# ACADEMIC REGULATIONS,
# COURSE STRUCTURE & SYLLABUS

FOR

MASTER OF COMPUTER APPLICATIONS
Two Year PG Day-Time Program
(With effect from 2020 – 21)


SCHOOL OF INFORMATION TECHNOLOGY
JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
KUKATPALLY, HYDERABAD – 500 085, TS

SCHOOL OF INFORMATION TECHNOLOGY
JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
HYDERABAD – 500 085 - TELANGANA

# ACADEMIC REGULATIONS FOR THE AWARD OF M.C.A DEGREE BASED ON CHOICE BASED CREDIT SYSTEM (CBCS)

## (WITH EFFECT FROM THE ACADEMIC YEAR 2020 – 2021)

The Master of Computer Applications Postgraduate Degree of the Jawaharlal Nehru Technological University Hyderabad (JNTUH) shall be conferred on candidates who are admitted to the program and who fulfill all the requirements for the award of the Degree.

JNTUH offers 2 Years (4 Semesters) Master of Computer Applications (M.C.A) degree program, under Choice Based Credit System (CBCS) at its constituent Autonomous College – ***School of Information Technology (SIT), JNTUH***, Hyderabad.

## 1. ELIGIBILITY FOR ADMISSIONS

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

**1.1.** Admissions shall be made on the basis of merit rank obtained by the qualifying candidate in **ICET, a state wide MCA Entrance exam by Telangana State Government**, subject to reservations prescribed by the University from time to time.

## 2. AWARD OF M.C.A. DEGREE

**2.1** A candidate shall be declared eligible for the award of the M.C.A. Degree, if he pursues a course of study in not less than two and not more than four academic years.

**2.2** A candidate who fails to fulfill all the academic requirements for the award of the degree within four academic years from the year of his admission shall forfeit his seat in M.C.A. course.

## 3. M.C.A PROGRAM STRUCTURE

**3.1** The M.C.A Program of SIT-JNTUH is of Semester Pattern, with 4 Semesters constituting 2 Academic Years, each Academic Year having TWO Semesters (First/Odd and Second/Even Semesters). Each Semester shall be of 22 Weeks duration (inclusive of Examinations).

**3.2** UGC/ AICTE specified Definitions/ Descriptions are adopted appropriately for various terms and abbreviations used in these Academic Regulations/ Norms, which are as listed below.

**3.3 Semester Scheme:** The program is of 2 Academic Years (4 Semesters), with the year being divided into two Semesters of 22 weeks ( ≥ 90 working days) each, each Semester having - 'Continuous Internal Evaluation (CIE)' and 'Semester End Examination (SEE)'. Choice Based Credit System (CBCS) and Credit Based Semester System (CBSS) as denoted by UGC, and Curriculum/ Course Structure as suggested by AICTE are followed.

**3.4 Credit Courses:** All Subjects/ Courses are to be registered by candidates in a Semester to earn Credits. Credits shall be assigned to each Subject/ Course in a L: P: C (Lecture Periods: Practical Periods: Credits) structure, based on the following general pattern.

- One Credit - for One hour/ Week for Theory/ Lecture (L) Courses
- One Credit - for Two hours/ Week for Laboratory/ Practical (P) Courses

## 4. COURSE WORK

**4.1** A candidate after securing admission shall pursue the M.C.A. in a minimum period of 2 Academic Years, and a maximum period of 4 Academic Years (starting from the Date of Commencement of I Year).

**4.2** Each candidate shall register for and secure the specified number of Credits required for the completion and award of the M.C.A. Degree in respective specialization.

**4.3** Each Semester is structured to provide 21 Credits, except final semester which is of 26 credits, totaling to 89 Credits (89 C) for the entire M.C.A. Program.

**4.4 The candidates shall register for all 89 credits and secure all the 89 credits.**

## 5. COURSE REGISTRATION

**5.1** A 'Course Coordinator or Faculty Advisor' shall be assigned to each candidate, who will advise him about them M.C.A. program, its Course Structure and Curriculum,

Choice/Option for Subjects/ Courses, based on his competence, progress, pre-requisites and interest.

**5.2** Academic Section of the College invites 'Registration Forms' from candidates' priory (before beginning of the III and IV Semester). The Registration Requests should be completed BEFORE the commencement of SEEs (Semester End Examinations) of the 'PRECEDING SEMESTER'.

**5.3** A candidate can register, ONLY AFTER obtaining the 'WRITTEN APPROVAL' from his Course Coordinator, which should be submitted to the College Academic Section.

**5.4** A candidate may be permitted to register for his Subjects/ Course with a typical total of 21 Credits per Semester (I - III Semesters): **Minimum being 18 Credits and Maximum being 24 Credits for III Semesters**, based on his PROGRESS and SGPA/ CGPA, and completion of the 'PRE-REQUISITES' as indicated for various Subjects/Courses, in the Department Course Structure and Syllabus contents. A candidate must register all the CORE subjects/courses.

**5.5** The candidate has to register for the audit courses and pass the audit courses for successful completion of the degree. However the credits earned in the audit courses are not included in the computation of SGPA/ CGPA.

**5.6** Choice for 'additional Subjects/ Courses' to reach the Maximum Permissible Limit of 24 Credits (above the typical 21 Credit norm) must be clearly indicated, which needs the specific approval and signature of the Course Coordinator.

**5.7** If the candidate submits ambiguous choices or multiple options or erroneous entries - during Registration for the Subject(s) / Course(s) under a given/ specified Course Group/ Category as listed in the Course Structure, only the first mentioned Subject/ Course in that Category will be taken into consideration.

**5.8** Subject/ Course Options exercised are final and CAN NOT be changed. However, if the Subject/ Course that has already been listed for Registration (by the Course Coordinator) in a Semester could not be offered due to any unforeseen or unexpected reasons, then the Candidates shall be allowed to have alternate choice - either for a new Subject (subject to offering of such a Subject), or for another existing Subject (subject to availability of seats), which may be considered. Such alternate arrangements will be made by the Course Coordinator, with due notification and time framed schedule, within the FIRST WEEK from the commencement of Class-work for that Semester.

**5.9** Dropping of Subjects/ Courses may be permitted, ONLY AFTER obtaining prior approval from the Faculty Advisor (subject to retaining a minimum of 18C), 'within 15 Days of Time' from the beginning of the current Semester.

