using UML notations.

UNIT I

Introduction to UML: The Meaning of Object-Orientation, Object Identity, Encapsulation, Information Hiding, Polymorphism, Genericity.

Why We Model: Importance of Modeling, Principles of Modeling, Object Oriented Modeling.

Introducing the UML: An Overview of the UML, A Conceptual Model of the UML, Architecture and Software Development Life Cycle.

UNIT II

Basic Structural Modeling:

Classes: Terms and Concepts, Common Modeling Techniques-Modeling the Vocabulary of a System, Modeling the Distribution of Responsibilities in a System, Modeling Nonsoftware Things, Modeling Primitive Types.

Relationships: Terms and Concepts, Common Modeling Techniques-Modeling Simple Dependencies, Modeling Single Inheritance, Modeling Structural Relationships.

Common Mechanisms: Terms and Concepts, Common Modeling Techniques-Modeling New Building Blocks, Modeling Comments, Modeling New Semantics, Modeling New Properties.

Diagrams: Terms and Concepts, Common Modeling Techniques-Modeling Different Views of a System, Modeling Different Levels of Abstraction, Modeling Complex Views.

Advanced Structural Modeling:

Advanced Classes: Terms and Concepts, Common Modeling Techniques-Modeling the Semantics of a Class.

Advanced Relationships: Terms and Concepts, Common Modeling Techniques-Modeling Webs of Relationships.

Interfaces, Types and Roles: Terms and Concepts, Common Modeling Techniques–Modeling the Seams in a System, Modeling Static and Dynamic Types.

Packages: Terms and Concepts, Common Modeling Techniques-Modeling Groups of Elements, Modeling Architectural Views.

Instances: Terms and Concepts, Common Modeling Techniques–Modeling Concrete Instances, Modeling Prototypical Instances.

Class Diagrams: Terms and Concepts, Common Modeling Techniques-Modeling Simple Collaborations, Modeling a Logical Database Schema, Forward and Reverse Engineering.

Object Diagrams: Terms and Concepts, Common Modeling Techniques-Modeling Object Structures, Forward and Reverse Engineering.

UNIT III

Basic Behavioral Modeling-I:

Interactions: Terms and Concepts, Common Modeling Techniques-Modeling a Flow of Control.

Interaction Diagrams: Terms and Concepts, Common Modeling Techniques-Modeling Flows of Control by Time Ordering, Modeling Flows of Control by Organization, Forward and Reverse Engineering.

Sequence Diagrams: Terms and Concepts, differences between Collaboration and Sequence Diagrams, Depicting Synchronous Messages with/without Priority Call Back Mechanism Broadcast Message.

Collaboration Diagrams: Terms and Concepts, Depicting a Message, Polymorphism in Collaboration Diagrams, Iterated Messages, use of self in messages.

UNIT IV

Basic Behavioral Modelling-II:

Use Cases: Terms and Concepts, Common Modeling Techniques-Modeling the Behaviour of an Element.

Use Case Diagrams: Terms and Concepts, Common Modeling Techniques-Modeling the Context of a System, Modeling the Requirements of a System, Forward and Reverse Engineering.

Activity Diagrams: Terms and Concepts, Common Modeling Techniques-Modeling a Workflow, Modeling an Operation, Forward and Reverse Engineering.

UNIT V

Advanced Behavioral Modeling:

Events and Signals: Terms and Concepts, Common Modeling Techniques–Modeling a Family of Signals, Modeling Exceptions.

State Machines: Terms and Concepts, Common Modeling Techniques–Modeling the Lifetime of an Object.

Processes and Threads: Terms and Concepts, Common Modeling Techniques–Modeling Multiple Flows of Control, Modeling Interprocess Communication.

Time and Space: Terms and Concepts, Common Modeling Techniques–Modeling Timing Constraints, Modeling the Distribution of Objects, Modeling Objects that Migrate.

State Chart Diagrams: Terms and Concepts, Common Modeling Techniques–Modeling Reactive Objects, Forward and Reverse Engineering.

UNIT VI

Architectural Modeling:

Components: Terms and Concepts, Common Modeling Techniques–Modeling Executables and Libraries, Modeling Tables, Files, and Documents, Modeling an API, Modeling Source Code.

Deployment: Terms and Concepts, Common Modeling Techniques–Modeling Processors and Devices, Modeling the Distribution of Components.

Component Diagrams: Terms and Concepts, Common Modeling Techniques–Modeling Source Code, Modeling an Executable Release, Modeling a Physical Database, Modeling Adaptable Systems, Forward and Reverse Engineering.

Deployment Diagrams: Terms and Concepts, Common Modeling Techniques–Modeling an Embedded System, Modeling a Client/Server System, Modeling a Fully Distributed System, Forward and Reverse Engineering.

Case Study: The Library Application, University Management System.

TEXT BOOKS

1. Grady Booch, James Rumbaugh, Ivar Jacobson : The Unified Modeling Language User Guide, Pearson Education, 2004, 2007, 2009(UNIT-I, UNIT-II, Chapter 1 & 2 of UNIT-III, UNIT-IV, UNIT-V, UNIT -VI)
2. Meilir Page-Jones: Fundamentals of Object-Oriented Design in UML, Pearson Education, 2005 (Chapter 3 & 4 of UNIT-III).

REFERENCE BOOKS

1. Applying UML and Patterns: An introduction to Object–Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.
2. The Unified Modeling Language Reference Manual, 2/e, Rambaugh, Grady Booch, etc., PEA
3. Object Oriented Analysis & Design, Satzinger, Jackson, Thomson
4. Object Oriented Analysis & Design, John Deacon, PEA
5. Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado: UML 2 Toolkit, WILEY-Dreamtech India Pvt. Ltd.

WEB REFERENCES

1. <https://www.uml-diagrams.org/uml-object-oriented-concepts.html>
2. <https://www.lucidchart.com/blog/uml-diagram-templates>
3. <https://www.smartdraw.com/uml-diagram/>
4. <https://www.ibm.com/developerworks/rational/library/769.html>

II MCA IV SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA4TH02	4	-	-	40	60	100	4
NETWORK SECURITY & CRYPTOGRAPHY							

COURSE OBJECTIVES:

This course will enable students

1. To understand the fundamentals of Cryptography.
2. To acquire knowledge on standard algorithms used to provide confidentiality, integrity and authenticity.
3. To understand the various key distribution and management schemes.
4. To understand how to deploy encryption techniques to secure data in **transit** across data networks.
5. To design security applications in the field of Information technology.

COURSE OUTCOMES:

At the end of this course, the student should be able to

CO 1: Analyze the vulnerabilities in any computing system and hence be able to design a Security solution.

CO 2: Identify the security issues in the network and resolve it.

CO 3: Evaluate security mechanisms using rigorous approaches, including theoretical.

CO 4: Compare and Contrast different IEEE standards and Electronic mail security.

CO 5: Understand and use Security & Privacy mechanisms in EDI like E-mail.

UNIT I

Introduction

Security Attacks (Interruption, Interception, Modification and Fabrication), Security Services (Confidentiality, Authentication, Integrity, Non-repudiation, access Control and Availability) and Mechanisms, A model for Internetwork security, Internet Standards and RFCs.

UNIT II

Conventional Encryption

Conventional Encryption Principles, Conventional encryption algorithms, Cipher block modes of operation, Location of encryption devices, key distribution, Approaches of Message Authentication, Secure Hash Functions and HMAC.

UNIT III

Public key Cryptography

Public key cryptography principles, public key cryptography algorithms, digital signatures, digital Certificates, Certificate Authority and key management, Kerberos, X.509 Directory Authentication Service.

UNIT IV

IP Security

IP Security Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations and Key Management.

UNIT V

Web Security

Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET).

Email Privacy

Pretty Good Privacy (PGP) and S/MIME.

UNIT VI

Intruder Types, Detection & Prevention Mechanisms

Intruders, Viruses and related threats.

Fire walls

Firewall Design principles, Trusted Systems, Intrusion Detection Systems.

TEXT BOOKS

1. Network Security Essentials: Applications and Standards, William Stallings, PEA.

REFERENCE BOOKS

1. Network Security & Cryptography, Bernard Menezes, Cengage, 2010.
2. Fundamentals of Network Security, Eric Maiwald, Dream Tech.
3. Network Security: Private Communication in a Public World, Kaufman, Perlman, PEA/PHI.
4. Principles of Information Security, Whitman, Thomson.
5. Cryptography and Network Security, 3/e, Stallings, PHI/PEA.
6. Network Security: The complete reference, Robert Bragg, Mark Rhodes, TMH.
7. Introduction to Cryptography, Buchman, Springer.

WEB REFERENCES:

1. <http://www.nptel.ac.in/noc/individual-course.php>

III MCA V SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA4TH03	4	-	-	40	60	100	4
DATA WAREHOUSING AND MINING							

COURSE OBJECTIVES:

This course helps the students to understand the overall architecture of a data warehouse and techniques and methods for data gathering and data pre-processing by OLAP tool and outlier detection.

COURSE OUTCOMES:

After completion of this course, students should be able to

CO 1: Examine the types of data to be mined and extract knowledge using Data Mining Techniques.

CO 2: Analyze the different operations and techniques involved in Data Warehouse.