**5.10 Candidates may register for NPTEL/SWAYAM as per the university rules.**

## 6. SUBJECTS/ COURSES TO BE OFFERED

**6.1** A typical Section (or Class) Strength for each Semester shall be 30.

**6.2** A Subject/ Course (ELECTIVE) may be offered to the Candidates, ONLY if a Minimum of $1/3^{rd}$ of the Section Strength opts for the same.

**6.3** More than ONE TEACHER may offer the SAME SUBJECT (Lab. / Practical's may be included with the corresponding Theory Subject in the same Semester) in any Semester. However, selection choice for candidates will be based on - 'FIRST COME FIRST SERVE Basis and CGPA Criterion'.

**6.4** If more entries for Registration of a Subject come into picture, then the concerned Course Coordinator shall take necessary action, whether to offer such a Subject/ Course for TWO (or multiple) SECTIONS or NOT .

**6.5** In case of options coming from Candidates of other Departments/ Branches/ Disciplines, in case of OPEN ELECTIVES, PRIORITY shall be given to the candidates of the 'Parent Department' first.

## 7. ATTENDANCE REQUIREMENTS

**7.1** The candidate shall put in a minimum of 75% attendance per semester independently for each of the course/subject registered.

**7.2** Condonation of shortage of attendance up to 10% in each course/subject registered (65% and above and less than 75%) may be given by the College/school Academic Committee.

**7.3**. Shortage of Attendance below 65% shall not be condoned. Condonation of shortage of attendance shall be granted only on genuine and valid reasons on representation by the candidate with supporting evidence and by paying stipulated condonation fee.

**7.4** Candidates whose shortage of attendance is not condoned in any course/subject registered are not eligible to write their end semester examination of that course/subject, they get **DETAINED** in that course/subject. The candidate will have to repeat that course/subject as and when offered; in case if there are any Professional Electives and/or Open Electives,

the same may also be re-registered if offered, however, if those electives are not offered in later semesters, then alternate electives may be chosen from the SAME set of ELECTIVE subjects offered under that category. In such a case, candidate has to pay tuition fee for that course/subject.

**7.5** A Candidate shall put in a minimum required attendance in at least three (3) theory subjects in each semester for promoting to next Semester. In order to qualify for the award of the MCA Degree, the candidate shall complete all the academic requirements of the subjects, as per the course structure.

## 8. EVALUATION

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

**8.1** Evaluation of candidates in a course/subject involves both external and internal components. External evaluation will be in the form of end semester examination in a course/subject for which is allocated 60% of the marks. The remaining 40% of marks are allocated to internal evaluation.

**8.2** The internal evaluation has two Mid Term-Examinations (each of 40 marks), one conducted in the middle of the Semester and the other immediately after the completion of instructions. Each midterm examination shall be conducted for a total duration of 120 minutes. The best one will be considered.

**8.3** The End semester Examination will be conducted in each subject for 60 marks. The question paper consists of 8 questions. Each of these questions may contain sub-questions. Each question carries 12 marks. The candidates will be required to answer 5 questions. The questions are to be prepared to cover the entire range of prescribed syllabi of the subjects and units.

**8.4** For practical subjects, 60 marks shall be awarded based on the performance in the End Semester Examinations and 40 marks for internal evaluation shall be awarded based on day-to-day performance and the internal exam.

**8.5** A candidate shall be deemed to have secured the minimum academic requirement in a subject if he secures a minimum of 40% marks in the End Examination and a minimum aggregate of 50% of the total marks in the End semester examination and internal evaluation taken together.

**8.6** In case the candidate does not secure the minimum academic requirement in any subject

(as specified in 8.5) he has to reappear for the Semester End Examination in that subject.

**8.7** A candidate can re-register for the subjects if the internal marks secured by him/her are less than 50% and failed in that subject for maximum of two subjects and should register within four weeks of commencement of the class work. In such a case, the candidate must re-register for the subjects and secure the required minimum attendance. The candidate's attendance in the re-registered subject(s) shall be calculated to decide upon his eligibility for writing the Semester End Examination in those subjects. In the event of the candidates taking another chance, his Internal Evaluation (internal) marks and Semester End Examination marks obtained in the previous attempt stands cancelled.

**9. Evaluation of Project / Dissertation Work.**

Every candidate shall be required to submit the thesis or dissertation after taking up a topic approved by the School/College.

**9.1 Registration of Project Work:** A candidate is permitted to register for the project work after satisfying the attendance requirement of all the subjects (theory and practical subjects).A candidate has to choose the topic in the first **2** weeks of the IV Semester in consultation with the Internal/External guides. After 2 weeks candidate has to submit an abstract of work to be carried out to the Project Review Committee (PRC), which in turn allows the candidate to register for thesis work if it is satisfied with the abstract submitted by the candidate.

**9.2** A Project Review Committee (PRC) shall be constituted with Course Coordinator and two members from the school.

**9.3** Only after obtaining the approval of Project Review Committee (PRC), the candidates can initiate the Project work.

**9.4** If a candidate wishes to change his supervisor or topic of the project he can do so with the approval of a committee appointed by the Director, SIT. However, the committee shall examine whether the change of topic/supervisor leads to a major change of his initial plans of project proposal. If so, his date of registration for the project work starts from the date of change of Supervisor or topic as the case may be.

**9.5** The total duration of the project is for 22 weeks which is spread across 22 weeks in IV semester. The candidates can submit the project only after 22 weeks from the date of registration after the approval of PRC.

**9.6** At end of the IV semester,

a. A candidate has to present a project seminar for which, Internal marks are for 40 marks which will be awarded by an internal committee consists of the concerned guide, course coordinator and a faculty member appointed by the Director, SIT.

External marks are for 60 marks which will be awarded by an external examiner appointed by the Director, SIT. Evaluation of project by Internal Committee (for 40 marks) and by the external examiner (for 60 marks) will be done simultaneously.

b. A candidate shall submit the thesis/dissertation in a hard bound copy form. He will attend the viva-voce conducted by the (External Viva-Voce) an external examiner appointed by the Director SIT. The project is evaluated for 100 marks. There will be no internal marks.