CO 3: Identify components in typical Data Warehouse architectures.

CO 4: Compare and contrast different dominant Data Mining Algorithms for classification, Association Analysis and Clustering.

UNIT I

Introduction to Data Mining: What Is Data Mining, Motivating Challenges, The Origins of Data Mining, Data Mining Tasks, Types of Data: Attributes and Measurement, Types of Data Sets, Data Quality: Measurement and Data Collection Issues, Issues Related to Applications.

UNIT II

Data: Data Preprocessing: Aggregation, Sampling, Dimensionality Reduction, Feature Subset Selection, Feature Creation, Discretization and Binarization, Variable Transformation, Measures of similarity and dissimilarity: Basics, Similarity and Dissimilarity between Simple Attributes, Dissimilarities between Data Objects, Similarities between Data Objects, Examples of Proximity Measures.

UNIT III

Data Warehouse and OLAP Technologies: What is a Data Warehouse, A Multidimensional Data Model: From tables to data cubes, Stars, snowflakes, and fact constellations(schemas for multidimensional databases), Examples for defining star, snowflake, and fact constellation schemas, Measures(their categorization and computation), Introducing concept hierarchies, OLAP operations in the multidimensional data model, A starnet query model for querying multidimensional databases, Data Warehouse Architecture: Steps for the design and construction of data warehouses, A three-tier data warehouse

architecture, OLAP server architectures: ROLAP vs. MOLAP vs. HOLAP, SQL extensions to support OLAP operations, Data Warehouse implementation: Efficient computation of data cubes, Indexing OLAP data, Efficient processing of OLAP queries, Metadata repository, Data warehouse back-end tools and utilities.

UNIT IV

Classification: Basic Concepts, Decision Trees, and Model Evaluation: General Approach to Solving a Classification Problem, Decision Tree Induction: How a Decision Tree Works, How to Build a Decision Tree, Methods for Expressing Attribute Test Conditions, Measures for Selecting the Best Split, Algorithm for Decision Tree Induction, Evaluating the Performance of a Classifier: Holdout Method, Random Subsampling, Cross-Validation, Bootstrap Method.

Classification: Alternative Techniques: Bayesian classifier: Bayes Theorem, Using Bayes theorem for classification, Naïve Bayesian classifier, SVM.

UNIT V

Association Analysis: Basic Concepts and Algorithms: Problem Definition, Frequent Itemset Generation: The Apriori principle, Frequent Itemset Generation in the Apriori Algorithm, Candidate Generation and Pruning, Support Counting, Rule Generation: Confidence-Based Pruning, Compact Representation of Frequent Itemsets: Maximal Frequent Itemsets, Closed Frequent Itemsets, FP-Growth algorithms: FP-Tree Representation, Frequent Itemset Generation in FP-Growth Algorithm.

Association Analysis: Advanced Concepts: Handling Categorical Attributes, Handling Continuous Attributes: Discretization-Based Methods, Statistics-Based Methods, Non-discretization Methods.

UNIT VI

Cluster Analysis: Basic Concepts and Algorithms: Overview: What is Cluster Analysis? Different Types of Clustering, Different Types of Clusters, K-means: The Basic K-means Algorithm, K-means and Different Types of Clusters, Strengths and Weaknesses, Agglomerative Hierarchical Clustering: Basic Agglomerative Hierarchical Clustering Algorithm, Specific Techniques, DBSCAN: Traditional Density (center based approach), The DBSCAN algorithm, Strengths and Weaknesses.

TEXT BOOKS

1. Introduction to Data Mining: Pang-Ning tan, Michael Steinbach, Vipin kumar, Addison-Wesley. (UNIT – 1, 2, 4, 5, 6)
2. Data Mining, Concepts and Techniques, 2/e, Jiawei Han, Micheline Kamber, Elsevier, 2006. (UNIT – 3)

REFERENCE BOOKS

1. Data Mining: Introductory and Advanced Topics, Margaret H Dunham, Pearson, 2008.
2. Introduction to Data Mining with Case Studies: GK Gupta; Prentice Hall.
3. Fundamentals of data warehouses, 2/e , Jarke, Lenzerini, Vassiliou, Vassiliadis, Springer.
4. Data Mining Theory and Practice, Soman, Diwakar, Ajay, PHI, 2006.

WEB REFERENCES

1. https://onlinecourses.nptel.ac.in/noc18_cs14
2. www.oracle.com/Data/Warehousing
3. www.databaseanswers.org/data_warehousing.htm
4. <https://www.wileyindia.com/data-warehousing-data-mining.html>

II MCA IV SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA4PE04	4	-	-	40	60	100	4
SOFTWARE DESIGN METHODOLOGIES							

COURSE OBJECTIVES:

1. To make the student understand the concept of Software Design Paradigm.
2. To understand the Design Principles, Quality Attributes of Design, Design Architectures and different scenarios of design.

COURSE OUTCOMES:

After completion of this course, the students should be able to

CO 1: Learn the Software Designing Skills.

CO 2: Analyze the various Design Principles.

CO 3: Understand fundamental concepts of GUIs and user interfaces; understand the basics of modeling and simulation

CO 4: Distinguish various architectural styles & Quality attributes of software design.

UNIT I

Basic Concepts of Design: Introduction, Characteristics of Design Activities, Essential Elements of Designs.

Design Quality: Software Quality Models: Hierarchical Models, Relational Models, The effect of Design on Software Quality: Efficiency, Correctness and Reliability, Portability, Maintainability, Reusability, Interoperability.

UNIT II

Quality Attributes of Software Design: Witt, Baker and Merritt's Design Objectives, Parnas and Weiss's requirements of Good Designs, Quality of Development Process.

Design Principles: Basic rules of Software Design: Causes of difficulties, Vehicles to overcome difficulties, Basic Rules of Software Design.

Design Processes: The context of design in Software Development Process, Generic Design Process: Descriptive Models, Structure of Software Design Methods.

UNIT III

Software Architecture: The notion of Architecture: Architecture in the Discipline of Buildings, Architecture in the Discipline of Computer Hardware.

The General Notion of Architecture: The notion of Software Architecture, Prescriptive Models, Descriptive Models, Multiple View Models, the roles of Architecture in Software Design.

Software Architectural Style: Introductory examples, the Notion of Software Architectural Style.

UNIT IV

Description of Software Architectures: The Visual Notation: Active and Passive Elements, Data and Control Relationships, Decomposition and Composition of Architectural Elements.

UNIT V**Typical Architectural Styles:**

Data Flow: The general data flow styles, the pipe and filter sub-style, the batch sequential processing sub-style.

Independent Components: The general independent components style, the event-based implicit invocation systems sub-style.

Call and Return: The general call and return style, the layered systems sub-style, Data abstraction: The abstract data type and object-oriented sub-styles, Data-centered style, Virtual Machine Architecture.

UNIT VI

Analysis and Evaluation: The Concept of Scenario, Scenarios for Evaluating Modifiability: Scenarios for evaluating reusability, specification of operational profiles, evaluation and analysis of performance, Scenarios for evaluating reusability.

TEXT BOOK

1. Software Design Methodology: From Principles to Architectural Styles, Hongzhu, Elsevier, 2009

REFERENCE BOOKS

1. Software Architecture: Perspectives on an Emerging discipline, Shaw, M. Garlan, PEA, 2008.
2. Software Architecture in Practice, Bass, L., Clements P, Kazman, PEA, 2003.
3. Evaluating Software Architectures: Methods and Case Studies, Clements Kazman, Klien, PEA, 2002.
4. Software Architecture and Design, Bernard Witt, Baker, Merritt, VonNostrandReinhold, NY, 1994.

WEB REFERENCES

1. <https://www.youtube.com/watch?v=3118sJuwOLs>
2. https://www.youtube.com/watch?v=aX4_s5_Hroc
3. <https://www.youtube.com/watch?v=LbBoUjW2Fgs>

II MCA IV SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA4PE05	4	-	-	40	60	100	4
MOBILE COMPUTING							

COURSE OBJECTIVES:

1. To make the student understand the concepts of mobile computing paradigm, its applications and limitations.
2. To understand the typical Mobile Networking infrastructure through GSM.
3. To understand the issues and solutions of various layers of Mobile Networks, i.e, MAC Layer, Network Layer & Transport Layer.
4. To understand the adhoc networks and related concepts.

COURSE OUTCOMES:

After completion of this course, the students should be able to:

- CO 1:** Understand the various types of Wireless Networks.
CO 2: Carry out the applications and architecture of Mobile Computing.
CO 3: Analyze the various Mobile TCP Variants and Mobile IP issues.

UNIT I

Mobile Communications: An Overview, Mobile Communication-Guided transmission, unguided transmission, Mobile Computing: Mobile Computing Architecture, Mobile System Networks.

UNIT II

Mobile Devices and Systems: Cellular Networks, Mobile Smart Phones, Smart Mobiles and Smart Systems, Handheld Pocket Computers, Limitations of Mobile Devices.

UNIT III

GSM and Other 2G Networks: GSM services and system architecture, Radio interfaces of GSM, Protocols of GSM, Localization, Call handling, GPRS system architecture.
Wireless Medium Access Control & 3G, 4G Networks: CDMA, WCDMA, SDMA, FDMA, TDMA, 3G and 4G Networks.

UNIT IV

Mobile IP Network Layer: IP and Mobile IP Network Layers: OSI layer functions, TCP/IP and Internet protocol, Mobile Internet Protocol, Packet delivery and Handover Management, Location Management, Agent Discovery, Mobile TCP.

UNIT V

Introduction to Mobile Adhoc Network: Fixed Infrastructure Architecture, MANET infrastructure architecture, MANET, properties, spectrum, applications; Security in Ad-hoc Network, Wireless sensor networks; Sensor Network Applications.

UNIT VI

Mobile Wireless Short Range Networks and Mobile Internet: Wireless LAN (WLAN) Architecture, IEEE 802.11 protocol layers, Wireless application protocol (WAP) architecture.

TEXT BOOKS

1. "Mobile Computing," RAJ KAMAL, second edition, Oxford(2009).
2. "Mobile Computing, Technology Applications and Service Creation" ASOKE K TALUKDER, HASAN AHMED, ROOPA R YAVAGAL, Second Edition, Mc Graw Hill.
3. "Principles of Mobile Computing," Second Edition, Springer.

REFERENCE BOOKS

1. Jochen Schiller, "Mobile Communications", Addison-Wesley, Second Edition, 2009.
2. Raj Kamal, "Mobile Computing", Oxford University Press, 2007, ISBN: 0195686772
3. ASOKE K TALUKDER, HASAN AHMED, ROOPA R YAVAGAL, "Mobile Computing, Technology Applications and Service Creation" Second Edition, Mc Graw Hill.
4. UWE Hansmann, Lothar Merk, Martin S. Nicklaus, Thomas Stober, "Principles of Mobile Computing," Second Edition, Springer.

WEB REFERENCES

1. <https://www.youtube.com/watch?v=bdKZQ2mUzIY>
2. <https://www.youtube.com/watch?v=WtRpFLx34BY>
3. <https://www.youtube.com/watch?v=AyvnhIB0-Bw>
4. <https://www.youtube.com/watch?v=f7jULYEC9IE>

II MCA IV SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA4PE06	4	-	-	40	60	100	4
BLOCKCHAIN TECHNOLOGY							

COURSE OBJECTIVES:

To understand the concepts of Block Chain, Block Chain networks and Security of Bitcoins.

COURSE OUTCOMES:

After successful completion of this course, the students should be able to:

- CO 1:** Interpret the working procedure of Bitcoins
- CO 2:** Analyze the Transactions in Bitcoin Network
- CO 3:** Analyze the Block Chain and Bitcoin Network
- CO 4:** Analyze the Mining and Consensus in Block Chain

UNIT I

Introduction: What is Bitcoin?, History of Bitcoin, Bitcoin Uses, Users and Their Stories ,Getting Started, Quick Start Getting your first bitcoins, Sending and receiving bitcoins.

How Bitcoin Works: Transactions, Blocks, Mining, and the Block chain, Bitcoin Overview, Bitcoin Transactions, Common Transaction Forms, Constructing a Transaction, Getting the right inputs, Creating the outputs, Adding the transaction to the ledger, Bitcoin Mining, Mining transactions in blocks, Spending the transaction

UNIT II

Keys, Addresses, Wallets: Generating a public key, Bitcoin Addresses-Base58 and Base58Check Encoding, Key Formats .Wallets -Non-Deterministic (Random) Wallets Deterministic (Seeded) Wallets, Mnemonic Code Words, Hierarchical Deterministic Wallets (BIP0032/BIP0044) . Advanced Keys and Addresses - Encrypted Private Keys (BIP0038), Pay To Script Hash (P2SH) and Multi-Sig Addresses, Vanity Addresses, Paper Wallets.

UNIT III

Transactions: Introduction, Transaction Lifecycle - Creating Transactions, Broadcasting Transactions to the Bitcoin Network, Propagating Transactions on the Bitcoin Network, Transaction Structure, Transaction Outputs and Inputs - Transaction Outputs, Transaction Inputs, Transaction Fees, Adding Fees to Transactions. Transaction Chaining and Orphan Transactions, Transaction Scripts and Script Language - Script Construction (Lock + Unlock), Scripting Language, Turing Incompleteness, Stateless Verification. Standard

Transactions - Pay to Public Key Hash (P2PKH), Pay-to-Public-Key, Multi-Signature, Data Output (OP_RETURN) ,Pay to Script Hash (P2SH).

UNIT IV

The Bitcoin Network: Peer-to-Peer Network Architecture , Nodes Types and Roles , The Extended Bitcoin Network , Network Discovery , Full Nodes , Exchanging “Inventory” ,Simplified Payment Verification (SPV) Nodes , Bloom Filters , Bloom Filters and Inventory Updates , Transaction Pools , Alert Messages.

UNIT V

Bitcoin Security: Security principles - Developing Bitcoin Systems Securely, The Root of Trust. User Security Best Practices - Physical Bitcoin Storage , Hardware Wallets , Balancing Risk (loss vs. theft) ,Diversifying Risk , Multi-sig and Governance , Survivability , Conclusion.

The Block chain: Introduction, Structure of a Block, Block Header, Block Identifiers - Block Header Hash and Block Height, The Genesis Block, Linking Blocks in the Block chain, Merkle Trees, Merkle Trees and Simplified Payment Verification (SPV).

UNIT VI

Mining and Consensus: Introduction - Bitcoin Economics and Currency Creation. Decentralized Consensus ,Independent Verification of Transactions, Mining Nodes , Aggregating Transactions into Blocks- Transaction Age, Fees, and Priority ,The Generation Transaction ,Coinbase Reward and Fees ,Structure of the Generation Transaction , Coinbase Data .Constructing the Block Header , Mining the Block - Proof-of-Work Algorithm , Difficulty Representation , Difficulty Target and Re-Targeting .Successfully Mining the Block , Validating a New Block ,Assembling and Selecting Chains of Blocks -Blockchain Forks .Mining and the Hashing Race - The Extra Nonce Solution ,Mining Pools . Consensus Attacks.

TEXT BOOKS:

1. Mastering bitcoin: unlocking digital crypto-currencies, Andreas M. Antonopoulos, 2nd edition, O'Reilly publishers.

II MCA IV SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA4PE07	4	-	-	40	60	100	4
INTERNET OF THINGS							

COURSE OBJECTIVES:

1. To understand this fundamentals of Internet of Things.
2. To build a small low cost embedded system using Arduino / Raspberry Pi or equivalent boards.
3. To apply the concept of Internet of Things in the real world scenario

COURSE OUTCOMES:

- CO 1:** Demonstrate knowledge and understanding of the security and ethical issues of the Internet of Things
- CO 2:** Conceptually identify vulnerabilities involving the Internet of Things
- CO 3:** Develop critical thinking skills
- CO 4:** Conceptually identify recent security attacks.

UNIT I

The Internet of Things: An Overview of Internet of things, Internet of Things Technology, behind IoTs Sources of the IoTs, M2M Communication, Examples of IoTs, Design Principles For Connected Devices.

UNIT II

Business Models for Business Processes in the Internet of Things ,IoT/M2M systems LAYERS AND designs standardizations ,Modified OSI Stack for the IoT/M2M Systems ,ETSI M2M domains and High-level capabilities ,Communication Technologies, Data Enrichment and Consolidation and Device Management Gateway Ease of designing and affordability

UNIT III

Design Principles for the Web Connectivity for connected-Devices, Web Communication protocols for Connected Devices, Message Communication protocols for Connected Devices, Web Connectivity for connected-Devices.

UNIT IV

Internet Connectivity Principles, Internet connectivity, Application Layer Protocols: HTTP, HTTPS, FTP, Telnet.

UNIT V

Data Acquiring, Organizing and Analytics in IoT/M2M, Applications/Services/Business Processes, IOT/M2M Data Acquiring and Storage, Organizing Data, Transactions, Business Processes, Integration and Enterprise Systems.

UNIT VI

Data Collection, Storage and Computing Using a Cloud Platform for IoT/M2M Applications/Services, Storage and Computing Using cloud platform Everything as a service and Cloud Service Models, IOT cloud-based services using the Xively (Pachube/COSM), Nimbits and other platforms Sensor, Participatory Sensing, Actuator, Sensor Network Technology, Sensing the World.

TEXT BOOKS

1. Internet of Things: Architecture, Design Principles And Applications, Rajkamal, McGraw Hill Higher Education.
2. Internet of Things, A. Bahgya and V. Madiseti, Univesity Press, 2015.

REFERNCE BOOKS

1. Designing the Internet of Things, Adrian McEwen and Hakim Cassimally, Wiley.
2. Getting Started with the Internet of Things CunoPfister , Oreilly.

WEB REFERENCES

1. <https://www.youtube.com/watch?v=G4-CtKkrOmc>
2. http://www.cse.wustl.edu/~jain/cse570-13/m_18iot.htm
3. <https://www.youtube.com/watch?v=9ZUFYyXhQm8>
4. <https://www.udemy.com/introduction-to-iot-using-raspberry-pi-2/>

II MCA IV SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA4PE08	4	-	-	40	60	100	4
DISTRIBUTED OPERATING SYSTEMS							

COURSE OBJECTIVES:

To gain knowledge on Distributed operating system concepts that includes architecture, Mutual exclusion algorithms, Deadlock detection algorithms and agreement protocols. The algorithms for implementation of distributed shared memory, recovery and commit protocols.