**9.7** The candidate has to submit two hard copies and one soft copy of Thesis/Dissertation, certified in the prescribed format by the supervisor to the school.

**9.8** In case the candidate fails in viva-voce examination, based on the recommendation of the board the candidate has to retake the viva-voce examination after three months. If she/he fails in this viva-voce examination also, he will not be eligible for the award of the degree unless the candidate is asked to revise and resubmit.

## 10. Award of Degree and Class – The Grading System

**10.1** A candidates shall be declared eligible for the award of M.C.A. degree, if he pursues a course of study and completes it successfully in not less than two academic years and not more than four academic years.

**10.2** A candidates, who fails to fulfill all the academic requirements for the award of the degree within four academic years from the year of his admission, for any reason whatsoever, shall forfeit his seat in M.C.A. Course.

**10.3 A** candidates shall register and put up minimum academic requirement in all 89 credits and earn the 89 credits. Marks obtained in all 89 credits shall be considered for the calculation of Cumulative Grade Point Average (CGPA) and percentage of marks.

**10.4** Marks will be awarded to indicate the performance of each candidates in each Theory Subject, or Lab/Practical's, or Seminar, or Project etc., based on the % of marks obtained in CIE + SEE (Continuous Internal Evaluation + Semester End Examination, both taken together) and a corresponding Letter Grade shall be given.

**10.5** As a measure of the candidate's performance, a 10-point Absolute Grading System using the following Letter Grades (UGC Guidelines) and corresponding percentage of marks shall be followed…..

| **Marks Obtained** | **Grade** | **Description of Grade** | **Grade Points (GP)**<br><br>**Value Per Credit** |
|---|---|---|---|
| >=90 | O | Outstanding | 10 |

| | | | |
|---|---|---|---|
| >=80 and <90 | A+ | Excellent | 9 |
| >=70 and <80 | A | Very Good | 8 |
| >=60 and <70 | B+ | Good | 7 |
| >=55 and <60 | B | Average | 6 |
| >=50 and <55 | C | Pass | 5 |
| <50 | F | Fail | 0 |
| Not Appeared the Exam(s) | AB | Absent | 0 |

**10.6** A candidate obtaining F Grade in any Subject shall be considered 'failed' and is required to reappear as 'Supplementary Candidate' in the Semester End Examination (SEE), as and when offered. In such cases, his Internal Marks (IE Marks) in those Subjects will remain the same as those he obtained earlier.

**10.7** A candidate not appeared for examination then 'AB' Grade will be allocated in any Subject shall be considered 'failed' and will be required to reappear as 'Supplementary Candidate' in the Semester End Examination (SEE), as and when offered.

**10.8** A Letter Grade does not imply any specific Marks percentage and it will be the range of marks percentage.

**10.9** In general, a candidate shall not be permitted to repeat any Subject/ Course (s) only for the sake of 'Grade Improvement' or 'CGPA Improvement'.

**10.10** A candidate earns Grade Point (GP) in each Subject, on the basis of the Letter Grade obtained by him in that Subject. The corresponding 'Credit Points' (CP) is computed by multiplying the Grade Point with Credits for that particular Subject.

**Credit Points (CP) = Grade Point (GP) x Credits …. For a Subject**

**10.11** The Candidates passes the Subject only when he **gets GP ≥ 5 (C Grade or above).**

**10.12** **The** Grade Point Average (GPA) is calculated by dividing the Sum of Credit Points (ΣCP) secured from ALL Subjects registered in a Semester or for the Exam appeared (like supplementary), by the Total Number of Credits registered during that Semester or for the Exam appeared (like supplementary). GPA is rounded off to FOUR Decimal Places. GPA is thus computed as

$$\text{GPA} = \frac{\sum_1^n C_{i \times GP_i}}{\sum_1^n C_i}$$

where n is the number of subjects Registered in that semester / exam.

$C_i$ is Credits for the subjects.
$GP_i$ is the grade point obtained for the subject

where 'i' is the Subject indicator index (takes into account all Subjects in a Semester or for the Exam appeared), 'N' is the no. of Subjects 'REGISTERED' for the Semester or for the Exam appeared, $C_i$ is the no. of Credits allotted to the i[th] Subject, and $G_i$ represents the Grade Points (GP) corresponding to the Letter Grade awarded for that i[th] Subject.

**10.13** The Cumulative Grade Point Average (CGPA) is a measure of the overall cumulative performance of a candidates over all Subjects in all considered for registration. The CGPA is the ratio of the Total Credit Points secured by a candidate in ALL registered Courses in ALL Semesters, and the Total Number of Credits registered in ALL the Semesters. CGPA is rounded off to FOUR Decimal Places. CGPA is thus computed as per the formula

$$\text{CGPA} = \frac{\sum_1^m GPA_j \times TC_j}{\sum_{j=1}^m TC_j}$$

where m is the number of subjects registered in the course.

$TC_j$ the total number of credits for a j[th] subject.

$GPA_j$ is the Grade point of the j[th] subject.

**10.14** For Calculations listed in Item 10.6 – 10.10, performance in failed Subjects/ Courses (securing F Grade) will also be taken into account, and the Credits of such Subjects/ Courses will also be included in the multiplications and summations.

## 11. AWARD OF DEGREE AND CLASS

**11.1** A Candidate who registers for all the specified Subjects/ Courses as listed in the Course Structure, satisfies all the Course Requirements, and passes the examinations prescribed in the entire PG Programme (PGP), and secures the required number of 89 Credits (with CGPA ≥ 5.0), shall be declared to have 'QUALIFIED' for the award of the M.C.A.

**11.2 Award of Class**: After a candidate has satisfied the requirements prescribed for the completion of the programme and is eligible for the award of M. C.A. Degree, he shall be placed in one of the following three classes based on the CGPA:

| Class Awarded | CGPA |
|---|---|
| First Class with Distinction | ≥ 8.00 |
| First Class | 7.00 ≤ CGPA < 8.00 |
| Second Class | 5.00 ≤ CGPA < 7.00 |

**11.3** A candidate with final CGPA (at the end of the PGP) < 5.00 will not be eligible for the Award of Degree.

## 12. WITHHOLDING OF RESULTS

**12.1** If the candidate has not paid the dues, if any, to the University or if any case of indiscipline is pending against him, the result of the candidate will be withheld and he will not be allowed into the next semester. His degree will be withheld in such cases.