COURSE OUTCOMES:

After completion of the course, students should be able to

- CO 1:** Student will gain the knowledge on various Operating systems like Windows and Unix.
- CO 2:** Student will gain knowledge on various internals of operating system.
- CO 3:** Describe, contrast and compare differing structures for operating systems.
- CO 4:** Understand and analyse theory and implementation of: processes, resource control (concurrency etc.), physical and virtual memory, scheduling, I/O and files.

UNIT I

Introduction: Introduction to Distributed System, Goals of Distributed system, Hardware and Software concepts, Design issues.

Communication in Distributed System: Layered protocols, ATM Networks, Client – Server model, Remote Procedure Calls and Group Communication, Middleware and Distributed Operating Systems.

UNIT II

Synchronization in Distributed System: Clock synchronization, Mutual Exclusion, Election algorithm, Bully algorithm, Ring algorithm, Atomic Transactions, Deadlock in Distributed Systems, Distributed Deadlock Prevention, Distributed Deadlock Detection.

UNIT III

Processes and Processors in Distributed Systems: Threads, System models, Processors Allocation, Scheduling in Distributed System, Real Time Distributed Systems.

UNIT IV

Distributed File Systems: Distributed file system Design, Distributed file system Implementation, Trends in Distributed file systems. Distributed Shared Memory: What is shared memory, Consistency models, Page based distributed shared memory, shared variables distributed shared memory.

UNIT V

Fault Tolerance: Concepts, Failure Models, Failure Masking by Redundancy.

Process Resilience: Design Issues, Failure Masking and Replication, Agreement in Faulty Systems.

Recovery: Introduction, Check –pointing, Message Logging – Synchronous and Asynchronous, Adaptive Logging.

UNIT VI

Case Study MACH: Introduction to MACH, process management in MACH, communication in MACH, UNIX emulation in MACH.

TEXT BOOK

1. Distributed Operating System – Andrew S. Tanenbaum, PHI.

REFERENCE BOOKS

1. Distributed Systems: Principles of Paradigms : Andrew Tannenbaum And Maarten van steen.
2. Distributed Operating Systems: Concepts and Design : Pradeep K.Sinha.
3. Distributed Operating Systems and Algorithm Analysis : Randy Chow , Theodore Jhonson.

WEB REFERENCES

1. <http://www.cs.colostate.edu/~cs551dl/externalLinks.php>
2. <http://www.personal.kent.edu/~rmuhamma/OpSystems/os.html>
3. <https://www.sanfoundry.com/operating-system-questions-answers-distributed-operating-system/>
4. <https://link.springer.com/journal/446>
5. <https://www.ukessays.com/.../the-distributed-operating-system-info...>
6. <https://www.youtube.com/watch?v=sK9MC5GREXg>

II MCA IV SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA4PE09	4	-	-	40	60	100	4
NETWORK PROGRAMMING							

COURSE OBJECTIVES:

In this course student will learn about

1. Various utilities and filters in Unix operating system.
2. Programming in networking environment.
3. Various operating system functionalities including process management, file management, networking etc.
4. Inter Process Communication.
5. Network Programming through sockets.

COURSE OUTCOMES:

At the end of this course student will be able to

- CO 1:** Utilize various utilities in Unix operating system.
- CO 2:** Implement various OS functionalities using system calls.
- CO 3:** Implement Inter Process Communication.
- CO 4:** Implement network programming using sockets.
- CO 5:** Implement Remote Procedure Call

UNIT-I

Introduction to Network Programming: OSI model, Unix standards, TCP and UDP & TCP connection establishment and Format, Buffer sizes and limitation, standard internet services, Protocol usage by common internet application.

UNIT-II

TCP client server: Introduction, TCP Echo server functions, Normal startup, terminate and signal handling server process termination, Crashing and Rebooting of server host shutdown of server host.

UNIT-III

Sockets: Address structures, value – result arguments, Byte ordering and manipulation function and related functions Elementary TCP sockets – Socket, connect, bind, listen, accept, fork and exec function, concurrent servers. Close function and related function. I/O Multiplexing and socket options: I/O Models, select function, Batch input, shutdown function, poll function, TCP Echo server, getsockopt and setsockopt functions. Socket states, Generic socket option IPV6 socket option ICMPV6 socket option IPV6 socket option and TCP socket options.

UNIT-IV

Elementary UDP sockets: Introduction UDP Echo server function, lost datagram, summary of UDP example, Lack of flow control with UDP, determining outgoing interface with UDP. Elementary name and Address conversions: DNS, gethost by Name function, Resolver option, Function and IPV6 support, uname function, other networking information.

UNIT-V

IPC : Introduction, File and record locking, Pipes, FIFOs streams and messages, Name spaces, system IPC, Message queues, Semaphores.

UNIT-VI

Remote Login: Terminal line disciplines, PseudoTerminals, Terminal modes, Control Terminals, rlogin Overview, RPC Transparency Issues.

TEXTBOOK

1. UNIX Network Programming, Vol. I, SocketsAPI, 2nd Edition. - W.Richard Stevens, Pearson Edn. Asia. 2. UNIX Network Programming, 1st Edition, - W.Richard Stevens. PHI.

REFERENCES:

1. UNIX Systems Programming using C++ T CHAN, PHI.
2. UNIX for Programmers and Users, 3rd Edition Graham GLASS, King abls, Pearson Education.
3. Advanced UNIX Programming 2nd Edition M. J. ROCHKIND, Pearson Education

WEB REFERENCES

1. <https://www.udemy.com/Networkprogramming/>
2. <https://www.edureka.co/Networkprogramming>

II MCA IV SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA4LB01	-	-	4	40	60	100	2
OBJECT-ORIENTED ANALYSIS AND DESIGN (UML) LAB							

COURSE OBJECTIVES:

The students acquire the knowledge of various phases of Object-Oriented Analysis and Design (OOAD) Principles and understand how to apply them towards implementation.

COURSE OUTCOMES:

After completion this course, the student should be able to

CO 1: Understand how to model requirements with Use Cases.

CO 2: Utilize the principles to design packages for large scale software projects.

CO 3: Design the Dynamic Behaviour and Structure of the design.

CO 4: Apply the pragmatic approach to Software Design and Development.

S. No.	Name of the Experiment
1	Banking System
2	Automatic Teller Machine (ATM) Application
3	Online Auction System
4	Railway Reservation System
5	Sending Secure File using Cryptography
6	Library Information System
7	University Course Registration System
8	Hospital Management System
9	Stock Inventory System

TEXT BOOK

1. Grady Booch, James Rumbaugh, Ivar Jacobson : The Unified Modeling Language User Guide, Pearson Education(2004).

REFERENCE BOOKS

1. Visual Modeling with Rational Rose 2002 and UML, Terry Quatrini, Pearson Education.
2. UML2 Toolkit, Hans-Erik Eriksson, etc; Wiley.

WEB REFERENCES

1. <https://www.uml-diagrams.org/uml-object-oriented-concepts.html>
2. <https://www.lucidchart.com/blog/uml-diagram-templates>
3. <https://www.smartdraw.com/uml-diagram/>
4. <https://www.ibm.com/developerworks/rational/library/769.html>

III MCA V SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5LB02	-	-	4	40	60	100	2
DATA WAREHOUSING AND MINING LAB							

COURSE OBJECTIVE:

The students acquire the knowledge of Data Mining Tools (weka) for retrieving Data from Warehouse.

SYLLABUS

1. Demonstration of preprocessing on dataset student.arff.
2. Demonstration of preprocessing on dataset labor.arff.
3. Demonstration of Association rule process on dataset contactlenses.arff using apriori algorithm.
4. Demonstration of Association rule process on dataset test.arff using apriori algorithm.
5. Demonstration of classification rule process on dataset student.arff using j48 Algorithm.
6. Demonstration of classification rule process on dataset employee.arff using j48 algorithm.
7. Demonstration of classification rule process on dataset employee.arff using id3 algorithm.
8. Demonstration of classification rule process on dataset employee.arff using naive bayes algorithm.
9. Demonstration of clustering rule process on dataset iris.arff using simple k-Means.
10. Demonstration of clustering rule process on dataset student.arff using simple k-means.

TEXT BOOK

1. Data Mining, Concepts and Techniques, 2/e, Jiawei Han, Micheline Kamber, Elsevier, 2006.

WEB REFERENCES

1. https://onlinecourses.nptel.ac.in/noc18_cs14
2. www.oracle.com/Data/Warehousing
3. www.databaseanswers.org/data_warehousing.htm

II MCA IV SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA4LB03	-	-	4	40	60	100	2
ANDROID APP DEVELOPMENT LAB							

SYLLABUS:

1. Write a Program to display text and toast on button click.
2. Write a Program for login page.
3. Write a Program for registration page with edittext(firstname, lastname), button(register), radiobutton(gender), checkbox(qualification), spinner(country). And display the entered fields in alert dialog on register button click.
4. Write a Program to calculate age (pick date of birth using date picker component).
5. Write a Program to make a call from the entered numbers in edit text.
6. Write a Program to navigate from login page (on signup text click) to Register page.
7. Write a Program to get Image from gallery or camera into image view.
8. Write a Program for navigation drawer with fragments.
9. Write a Program to show static list of names/movies/countries.
10. Write a Program to store contacts using SQLite.