## 13. TRANSITORY REGULATIONS

**13.1** If any candidate is detained due to shortage of attendance in one or more subjects, they are eligible for re-registration to same or equivalent subjects at a time as and when offered.

**13.2** In case any candidate makes a re-registration then the academic regulations which were applicable for the year of his joining year will be applicable.

## 14. GENERAL

**14.1 Credit**: A unit by which the course work is measured. It determines the number of hours of instructions required per week. One credit is equivalent to one hour of teaching (lecture or tutorial) or two hours of practical work/field work per week.

**14.2 Credit Point:** It is the product of grade point and number of credits for a course.

**14.3** Wherever the words "he", "him", "his", occur in the regulations, they include "she", "her".

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

**14.5** In the case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Vice-Chancellor is final.

**14.6** The University may change or amend the academic regulations or syllabi at any time and the changes or amendments made shall be applicable to all the candidates with effect from the dates notified by the University.

## 15. MALPRACTICES RULES

### 15.1 DISCIPLINARY ACTION FOR / IMPROPER CONDUCT IN EXAMINATIONS

| | **Nature of Malpractices/Improper conduct** | **Punishment** |
|---|---|---|
| | ***If the candidate:*** | |
| 1. (a) | Possesses or keeps accessible in examination hall, any paper, note book, programmable calculators, Cell phones, pager, palm computers or any other form of material concerned with or related to the subject of the examination (theory or practical) in which he is appearing but has not made use of | Expulsion from the examination hall and cancellation of the performance in that subject only. |

| | | |
|---|---|---|
| | (material shall include any marks on the body of the candidate which can be used as an aid in the subject of the examination) | |
| (b) | Gives assistance or guidance or receives it from any other candidate orally or by any other body language methods or communicates through cell phones with any candidate or persons in or outside the exam hall in respect of any matter. | 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 paper, book, programmable calculators, palm computers or any other form of material relevant to the subject of the examination (theory or practical) in which the candidate is appearing. | Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted to appear for the remaining examinations of the subjects of that Semester/year.<br><br>The Hall Ticket of the candidate is to be cancelled and sent to the University. |
| 3. | Impersonates any other candidate in connection with the examination. | The candidate who has impersonated shall be expelled from examination hall. The candidate is also debarred and forfeits 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 University 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. |
| 4. | Smuggles in the Answer book or additional sheet or takes out or arranges to send out the question paper during the examination or answer book or additional sheet, during or after the examination. | Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all University 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 candidates 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 University 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<br><br>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 candidates 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. | Candidates of the colleges expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat. Person(s) who do not belong to the College will be handed over to police and, a police case will be registered against |

| | | |
|---|---|---|
| | | them. |
| 10. | Comes in a drunken condition to the examination hall. | Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. |
| 11. | Copying detected on the basis of internal evidence, such as, during valuation or during special scrutiny. | Cancellation of the performance in that subject and all other subjects the candidate has appeared including practical examinations and project work of that semester/year examinations. |
| 12. | If any malpractice is detected which is not covered in the above clauses 1 to 11 shall be reported to the University for further action to award suitable punishment. | |

**15.2 Malpractices identified by squad or special invigilators**

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

**JNTUH SCHOOL OF INFORMATION TECHNOLOGY**
(Autonomous)
**MASTER OF COMPUTER APPLICATIONS**

**I YEAR I SEMESTER**

| S. No. | Subject Code | Subject | L | T | P | Max Marks(100) | | Credits |
|---|---|---|---|---|---|---|---|---|
| | | | | | | Int | Ext | |
| 1. | MC110 | Mathematics for Computer Science | 3 | - | - | 40 | 60 | 3 |
| 2. | MC120 | Computer Architecture & Organization | 3 | - | - | 40 | 60 | 3 |
| 3. | MC130 | Object Oriented programming through C++ | 3 | - | - | 40 | 60 | 3 |
| 4. | MC140 | Operating Systems | 3 | - | - | 40 | 60 | 3 |
| 5 | MC150 | Database Systems | 3 | - | - | 40 | 60 | 3 |
| 6. | MC160 | Computer Architecture & Organization & Operating Systems Lab | - | - | 4 | 40 | 60 | 2 |
| 7. | MC170 | Object Oriented programming through C++ Lab | - | - | 4 | 40 | 60 | 2 |
| 8. | MC180 | Database Systems lab | - | - | 4 | 40 | 60 | 2 |
| 9. | | **Audit-1** | 2 | - | - | 40 | 60 | 0 |
| | MC19A | Professional Communication Skills | | | | | | |
| | MC19B | Personality Development through Life Enlightenment Skills | | | | | | |
| | MC19C | Value Education | | | | | | |
| | MC19D | Constitution of India | | | | | | |
| | | **Total** | **15** | | **12** | **-** | **-** | **21** |

**I Year –II Semester**

| S. No. | Subject Code | Subject | L | T | P | Max Marks(100) | | Credits |
|---|---|---|---|---|---|---|---|---|
| | | | | | | Int | Ext | |
| 1. | MC210 | Python programming | 3 | - | - | 40 | 60 | 3 |
| 2. | MC220 | Data Structures through java | 3 | - | - | 40 | 60 | 3 |
| 3. | MC230 | Design and Analysis of Algorithms | 3 | - | - | 40 | 60 | 3 |
| 4. | MC240 | Computer Networks | 3 | - | - | 40 | 60 | 3 |
| 5 | MC250 | Accounting & Financial Management | 3 | - | | 40 | 60 | 3 |
| 6. | MC260 | Python programming Lab | - | - | 4 | 40 | 60 | 2 |
| 7. | MC270 | Data Structures through java Lab. | - | - | 4 | 40 | 60 | 2 |
| 8 | MC280 | Computer Networks Lab. | - | - | 4 | 40 | 60 | 2 |
| 9. | | **Audit-2** | 2 | - | - | 40 | 60 | 0 |
| | MC29A | Disaster Management | | | | | | |
| | MC29B | Soft Skills | | | | | | |
| | MC29C | Stress Management by yoga | | | | | | |
| | MC29D | Sanskrit for Technical Knowledge | | | | | | |
| | | **Total** | **15** | | **12** | **-** | **-** | **21** |