II MCA V SEMESTER SYLLABUS

III YEAR V SEMESTER										
S.No	Course Code	Courses	Cat. Code	Int	Ext	Tot	L	T	P	C
1	19MCA5TH01	Software Testing Methodology	PC	40	60	100	4	-	-	4
2	19MCA5TH02	Artificial Intelligence & Machine Learning	PC	40	60	100	4	-	-	4
3	19MCA5TH03	Cloud Computing	PC	40	60	100	4	-	-	4
4	Professional Elective – III		PE	40	60	100	4	-	-	4
	19MCA5PE04	Computer Forensics								
	19MCA5PE05	Cyber Security								
	19MCA5PE06	E-Commerce								
5	Professional Elective – IV		PE	40	60	100	4	-	-	4
	19MCA5PE07	Agile Methodology & Devops								
	19MCA5PE08	Embedded & Real Time Operating Systems								
	19MCA5PE09	Big Data Analytics								
6	19MCA5LB01	Software Testing Methodology Lab	PC	40	60	100	-	-	4	2
7	19MCA5LB02	Machine Learning using Python Lab	PC	40	60	100	-	-	4	2
8	19MCA5LB03	Frameworks & Devops Lab	PC	40	60	100	-	-	4	2
TOTAL				320	480	800	20	-	12	26

III MCA V SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5TH01	4	-	-	40	60	100	4
SOFTWARE TESTING METHODOLOGIES							

COURSE OBJECTIVES:

This course will enable students

1. To study fundamental concepts in software testing strategies and methods.
2. To gain the techniques and skills on how to use modern software testing tools to support Software testing projects.
3. Understand and use transaction flow testing techniques, data flow testing & domain testing.
4. To gain an understanding of Decision tables, kv charts, path expressions and state graphs.

COURSE OUTCOMES:

After completion of this course, the student should be able to

CO 1: Apply software testing knowledge and engineering methods.

CO 2: Design and conduct a software test process for a software testing project.

CO 3: Understand and identify various software testing problems.

CO 4: Solve the software testing problems by designing and selecting software test models, criteria, strategies and methods.

UNIT I

Introduction: Purpose of testing, Dichotomies, Model for testing, Consequences of bugs, Taxonomy of bugs.

UNIT II

Flow graphs and Path testing: Basic concepts of path testing, Predicates, Path predicates and achievable paths, Path sensitizing, Path instrumentation, Application of path testing.

UNIT III

Transaction Flow Testing: Transaction flows, Transaction flow testing techniques.

Dataflow testing: Basics of dataflow testing, Strategies in dataflow testing, Applications of dataflow testing.

UNIT IV

Domain Testing: Domains and paths, Domain testing, Domain and interface testing, Domains and testability.

UNIT V

Logic Based Testing: Overview, Decision tables, Path expressions, kv charts, Specifications.

UNIT VI

State, State Graphs and Transition testing: State graphs, good & bad state graphs, State testing, Testability tips.

TEXT BOOKS

1. Software testing techniques - Boris Beizer, International Thomson computer press, second edition.
2. Software Testing- Yogesh Singh, CAMBRIDGE.

REFERENCE BOOKS

1. Introduction to Software Testing, Paul Amman, Jeff Offutt, CAMBRIDGE.
2. Effective Software testing, 50 Specific ways to improve your testing, Elfriede Dustin,PEA.

WEB REFERENCES:

1. <http://www.nptel.ac.in/topic/testing-methodology>.
2. <http://guru99.com/testing-methodology.html>

II MCA IV SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5TH02	4	-	-	40	60	100	4
ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING							

COURSE OBJECTIVES:

The objectives of this course are

1. Develop a basic understanding of the building blocks of AI as presented in terms of intelligent agents: Search, Knowledge representation, inference, logic and learning.
2. Students will implement a small AI system in a team environment.
3. The knowledge of artificial intelligence plays a considerable role in some applications students develop for courses in the program.
4. Understand the different learning system methods.

COURSE OUTCOMES:

At the end of this course, the student should be able to

CO 1: Acquire ability to explain and apply key ideas in artificial intelligence and machine learning.

CO 2: Gain knowledge of related fields such as natural language processing, text mining.

CO 3: Acquire ability to design intelligent solutions to problems in a variety of domains and business applications.

CO 4: Acquire the knowledge of related fields such as robotics, reasoning and problem solving.

UNIT I

Introduction to AI, Problems, Problem Spaces and Search: Defining the Problem as a State space Search, Production Systems, Problem Characteristics, Production system characteristics, Issues in the Design of Search Programs.

UNIT II

Heuristic Search Techniques: Generate-and-test, Hill Climbing, Best-First Search, Problem Reduction, Constraint Satisfaction, Means-Ends Analysis.

UNIT III

Knowledge Representation Using Predicate Logic: Representing Simple Facts in logic, Representing Instance and Isa Relationships, Computable Functions and Predicates, Resolution Representing.

Knowledge Using Rules: Procedural versus Declarative Knowledge, Logic Programming, Forward versus Backward Reasoning, Matching, Control Knowledge. Weak slot and-filler structures: Semantic Nets, Frames

UNIT IV

Machine learning Introduction: Well -posed Learning problems. Designing a Learning System: Choosing the Training Experience, Choosing the Target Function, Choosing a Representation for the Target function, choosing a Function Approximation Algorithm, The final Design. Perspective and Issues in Machine Learning: Issues in Machine Learning.

UNIT V

Decision Tree Learning: Introduction, Decision Tree Representation, Appropriate Problems for Decision Tree Learning. The Basic Decision Tree Learning Algorithm: Which attribute is the Best classifier, an illustrative example, Hypothesis Space Search in Decision Tree Learning Inductive Bias in Decision Tree.

UNIT VI

Artificial Neural Networks: Introduction Neural Network Representations, Appropriate Problems for Neural Network Learning, Perceptrons, Multi-Layer Networks and BACK PROPAGATION Algorithm, Remarks on the BACK PROPAGATION Algorithm, An Illustrative Example: Advanced Topics in Artificial Neural Networks

TEXT BOOKS

1. Elaine Rich & Kevin Knight, 'Artificial Intelligence', 3rd Edition, Tata McGraw Hill Edition, Reprint (2008) (Unit 1,2,3).
2. Tom M. Mitchell "Machine Learning", McGraw Hill, 1997 (Unit 4,5,6).

REFERENCE BOOKS

1. Russel and Norvig, 'Artificial Intelligence', Pearson Education, PHI, (2003).
2. Ethem Alpaydin, "Introduction to Machine Learning", The MIT Press, (2010).

WEB REFERENCES

1. <https://www.coursera.org/learn/machine-learning>
2. <https://www.simplilearn.com/big-data-and-analytics/machine-learning>
3. <https://www.applidaicourse.com/course/applied-ai-course-online>
4. <http://nptel.ac.in/courses/106105152>

III MCA V SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5TH03	4	-	-	40	60	100	4
CLOUD COMPUTING							

COURSE OBJECTIVES:

This course will enable students

1. To gain knowledge about virtualization and Virtual Machines.
2. To familiarize Cloud Computing and its services.
3. Understand various basic concepts related to cloud computing technologies.
4. Understand different management applications.

COURSE OUTCOMES:

After Completion of this course the student should be able to

CO 1: Understand the Virtualization and applications for the state-of-the-art cloud computing.

CO 2: Carry out the Cloud Scale and value of Cloud Computing.

CO 3: Analyze the infrastructure of cloud computing including public, private and hybrid clouds and various services like PaaS, SaaS, IaaS etc.

CO 4: Monitor the Security in cloud.

UNIT I

Foundations: Introduction to Cloud Computing, Migrating into a Cloud, Enriching the 'Integration as a Service' Paradigm for the Cloud Era. The Enterprise Cloud computing Paradigm.

UNIT II

Infrastructure as a Service (IAAS): Virtual machines provisioning and Migration services, On the Management of Virtual machines for Cloud Infrastructures, Enhancing Cloud Computing Environments using a cluster as a Service. Secure Distributed Data Storage in Cloud Computing.

UNIT III

Platform and Software as a Service (PAAS / SAAS):

Aneka, Comet Cloud, T-Systems, Workflow Engine for Clouds. Understanding Scientific Applications for Cloud Environments.

UNIT IV

Monitoring, Management and Applications: An Architecture for Federated Cloud Computing, SLA Management in Cloud Computing.

UNIT V

Performance Production for HPC on Clouds, Best Practices in Architecture Cloud Applications in the AWS cloud, Building Content Delivery networks Clouds, Resource Cloud Mashups.

UNIT VI

Governance and Case Studies: Organizational Readiness and Change management in the Cloud age. Data Security in the Cloud, Legal issues in Cloud computing. Achieving Production Readiness for Cloud Services.

TEXT BOOKS

1. Cloud Computing: Principles and Paradigms by Rajkumar Buyya, James Broberg.
2. Distributed and Cloud Computing. Kal Hwang. Geoffeiy C.Fox. Jack J.Dongarra. Esevier. 2012.

REFERENCE BOOKS

1. Cloud Computing: A Practical Approach. Anthony T.Velte. Toby J.VeFte, Robert Elsenpeter. Tata McGraw Hill. rp2011.
2. Enterprise Cloud Computing Gautam Shroif, Cambridge University Press. 2010.
3. Cloud Computing: Implementation, Management and Security, John W. Rittinouse, James Ransome. CRC Press, rp2012.
4. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud. George Reese, O'RedI SPD, rp2011.
5. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Ktriaraswamy, Shahed Latif, O'Redç SPD, rp2011.

WEB REFERENCES:

1. <http://nptel.ac.in/courses/106106129/21>
2. <https://freevideolectures.com/course/3649/cloud-computing>
3. https://www.youtube.com/watch?v=Eg4AAGCE7X4&list=PL2UlrhJ_JwyA5IIOCdEWI NArFke4jgtlg.

III MCA V SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5PE04	-	-	4	40	60	100	4
COMPUTER FORENSICS							

COURSE OBJECTIVES:

This course is intended to provide students with greater depth of study in a number of key topics in the area of computer security in society.

- 1.cybercrime.
- 2.computer and forensics, analysis.

COURSE OUTCOMES:

- CO 1:** Understand accounting forensics, and explain their role in preventing various forms of Fraud.
- CO 2:** Understand financial forensics, and explain their role in preventing various forms of Fraud.
- CO 3:** Distinguish various types of computer crime, and use computer forensic techniques.
- CO 4:** Identify the digital fingerprints associated with criminal activities.

UNIT I:

Computer Forensics and Investigations: Understanding Computer Forensics, Preparing for Computer Investigations, Taking A Systematic Approach, Procedure for Corporate High-Tech Investigations, Understanding Data Recovery Workstations and Software.

Investor’s Office and Laboratory: Understanding Forensics Lab Certification Requirements, Determining the Physical Requirements for a Computer Forensics Lab, Selecting a Basic Forensic Workstation.

UNIT II:

Data Acquisition: Understanding Storage Formats for Digital Evidence, Determining the Best Acquisition Method, Contingency Planning for Image Acquisitions, Using Acquisition Tools, Validating Data Acquisition, Performing RAID Data Acquisition, Using Remote Network Acquisition Tools, Using Other Forensics Acquisition Tools.

UNIT III:

Processing Crime and Incident Scenes: Identifying Digital Evidence, Collecting the Evidence in Private-Sector Incident Scenes, Processing law Enforcement Crime Scenes, Preparing for a Search, Securing a Computer Incident or Crime Scene, Sizing Digital evidence at the Scene, Storing Digital evidence, obtaining a Digital Hash.

UNIT IV:

Current Computer Forensics Tools: Evaluating Computer Forensics Toll Needs, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software Computer Forensics Analysis and Validation: Determining What Data to Collect and Anlyze, Validating Forensic Data, Addressing Data-Hiding Techniques, Performing Remote Acquisition.

UNIT V:

Recovering Graphics and Network Forensics: Recognizing a Graphics File, Understanding Data Compression, Locating and Recovering Graphics Files, Understanding Copyright Issues with Graphics, Network Forensic, Developing Standard Procedure for Network Forensics, Using Network Tools, Examining Honey Project.

UNIT VI:

E-mail Investigations Cell Phone and Mobile Device Forensics: Exploring the Role of E-mail in Investigations, Exploring the Role of Client and Server in E-mail, Investigating E-mail Crimes and Violations, Understanding E-mail Servers, Using Specialized E-mail Forensics Tools, Understanding Mobile Device Forensics, Understanding Acquisition Procedure for Cell Phones and Mobile Devices

TEXT BOOK:

1. Nelson, Phillips Einfinger, Steuart, “ Computer Forensics and Investigations, Cengage Learning.

III MCA V SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5PE05	4	-	-	40	60	100	4
CYBER SECURITY							

COURSE OBJECTIVES:

This course will enable students

1. To gain a fundamental knowledge of what Cyber Security is and how it applies to your daily work.
2. To gain an understanding of terms commonly used in Cyber Security such as “vulnerability”.
3. To know how vulnerabilities occur and how to limit your exposure to them.
4. To gain a fundamental understanding of what an attack is and how to identify and prevent them from occurring.
5. To provide the fundamental skills and understanding needed to identify Cyber Security threats.

COURSE OUTCOMES:

At the end of this course, the student should be able to

- CO 1:** Possess a fundamental knowledge of Cyber Security.
- CO 2:** Understand what vulnerability is and how to address most common vulnerabilities.
- CO 3:** Know basic and fundamental Computer Forensics principles as it relates to Cyber Security.
- CO 4:** Have the knowledge needed to practice safer computing and safeguard your information.
- CO 5:** Understand basic technical controls in use today, such as firewalls and Intrusion Detection systems.

UNIT I

Understanding Computer Network Security: Securing the Computer Network Forms of Protection, Security Standards.

Security Threats to Computer Networks: Sources of Security Threats, Security Threat Motives, Security Threat Management, Security Threat Correlation, Security Threat Awareness.

Computer Network Vulnerabilities: Sources of Vulnerabilities, Vulnerability Assessment.

Cyber Crimes and Hackers: Cyber Crimes, Hackers, Dealing with the Rising Tide of Cyber Crimes.

UNIT II

Security Assessment, Analysis and Assurance: System Security Policy, Building a Security Policy, Security Requirements Specification, Threat Identification, Threat Analysis, Vulnerability Identification and Assessment, Security Certification, Security Monitoring and Auditing, Products and Services.

Access Control and Authorization: Access Rights, Access Control Systems, Authorization, Types of Authorization Systems, Authorization Principles, Authorization Granularity, Web Access and Authorization.

Authentication: Multiple Factors and Effectiveness of Authentication, Authentication Elements, Types of Authentication, Authentication Methods, Developing an Authentication Policy.

UNIT III

Firewalls: Types of Firewalls, Configuration and Implementation of a Firewall, The Demilitarized Zone (DMZ), Improving Security Through the Firewall, Firewall Forensics, Firewall Services and Limitations.

System Intrusion Detection and Prevention: Intrusion Detection, Intrusion Detection Systems (IDSs), Types of Intrusion Detection Systems, The Changing Nature of IDS Tools, Other Types of Intrusion Detection Systems, Response to System Intrusion, Challenges to Intrusion Detection Systems, Implementing an Intrusion Detection System, Intrusion Prevention Systems (IPSs), Intrusion Detection Tools.

The Art of Intrusion Detection: Basic Ideas of Intrusion Detection, Network-Based Detections and Host-Based Detections, Signature Detections, Statistical Analysis, Behavioral Data Forensics, Honeypots.

UNIT IV

The Art of Anti Malicious Software: Viruses, Worms, Virus Defense, Trojan Horses, Hoaxes, Peer-to-Peer Security, Web Security, Distributed Denial of Service Attacks.

Computer and Network Forensics: Computer Forensics, Network Forensics, Forensics Tools.

Virus and Content Filtering: Scanning, Filtering, and Blocking, Virus Filtering, Content Filtering, Spam.

UNIT V

Security in Wireless Networks and Devices: Cellular Wireless Communication Network Infrastructure, Wireless LAN (WLAN) or Wireless Fidelity (Wi-Fi), Standards for Wireless Networks, Security in Wireless Networks.

Security in Sensor Networks: The Growth of Sensor Networks, Design Factors in Sensor Networks, Security in Sensor Networks, Security Mechanisms and Best Practices for Sensor, Trends in Sensor Network Security Research.

UNIT VI

Security beyond Computer Networks: Information Assurance: Collective Security Initiatives and Best Practices.

Network Perimeter Security: General Framework, Packet Filters, circuit Gateways, Application Gateways, Trusted Systems and Bastion Hosts, Firewall Configurations, Network Address Translations, Setting up Firewalls.

TEXT BOOKS

1. Computer Network Security- Theory and Practice by Jie Wang, 2009 edition, Higher Education Press, Beijing and Springer-Verlag.
2. A Guide to Computer Network Security by Joseph Migga Kizza, 2009 Edition, Springer-Verlag London Limited.

REFERENCE BOOKS

1. Network Security Essentials- Applications and Standards by William Stallings, 4th edition.
2. Modern Cryptography: Theory and Practice by Wenbo Mao Hewlett-Packard Company, 1st edition, Prentice Hall PTR.
3. Network Security: Private Communication in a Public World by Charlie Kaufman, Radia Perlman, Mike Speciner, 2nd edition.
4. Cryptography and Network Security: Principles and Practice by William Stallings, 6th Edition.

III MCA V SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5PE06	4	-	-	40	60	100	4
E-COMMERCE							

COURSE OBJECTIVES:

This course will enable students

1. Explain to students why information systems are so important today for business and management.
2. Evaluate the role of the major types of information systems in a business environment and their relationship to each other.
3. Identify the major management challenges to building and using information systems and learn how to find appropriate solutions to those challenges.