**II Year –I Semester**

| S. No. | Subject Code | Subject | L | T | P | Max Marks(100) | | Credits |
|---|---|---|---|---|---|---|---|---|
| | | | | | | Int | Ext | |
| 1. | MC310 | Internet Technologies | 3 | - | - | 40 | 60 | 3 |
| 2. | MC320 | Data Warehousing & Data Mining | 3 | - | - | 40 | 60 | 3 |
| 3. | MC330 | Software Engineering | 3 | - | - | 40 | 60 | 3 |
| 4. | | **Elective-1** | 3 | - | - | 40 | 60 | 3 |
| | MC341 | Embedded Systems | | | | | | |
| | MC342 | Artificial Intelligence | | | | | | |
| | MC343 | Distributed Databases | | | | | | |
| | MC344 | E-commerce | | | | | | |
| 5 | | **Elective-2** | 3 | - | | 40 | 60 | 3 |
| | MC351 | Android Application Development | | | | | | |
| | MC352 | Information Extraction and Information Retrieval Systems | | | | | | |
| | MC353 | Information Security | | | | | | |
| | MC354 | Mobile Computing | | | | | | |
| 6. | MC360 | Internet Technologies Lab. | - | - | 4 | 40 | 60 | 2 |
| 7. | MC370 | Data Warehousing and Data Mining Lab | - | - | 4 | 40 | 60 | 2 |
| 8. | MC380 | Software Engineering Lab | - | - | 4 | 40 | 60 | 2 |
| | | **Total** | **15** | | **12** | **-** | **-** | **21** |

**II-Year –II Semester**

| S. No. | Subject Code | Subject | L | T | P | Max Marks(100) | | Credits |
|---|---|---|---|---|---|---|---|---|
| | | | | | | Int | Ext | |
| 1. | | **Elective-3** | 3 | - | - | 40 | 60 | 3 |
| | MC411 | Internet of Things | | | | | | |
| | MC412 | Software Testing & Quality Assurance | | | | | | |
| | MC413 | Network Programming | | | | | | |
| | MC414 | Data Science | | | | | | |
| 2. | | **Open Elective -1** | 3 | - | - | 40 | 60 | 3 |
| | MC421 | Machine Learning | | | | | | |
| | MC422 | Software Project Management | | | | | | |
| | MC423 | Cloud Computing | | | | | | |
| | MC424 | Computer Forensics | | | | | | |
| | MC425 | Organization Structure and Personnel Management | | | | | | |
| | MC426 | Operations Research | | | | | | |
| 3. | MC430 | Project Seminar | - | - | - | 40 | 60 | 4 |
| 4. | MC440 | Project (viva-voce) | - | - | 32 | - | 100 | 16 |
| | | **Total** | 6 | - | 32 | - | - | 26 |

# MASTER OF COMPUTER APPLICATIONS

**I Year I Semester**

**MC110 MATHEMATICS FOR COMPUTER SCIENCE**

**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, proof of contradiction, automatic theorem Proving.

Predicate calculus: Predicative logic, Free and Bound variables, The Universe of Discourse. Inference theory of predicate calculus involving quantifiers.

**UNIT II: Relations& functions, Recurrence relation**

Properties of binary Relations, Relation matrix and graph of a relation, partition and covering of a set, equivalence relation, compatibility relations, composition of binary relations, partial ordering,

Partially ordered set: Lattices, Hesse diagram. Functions: Composition of functions, Inverse Function, Hashing functions.

Recurrence Relations: Generating Functions, Function of Sequences, Calculating Coefficients of generating functions, Recurrence relations, Solving recurrence relation by substitution and Generating functions, the method of Characteristic roots, solution of Inhomogeneous Recurrence Relations.

**UNIT III : Elementary Combinatorics and Probability theory**

Principle of counting, Combinations & Permutations, with repetitions, Constrained repetitions, Probability: Sample space and events – Probability – The axioms of probability - Some elementary theorems - Conditional probability – Bayes' theorem.

**UNIT IV: Random variables and Probability distributions**

Random variables – Discrete and continuous –probability Distribution – Distribution function, density function , Statistical constants of Probability distributions. Distributions - Binomial, Poisson and normal distributions, their related properties.

**UNIT V: Curve fitting, correlation &regression**

Curve fitting: The method of least squares, linear fit, non- linear fit.

Correlation, regression, fitting of regression lines Properties of correlation coefficients and regression coefficients , Rank correlation .

TEXTBOOKS:

1) Discrete Mathematics with Applications to Computer Science, J P Trembley and R Manohar, TMH, rp 2008. (Units I and II)
2) Discrete Mathematics for Computer Scientists and Mathematicians, second edition, J.L.Mott, A. Kandel, T.P. Baker, PHI(Unit III, IV)
3) Probability and statistics for engineers (Erwin Miller and John E. Freund), R A Johnson and C.B.Gupta.7th edition, PHI.

4) Introduction to Probability and Statistics, 12th edition, W.Mendenhall, R.J.Beaver and B.M.Beaver, Cengage Learning.

REFERENCE BOOKS:

1) Elements of Discrete Mathematics- A Computer Oriented Approach,C.L.Liu, D.P. Mohapatra,3rd edition,TMH.
2) Discrete and Combinatorial Mathematics- An Applied Introduction-5th Edition Ralph. P.Grimaldi, Pearson Education..
3) Discrete Mathematics, S K Chakraborthy and B K Sarkar, Oxford, 2011.
4) Probability and Statistics in Engineering, 4th Edition, William W.Hines, Douglas C.Montgomery, David M.Goldsman, Connie M.Borror, Wiley Student Edition.
5) Introduction to Probability and Statistics, J.S.Milton, Jesse C.Arnold, 4th edition, TMH.
6) Probability and Statistics for Engineers and Scientists,R.E.Walpole,S.L.Myers,K.Ye, Pearson.

**MASTER OF COMPUTER APPLICATIONS**

**I Year I Semester**

**MC120 COMPUTER ARCHITECTURE & ORGANIZATION**

**UNIT I**

NUMBER SYSTEMS AND COMPUTER ARITHMETIC- Signed and unsigned numbers, Addition and subtraction, multiplication, division, Floating point representation, logical operation, Gray code, BCD codes, Error detecting codes, Boolean algebra, Simplification of Boolean expressions, K-Maps,Combinational and Sequential Circuits- decoders, Encoders, Multiplexers, Half and Full adders, Shift registers, Sequential circuits- flip-flops.