COURSE OUTCOMES:

After completion of this course, the students should be able to

CO 1: Demonstrate an understanding of the foundations and importance of E-commerce.

CO 2: Demonstrate an understanding of retailing in E-commerce.

CO 3: Analyse the impact of E-commerce on business models and strategy.

CO 4: Describe Internet trading relationships including Business to Consumer, Business-to-Business & Intra-organizational.

UNIT I

Electronic Commerce, Frame work, anatomy of E-Commerce applications, E-Commerce Consumer applications, E-Commerce organization applications.

UNIT II

Consumer Oriented Electronic commerce, Mercantile Process models.

UNIT III

Electronic payment systems , Digital Token-Based, Smart Cards, Credit Cards, Risks in Electronic Payment systems.

UNIT IV

Intra Organizational Commerce: Work Flow, Automation Customization and internal Commerce, Supply chain Management.

UNIT V

Corporate Digital Library: Document Library, Digital Document types, Corporate Data Warehouses. Advertising and Marketing, Information based marketing, Advertising on Internet, on-line marketing process, market research.

UNIT VI

Consumer Search and Resource Discovery, Information search and Retrieval, Commerce Catalogues, Information Filtering. Multimedia - key multimedia concepts, Digital Video and electronic Commerce, Desktop video processing's, Desktop video conferencing.

TEXT BOOK

1. Frontiers of Electronic Commerce , Kolkata, Whinstone, PEA,2006.

REFERENCE BOOKS

1. E-Commerce Fundamentals and Applications Hendry Chan, Raymond Lee, Dillon, Chang, John Wiley.
2. E-Commerce, A Managerial Perspective, Turban E, Lee J , King, Chung H.M.,PEA,2001.
3. E-Commerce An Indian Perspective , 3/e, P.T. Joseph, PHI,2009.
4. E-Commerce, S.Jaiswal. , Golgothia.
5. Electronic Commerce , Gary Schneider, Thomson.

WEB REFERENCES:

1. <http://www.udemy.com/topic/e-commerce>.
2. <http://www.nptel.ac.in/topic/e-commerce>.

III MCA V SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5PE07	4	-	-	40	60	100	4
AGILE METHODOLOGIES & DEVOPS							

COURSE OBJECTIVES:

DevOps improves collaboration and productivity by automating infrastructure and workflows and continuously measuring applications performance.

COURE OUTCOMES: At the end of the course, student will be able to

1. Understand the principles of continuous development and deployment, automation of configuration management, inter-team collaboration, and IT service agility.
2. Describe DevOps & DevSecOps methodologies and their key concepts.
3. Explain the types of version control systems, continuous integration tools, continuous monitoring tools, and cloud models.
4. Set up complete private infrastructure using version control systems and CI/CD tools.

UNIT I:

Phases of Software Development life cycle. Values and principles of agile software development.

UNIT II:

Introducing devops & Continuous delivery:

Introducing devops, Agile wheel of wheel, Devops & ITIL

UNIT III:

Fundamentals of DevOps: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system.

UNIT IV:

DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes.

UNIT V:

CI/CD: Introduction to Continuous Integration, Continuous Delivery and Deployment, Benefits of CI/CD, Metrics to track CICD practices.

UNIT VI:

Devops Maturity Model: Key factors of DevOps maturity model, stages of Devops maturity model, DevOps maturity Assessment

TEXT BOOKS:

1. The DevOPS Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations by Gene Kim , John Willis , Patrick Debois , Jez Humb, O'Reilly publications
2. What is Devops? Infrastructure as code By in Mike Loukides ,O'Reilly publications.
3. Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation, by Jez Humble and David Farley
4. Achieving DevOps: A Novel About Delivering the Best of Agile, DevOps, and Microservices by Dave Harrison, Knox Lively
5. Practical Devops by Joakim Verona.

REFERENCE BOOKS:

1. Building a DevOps Culture by Mandi Walls, O'Reilly publications
2. The DevOps 2.0 Toolkit: Automating the Continuous Deployment Pipeline With Containerized Microservices by Viktor Farcic.

III MCA V SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5PE08	4	-	-	40	60	100	4
EMBEDDED AND REAL TIME OPERATING SYSTEMS							

COURSE OBJECTIVES:

This course will enable students

1. TO Develop an understanding of the technologies behind the embedded computing Systems.
2. Imparting technology capabilities and limitations of the hardware, software components.
3. Design methods to evaluate design tradeoffs between different technology choices and Also design the methodologies.

COURSE OUTCOMES:

At the end of this course, the student should be able to

CO 1: Understand the basics of an embedded system

CO 2: Program an embedded system

CO 3: Design, implement and test an embedded system.

CO 4: Identify the unique characteristics of real-time systems

CO 5: Explain the general structure of a real-time system

CO 6: Define the unique design problems and challenges of real-time systems

UNIT I:

Introduction to Embedded systems: What is an embedded system Vs. General computing system, history, classification, major application areas, and purpose of embedded systems. Core of embedded system, memory, sensors and actuators, communication interface, embedded firmware, other system components, PCB and passive components.

UNIT II:

8—bit microcontrollers architecture: Characteristics, quality attributes application specific, domain specific, embedded systems. Factors to be considered in selecting a controller, 8051 architecture, memory organization, registers, oscillator unit, ports, source current, sinking current, design examples.

UNIT III:

RTOS and Scheduling, Operating basics, types, RTOS, tasks, process and threads, multiprocessing and multitasking, types of multitasking, non preemptive, preemptive scheduling.

UNIT IV:

Task communication of RTOS, Shared memory, pipes, memory mapped objects, message passing, message queue, mailbox, signaling, RPC and sockets, task communication / synchronization issues, racing, deadlock, live lock, the dining philosopher's problem.

UNIT V:

The producer-consumer problem, Reader writers problem, Priority Inversion, Priority ceiling, Task Synchronization techniques, busy waiting, sleep and wakery, semaphore, mutex, critical section objects, events, device, device drivers, how to clause an RTOS, Integration and testing of embedded hardware and fire ware.

UNIT VI:

Simulators, emulators, Debuggers, Embedded Product Development life cycle (EDLC), Trends in embedded Industry, Introduction to ARM family of processor.

TEXT BOOK

1. Introduction to embedded systems Shibu. K.V, TMH, 2009.

REFERENCE BOOKS

1. Ayala & Gadre: The 8051 Microcontroller & Embedded Systems using Assembly and C, CENGAGE
2. Embedded Systems, Rajkamal, TMH, 2009.
3. Embedded Software Primer, David Simon, Pearson.
4. The 8051 Microcontroller and Embedded Systems, Mazidi, Mazidi, Pearson,.

III MCA V SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5PE09	4	-	-	40	60	100	4
BIG DATA ANALYTICS							

COURSE OBJECTIVES:

This course will enable students

1. Introducing Java concepts required for developing map reduce programs.
2. Optimize business decisions and create competitive advantage with Big Data analytics.
3. Derive business benefit from unstructured data.
4. Imparting the architectural concepts of Hadoop and introducing map reduce paradigm.
5. To introduce programming tools PIG & HIVE in Hadoop ecosystem.

COURSE OUTCOMES:

At the end of this course, the student should be able to

CO 1: Preparing for data summarization, query, and analysis.

CO 2: Applying data modelling techniques to large data sets.

CO 3: Creating applications for Big Data analytics.

CO 4: Building a complete business data analytic solution.

UNIT I

Data structures in Java: Linked List, Stacks, Queues, Sets, Maps; Generics: Generic classes and Type parameters, Implementing Generic Types, Generic Methods, Wrapper Classes, Concept of Serialization.

UNIT II

Working with Big Data: Google File System, Hadoop Distributed File System (HDFS) – Building blocks of Hadoop (Name node, Data node, Secondary Name node, Job Tracker, Task Tracker), Introducing and Configuring Hadoop cluster (Local, Pseudo-distributed mode, Fully Distributed mode), Configuring XML files.

UNIT III

Writing Map Reduce Programs: A Weather Dataset, Understanding Hadoop API for Map Reduce Framework (Old and New), Basic programs of Hadoop Map Reduce: Driver code, Mapper code, Reducer code, Record Reader, Combiner, Partitioner.

UNIT IV

Hadoop I/O: The Writable Interface, Writable Comparable and comparators, Writable Classes: Writable wrappers for Java primitives, Text, Bytes Writable, Null Writable, Object Writable and Generic Writable, Writable collections, Implementing a Custom Writable: Implementing a Raw Comparator for speed, Custom comparators.