**UNIT II**

MEMORY ORGANIZATION-Memory hierarchy, Main memory-RAM, ROM chips, Memory address map, memory contention to CPU, Associative Memory-Hardware logic, match, read and write logic, Cache Memory-Associative mapping, Direct mapping, Set-associative mapping, hit and miss ratio.

**UNIT III**

BASIC CPU ORGANIZATION-Introduction to CPU,Instruction formats-INTEL-8086 CPU architecture-Addressing modes - generation of physical address- code segment registers, Zero, one, two, and three address instructions.INTEL 8086 ASSEMBLY LANGUAGE INSTRUCTIONS-Data transfer instructions-input- output instructions, address transfer, Flag transfer, arithmetic, logical, shift, and rotate instructions.conditional and unconditional transfer, iteration control, interrupts and process control instructions, assembler directives, Programming with assembly language instructions.

**UNIT IV**

INPUT -OUTPUT ORGANIZATION-Peripheral devices, input-output interface-I/0 Bus and interface modules, I/O versus Memory bus, isolated versus memory mapped I/O, Modes of transfer-Programmed I/O, Interrupt-initiated I/O, priority interrupts-Daisy chaining, parallel priority, interrupt cycle, DMA-DMA control, DMA transfer, Input output processor-CPU-IOP communication.

**UNIT V**

PIPELINE AND VECTOR PROCESSING**:** Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processors.

MULTI PROCESSORS**:** Characteristics or Multiprocessors, Interconnection Structures, Interprocessor Arbitration, InterProcessor Communication and Synchronization Cache Coherance,Shared Memory Multiprocessors.

TEXT BOOKS:

1. Computer System Architecture, M. Morris Mano , 3rd Edition, Pearson Education,2008.
2. Microprocessors and Interfacing, Douglas Hall, Tata McGraw-Hill.

REFERENCE BOOKS:

**1.** Computer Organization, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Vth Edition, McGraw Hill.

**MASTER OF COMPUTER APPLICATIONS**

**I Year I Semester**

**MC130 OBJECT ORIENTED PROGRAMMING THROUGH C++**

**UNIT I**

Different paradigms for problem solving, need for OOP paradigm, classes and instances, fundamental characteristics of OOP (Alan key), differences between OOP and Procedure Oriented Programming.
C++ Basics: Structure of a C++ program, Data types, Declaration of variables, Expressions, Operators, Operator Precedence, Evaluation of expressions, Type conversions, Pointers, Arrays, Pointers and Arrays, Strings, Structures, References. Flow control statements- if, switch, while, for, do, break, continue, goto statements.

**UNIT II**

C++ Functions-Scope of variables, Parameter passing methods, Default arguments, inline functions, Recursive functions, Pointers to functions.
C++ Classes And Data Abstraction: Class definition, Class objects, Class scope, this pointer, Friends to a class, Static class members, Constant member functions, Constructors and Destructors, Data abstraction, ADT and information hiding.

**UNIT III**

Dynamic memory allocation and deallocation operators-new and delete, Dynamic creation and destruction of objects, Preprocessor directives, command line arguments, name spaces.
Polymorphism: Function overloading, Operator overloading, Generic programming-necessity of templates, Function templates and class templates.

**UNIT IV**

Inheritance: Defining a class hierarchy, Different forms of inheritance, Defining the Base and Derived classes, Access to the base class members, Base and Derived class construction, Destructors, Virtual base class.
Virtual Functions And Run Time Polymorphism: Overriding, Static and Dynamic bindings, Base and Derived class virtual functions, Dynamic binding through virtual functions, Virtual function call mechanism, Pure virtual functions, Abstract classes,Virtual destructors.

**UNIT V**

C++ I/O: I/O using C functions, C++ Stream classes hierarchy, Stream I/O, File streams and String streams,File Operations, Overloading << and >> operators, Error handling during file operations, Formatted I/O.
Exception Handling: Benefits of exception handling, Throwing an exception, The try block, Catching an exception, Exception objects, Exception specifications, Stack unwinding, Rethrowing an exception, Catching all exceptions.

**TEXT BOOKS:**

1. C++, The Complete Reference, 4th Edition, Herbert Schildt, TMH.
2. Object Oriented Programming in C++, 4th Edition, R.Lafore, Pearson Education

REFERENCE BOOKS:

1. An Introduction to OOP, 3rd Edition, T. Budd, Pearson Education,2008.
2. Programming Principles and Practice Using C++, B.Stroutstrup, Pearson Education.
3. Problem solving with C++, 6th Edition, Walter Savitch, Pearson Education,2007.
4. Mastering C++, K.R.Venu Gopal, Raj Kumar and T.Ravi Shankar, TMH.
5. OOP in C++, 3rd Edition, T.Gaddis, J.Walters and G.Muganda, Wiley DreamTech Press.
6. An Introduction to OOP in C++ with applications in Computer Graphics, 2nd Edition, G.M.Seed, Springer.
7. Programming with ANSI C++, B.Trivedi,Oxford Press.
8. Programming in C++,M.T.Somasekara,PHI.

**MASTER OF COMPUTER APPLICATIONS**
**I Year I Semester**

**MC140 OPERATING SYSTEMS**

**UNIT I**

Operating System Introduction: Operating Systems objectives and functions, Computer System Architecture, OS Structure, OS Operations, Evolution of Operating Systems - Simple Batch, Multi programmed, time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, Special -Purpose Systems, Operating System services, User OS Interface, System Calls, Types of System Calls, System Programs, Operating System Design and Implementation, OS Structure, Virtual Machines.

**UNIT II**

Process and CPU Scheduling - Process concepts-The Process, Process States, Process Control Block, Threads, Process Scheduling-Scheduling Queues, Schedulers, Context Switch, Dispatcher, Scheduling Criteria, Scheduling algorithms, Multiple-Processor Scheduling, Real-Time Scheduling, Thread scheduling.

Process Coordination – Process Synchronization, The Critical Section Problem, Peterson's solution, Synchronization Hardware, Semaphores.

**UNIT III**

Memory Management and Virtual Memory - Logical & Physical Address Space, Swapping, Contiguous Allocation, Paging, Structure of Page Table, Segmentation, Segmentation with Paging, Virtual Memory, Demand Paging, Page Replacement, Page Replacement Algorithms, Allocation of Frames, Thrashing.

**UNIT IV**

File System Interface - The Concept of a File, Access methods, Directory Structure, File System Mounting, File Sharing, Protection, File System Implementation - File System Structure, File System Implementation, Directory Implementation, Allocation methods, Free-space Management.

**UNIT V**

Mass Storage Structure – Overview of Mass Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling.

Deadlocks - System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery from Deadlock.

**TEXT BOOKS:**

1. Operating System Principles , Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 8th Edition, Wiley Student Edition
2. Operating Systems – Internals and Design Principles, W. Stallings, 6th Edition, Pearson.

**REFERENCE BOOKS:**

1. Modern Operating Systems, Andrew S Tanenbaum, 3rd Edition, PHI
2. Operating Systems A concept-based Approach, 2nd Edition, D.M.Dhamdhere, TMH.
3. Principles of Operating Systems , B.L.Stuart, Cengage learning, India Edition.
4. Operating Systems, A.S.Godbole,2nd Edition, TMH
5. An Introduction to Operating Systems, P.C.P. Bhatt, PHI.
6. Operating Systems,S.Haldar and A.A.Aravind,Pearson Education.

**MASTER OF COMPUTER APPLICATIONS**
**I Year I Semester**

## MC150 DATABASE SYSTEMS

**Objectives:**

By the end of the course, you will know:

- History and Structure of databases
- How to design a database
- How to convert the design into the appropriate tables
- Handling Keys appropriately
- Enforcing Integrity Constraints to keep the database consistent
- Querying relational data ,Triggers, Procedures and Cursors
- Normalizing the tables to eliminate redundancies
- Transaction Management
- Storage Optimizing Strategies for easy retrieval of data through index
- Processing the queries

**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, ER diagrams,. Relational Model: Introduction to the Relational Model – Integrity Constraints Over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design, Introduction to Views –Altering Tables and Views, Relational Algebra, Basic SQL Queries, Nested Queries, Complex Integrity Constraints in SQL, Triggers.

**UNIT II**

**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.

**UNIT III**

**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 Deadlocks, Specialized Locking Techniques – Concurrency Control without Locking.

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

**UNIT IV**

**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, Indexes and Performance Tuning

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

Disk Space Management, Buffer Manager, Files of Records, Page Formats, Record Formats 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, Extendable Vs Linear Hashing.

**UNIT V**

**Overview Of Query Evaluation:** The System Catalog, Introduction to Operator Evaluation, Algorithms for Relational Operations, Introduction to Query Optimization, Alternative Plans: A Motivating Example, What a Typical Optimizer Does?

**Evaluating Relational Operators:** The' Selection Operation, General Selection Conditions, The Projection Operation, The Join Operation, The Set Operations, Aggregate Operations, The Impact of Buffering.

**A Typical Relational Query Optimizer:** Translating SQL Queries into Algebra, Estimating the Cost of a Plan, Relational Algebra Equivalences, Enumeration of Alternative Plans, Nested Subqueries, The System R Optimizer, Other Approaches to Query Optimization.

**TEXT BOOKS:**

1. Data base Management Systems, Raghu Ramakrishnan, Johannes Gehrke, TMH, 3rd Edition.Database Systems: The Complete Book, Hector Garcia-Molina, Jeffrey Ullman, Jennifer Widom

**REFERENCE BOOKS:**

1. Database Systems implementation Hector Garcia-Molina, Jeffrey Ullman, Jennifer Widom
2. Introduction to Database Systems, C.J.Date, Pearson Education.
3. Data base System Concepts, A.Silberschatz, H.F. Korth, S.Sudarshan, McGraw hill, VI edition
4. Fundamentals of Database Systems , Ramez Elmasri, Shamkant B.Navathe, Pearson Education,
5. Database Management System Oracle SQL and PL/SQL, P.K.Das Gupta, PHI.
6. Database System Concepts, Peter Rob & Carlos Coronel, Cengage Learning, 2008.

**MASTER OF COMPUTER APPLICATIONS**

**I Year I Semester**

**MC160 COMPUTER ARCHITECTURE & ORGANIZATION AND OPERATING SYSTEMS LAB**

**COMPUTER ARCHITECTURE & ORGANIZATION LAB:**

Write assembly language programs for the following using MASAM.

1. Write assembly language programs to evaluate the expressions:
   i) a = b + c –d * e
   ii) z = x * y + w – v +u / k
   a. Considering 8-bit, 16 bit and 32 bit binary numbers as b, c, d, e.
   b. Considering 2 digit, 4digit and 8 digit BCD numbers.
   Take the input in consecutive memory locations and results also.
   Display the results by using "int xx" of 8086. Validate program for the boundary conditions.
2. Write an ALP of 8086 to add two exponential numbers which are in IEEE 754 notation. Display the results by using "int xx" of 8086. Validate program for the boundary conditions.
3. Write an ALP of 8086 to take N numbers as input. And do the following operations on them.
   a) Arrange in ascending and Descending order.
   b) Find max and minimum
   c) Find average
   Considering 8-bit, 16 bit binary numbers and 2 digit, 4digit and 8 digit BCD numbers. Display the results by using "int xx" of 8086. Validate program for the boundaryconditions.
4. Write an ALP of 8086 to take a string of as input (in 'C' format)and do the following Operations on it.
   a) Find the length b) Find it is Palindrome or not
   c) Find whether given string substring or not.
   d) Reverse a string e) Concatenate by taking another sting
   Display the results by using "int xx" of 8086.
5. Write the ALP to implement the above operations as procedures and call from the main procedure.
6. Write an ALP of 8086 to find the factorial of a given number as a Procedure and call from the main program which display the result.
7. Write an assembly language program to encrypt digits as shown below:
   Input digit : 0 1 2 3 4 5 6 7 8 9
   Encrypted digit : 4 6 9 5 0 3 1 8 7 2

Your program should accept a string consisting of digits. The encrypted string should be displayed using "int xx" of 8086.