UNIT V:

Pig: Hadoop Programming Made Easier

Admiring the Pig Architecture, Going with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Evaluating Local and Distributed Modes of Running Pig Scripts, Checking out the Pig Script Interfaces, Scripting with Pig Latin

UNIT VI:

Applying Structure to Hadoop Data with Hive: Saying Hello to Hive, Seeing How the Hive is Put Together, Getting Started with Apache Hive, Examining the Hive Clients, Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, Querying and Analyzing Data

TEXT BOOKS:

1. Big Java 4th Edition, Cay Horstmann, Wiley John Wiley & Sons, INC
2. Hadoop: The Definitive Guide by Tom White, 3rd Edition, O'reilly
3. Hadoop in Action by Chuck Lam, MANNING Publ.
4. Hadoop for Dummies by Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk, Bruce Brown, Rafael Coss

REFERENCE BOOKS:

1. Hadoop in Practice by Alex Holmes, MANNING Publ.
2. Hadoop MapReduce Cookbook, Srinath Perera, Thilina Gunarathne.

WEB REFERENCES:

1. Hadoop: <http://hadoop.apache.org/>
2. Hive: <https://cwiki.apache.org/confluence/display/Hive/Home>
3. Piglatin: <http://pig.apache.org/docs/r0.7.0/tutorial.html>

III MCA V SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5LB01	-	-	4	40	60	100	2
SOFTWARE TESTING METHODOLOGY LAB							

COURSE OBJECTIVES:

This course will enable students

1. Learn the importance of software testing.
2. Learn different testing methodologies.
3. Learn how to test functional & Non-functional requirements.
4. Learn the importance of modern testing tools.

COURSE OUTCOMES:

After completion of this course student must be able to

CO 1: Perform manual testing.

CO 2: Perform boundary value analysis.

CO 3: Apply path testing on any given program.

CO 4: Differentiate Block box, White box, Alpha & Beta testing.

Problem Statement 01

Consider an automated banking application. The user can dial the bank from a personal computer, provide a six-digit password, and follow with a series of keyword commands that activate the banking function. The software for the application accepts data in the following form:

Area Code	Blank or three-digit number
Prefix	Three-digit number, not beginning with 0 or 1
Suffix	Four-digit number
Password	Six-character alphanumeric
Commands	“Check status”, “Deposit”, “Withdrawal”

- (a) Design adhoc test cases to test the system.
- (b) Design the test cases to test the system using following Black Box testing technique:
BVA, Worst BVA, Robust BVA, Robust Worst BVA Equivalence class testing
(Input/ Output domain)

Problem Statement 02

Consider an application that is required to validate a number according to the following simple rules:

- A number can start with an optional sign.
- The optional sign can be followed by any number of digits.
- The digits can be optionally followed by a decimal point, represented by a period.

- If there is a decimal point, then there should be two digits after the decimal.
- Any number-whether or not it has a decimal point, should be terminated a blank.

Generate test cases to test valid and invalid numbers.

(HINT) Use Decision table and cause-effect graph to generate test cases.

Problem Statement 03

Generate test cases using Black box testing technique to Calculate Standard Deduction on Taxable Income. The standard deduction is higher for tax payers who are 65 or older or blind. Use the method given below to calculate tax.

1. The first factor that determines the standard deduction is the filing status.
The basic standard deduction for the various filing statuses are:
Single \$4,750
Married, filing a joint return \$9,500
Married, filing a separate return \$7,000
2. If a married couple is filing separate returns and one spouse is not taking standard Deduction, the other spouse also is not eligible for standard deduction.
3. An additional \$1,000 is allowed as standard deduction, if either the filer is 65 years or the spouse is 65 years or older
(the latter case applicable when the filing status is “Married” and filing “joint”).
4. An additional \$1,000 is allowed as standard deduction, if either the filer is blind or the spouse is blind (the latter case applicable when the filing status is “married” and filing “joint”).

(HINT): From the above description, it is clear that the calculation of standard deduction depends on the following 3 factors:

1. Status of filing of the filer
2. Age of the filer
3. Whether the filer is blind or not

In addition, in certain cases, the following additional factors also come into play in calculating the standard deduction.

1. Whether spouse has claimed standard deduction
2. Whether spouse is blind
3. Whether the spouse is more than 65 years old

Problem Statement 04

Consider the following program segment:

1. int max (int i, int j, int k)
2. {
3. int max;
4. if (i>j) then
5. if (i>k) then max=i;
6. else max=k;
7. else if (j > k) max=j
8. else max=k

9. return (max);
10. }

- a) Draw the control flow graph for this program segment
- b) Determine the cyclomatic complexity for this program
- c) Determine the independent paths

Problem Statement 05

Source code of simple insertion sort implementation using array in ascending order in c programming language

```
#include
int main(){
int i,j,s,temp,a[20];
Printf (“Enter total elements: “); Scanf (“%d”,&s);
printf(“Enter %d elements: “,s); for(i=0;i<s;i++) scanf(“%d”,&a[i]); for(i=1;i<s;i++){
temp=a[i]; j=i-1; while((temp<a[j])&&(j>=0)){ a[j+1]=a[j];
j=j-1;
}
a[j+1]=temp;
}
printf(“After sorting: “);
for(i=0;i<s;i++) printf(“ %d”,a[i]); return 0;
}
```

HINT: for loop is represented as while loop

- a) Draw the program graph for given program segment
- b) Determine the DD path graph
- c) Determine the independent paths
- d) Generate the test cases for each independent path

Problem Statement 06

Consider a system having an FSM for a stack having the following states and transitions:

States Initial: Before creation Empty: Number of elements = 0

Holding: Number of elements > 0, but less than the maximum capacity

Full: Number elements = maximum

Final: After destruction

Initial to Empty: Create

Empty to Holding, Empty to Full, Holding to Holding, Holding to Full: Add

Empty to Final, Full to Final, Holding to Final: Destroy

Holding to Empty, Full to Holding, Full to Empty: Delete

Design test cases for this FSM using state table-based testing.

Problem Statement 07

Given the following fragment of code, how many tests are required for 100% decision coverage? Give the test cases.

```
if width > length
then biggest_dimension = width
if height > width
then biggest_dimension = height
end_if
else if biggest_dimension = length
then if height > length
then biggest_dimension = height
end_if
end_if
```

Hint 04 test cases

Problem Statement 08

Given the following code, how much minimum number of test cases is required for full statement and branch coverage?

```
read p read q
if p+q > 100
then print "Large"
endif
if p > 50
then print "p Large"
endif
```

Hint 1 test for statement coverage, 2 for branch coverage

Problem Statement 09

Consider a program to input two numbers and print them in ascending order given below. Find all du paths and identify those du-paths that are not feasible. Also find all dc paths and generate the test cases for all paths (dc paths and non dc paths).

```
#include
#include
1. void main ()
2. {
3. int a, b, t;
4. clrscr ();
5. printf ("Enter first number");
6. scanf ("%d",&a);
7. printf ("Enter second number");
8. scanf ("%d",&b);
9. if (a<b){
10. t=a;
11. a=b;
12. b=t;
13. }
14. printf ("%d %d", a, b);
15. getch ();
}
```

Problem Statement 10

Consider the code to arrange the nos. in ascending order. Generate the test cases for relational coverage, loop coverage and path testing. Check the adequacy of the test cases through mutation testing and also compute the mutation score for each.

```
i = 0;
n=4; //N-Number of nodes present in the graph
While (i<n-1) do j = i + 1;
While (j<n) do
if A[i]<A[j] then swap (A[i], A[j]); end do;
i=i+1;
end do
```

REFERENCE BOOKS

1. Software testing concepts & Tools, P. Nageswara Rao, Dreamtech press.
2. Software testing tools, Dr. K.V.K. Prasad, Dreamtech press.

II MCA IV SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5LB02	-	-	4	40	60	100	2
MACHINE LEARNING USING PYTHON LAB							

COURSE OBJECTIVES:

This course will enable students to

1. Make use of Data sets in implementing the machine learning algorithms
2. Implement the machine learning concepts and algorithms in any suitable language of choice.

SYLLABUS

1. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
3. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
4. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.
5. Write a program to implement the naive Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
6. Assuming a set of documents that need to be classified, use the naive Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
7. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
8. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using *k*-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
9. Write a program to implement *k*-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.

III MCA V SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
19MCA5LB03	-	-	4	40	60	100	2
FRAMEWORKS & DEVOPS LAB							

COURSE OBJECTIVE:

Learn Single Page web application development Framework using Angularjs.

SYLLABUS:

1. Create Scalable.
2. Easy to Use (User Friendly).
3. Platform/Device Independent Web Applications.

Create Sample Web applications to the following:

1. Angular Expressions
2. Angular Modules
3. Angular Directives
4. Angular Model
5. Angular Data Binding
6. Angular Controllers
7. Angular Scopes
8. Angular Filters
9. Angular Services
10. Angular Http
11. Angular Tables
12. Angular Select
13. Angular SQL
14. Angular DOM
15. Angular Events
16. Angular Forms
17. Angular Validation
18. Angular APIs
19. Angular W3.CSS
20. Angular Includes
21. Angular Animations
22. Angular Routing
23. Angular Application

Agile Planning and Portfolio Management with Azure Boards.

Exercise 1: Agile Project Management

- Task 1: Working with teams, areas, and iterations
- Task 2: Working with work items
- Task 3: Managing sprints and capacity
- Task 4: Customizing Kanban boards
- Task 5: Defining dashboards
- Task 6: Customizing team process

REFERENCE BOOKS:

1. Learning AngularJS by Ken Williamson O'Reilly Media Inc.,
2. AngularJS, co-written by S. Brad Green and Shyam Seshadri, O'Reilly.

WEB REFERENCES:

1. <https://www.w3schools.com/angular/>
2. <https://www.edx.org/course/html5-css-fundamentals-w3cx-html5-0x-0>