8. Write a procedure to locate a character in a given string. The procedure receives a pointer to a string and character to be located. When the first occurrence of the character is located , its position is returned to main. If no match is found , a negative value is returned. The main procedure requests a character string and a character to be located and displays the result.
9. Write an assembly language program to read a string of characters from the user and that prints the vowel count . Display the results by using "int xx" of 8086.

ex. Input : Advanced Programming in UNIX

Out put:

| Vowel | count |
|---|---|
| a or A | 3 |
| e or E | 1 |
| i or I | 3 |
| o or O | 1 |
| u or U | 1 |

10. A computer uses RAM chips of 1024 X 1 capacity.
    a) How many chips are needed, and how should their address lines be connected to provide a memory capacity of 1024 bytes?
    b) How many chips are needed to provide a memory capacity of 16K bytes?
11. A computer employs RAM chips of 256X8 and ROM chips of 1024 X 8. The computer needs 2K bytes of RAM, 4K bytes of ROM, and four interface units, each with four registers. A memory-mapped I/O configuration is used. The two highest-order bits of the address bus are assigned 00 for RAM, 01 for ROM, 10 for interface registers.
    a. How many RAM and ROM chips are needed?
    b. Draw a memory-address map for the system.
    c. Give the address range in hexadecimal for RAM, ROM and interface.
12. Obtain the complement function for the match logic of one word in an associative memory. Draw the logic diagram for it and compare with the actual match logic diagram.
13. A two-way set associative cache memory uses blocks of four words. The cache can accommodate a total of 2048 words from main memory. The main memory size is 128K X 32.
    a. Formulate all pertinent information required to construct the cache memory.
    b. What is the size of the cache memory?
14. A digital computer has a memory unit of 64K X 16 and a cache memory of 1K words. The cache uses direct mapping with a block size of four words.
    a. How many bits are there in each word of cache, and how are they divided into functions? Include a valid bit.
    b. How many bits are there in the tag, index, block, and word fields of the address format?
    c. How many blocks can the cache accommodate?
15. An address space is specified by 24 bits and the corresponding memory space by 16 bits.
    a. How many words are there in the address space?

b. How many words are there in the memory space?

c. If a page consists of 2K words, how many pages and blocks are there in the system.

16. A virtual memory has a page size of 1K words. There are eight pages and four blocks. The associative memory page table contains the following entries. Make a list of all virtual addresses(in decimal) that will cause a page fault.

| Page | Block |
|---|---|
| 0 | 3 |
| 1 | 1 |
| 4 | 2 |
| 6 | 0 |

**TEXT BOOKS:**


1. IBM PC Assembly Language and Programming, P. Abel, 5th Edition, PHI.
2. Introduction to Assembly Language Programming, Sivarama P.Dandamudi, Springer Int. Edition,2003.
3. The 8088 and 8086 Microprocessors: Programming , Interfacing,Software,Hardware and Application,4th edition,W.A.Triebel,A.Singh,N.K.Srinath,Pearson Education.


**OPERATING SYSTEMS LAB:**

**List of Sample Problems:**

1. Simulate FCFS CPU scheduling algorithm.
2. Simulate non preemptive SJF CPU scheduling algorithm.
3. Simulate preemptive SJF CPU scheduling algorithm.
4. Simulate non preemptive priority CPU scheduling algorithm.
5. Simulate preemptive priority CPU scheduling algorithm.
6. Simulate round robin CPU scheduling algorithm.
7. Simulate Bankers Algorithm for Dead Lock Avoidance.
8. Write a C program to simulate producer-consumer problem using Semaphores
9. Write a C program to simulate the concept of Dining-philosophers problem.
10. Simulate following page replacement algorithms:

    a)FIFO b) LRU c) OPTIMAL

11. : Write a C program to simulate disk scheduling algorithms

    a) FCFS b) SCAN c) C-SCAN d) SSTF e) LOOK f) C-LOOK

**MASTER OF COMPUTER APPLICATIONS**

**I Year I Semester**

**MC170 OBJECT ORIENTED PROGRAMMING THROUGH C++ LAB**

1. Write a C++ program to find the sum of individual digits of a positive integer.
2. 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.
3. Write a C++ program to generate all the prime numbers between 1 and n , where n is a value supplied by the user.
4. Write C++ programs that use both recursive and non-recursive functions
   a) To find the factorial of a given integer. b) To find the GCD of two given integers.
   c) To find the nth Fibonacci number.
5. Write a C++ program that uses a recursive function for solving Towers of Hanoi problem.
6. Write a C++ program that uses functions
   a) To swap two integers. b) To swap two characters.
   c) To swap two reals. Note: Use overloaded functions.
7. Write a C++ program to find both the largest and smallest number in a list of integers.
8. Write a C++ program to sort a list of numbers in ascending order.
9. Write a C++ program that uses function templates to solve problems-7&8.
10. Write a C++ program to sort a list of names in ascending order.
11. Write a C++ program to implement the matrix ADT using a class. The operations supported by this ADT are:
    a)Reading a matrix. c) Addition of two matrices.
    b)Printing a matrix. d)Multiplication of two matrices.
12. Implement the matrix ADT presented in the problem-11 using overloaded operators (<<, >>, +, *) and templates.
13. Implement the complex number ADT in C++ using a class. The complex ADT is used to represent complex numbers of the form c=a+ib, where a and b are real numbers. The operations supported by this ADT are:
    a) Reading a complex number. b) Writing a complex number.
    c) Addition of two complex numbers. d) Multiplication of two complex numbers.
14. Write a C++ program that overloads the + operator and relational operators (suitable) to perform the following operations:
    a) Concatenation of two strings. b)Comparison of two strings.
15. Implement the complex number ADT in C++ using a class. The complex ADT is used to represent complex numbers of the form c=a+ib, where a and b are real numbers. The operations supported by this ADT are:
    a) Reading a complex number. b) Writing a complex number.
    c) Addition of two complex numbers. d) Multiplication of two complex numbers.
    Note: 1. overload << and >> operators in part a) and part b).
    2. overload +, * operators in parts c) and d).

16. Write a template based C++ program that determines if a particular value occurs in an array of values.
17. Write a C++ program that uses functions to perform the following operations:
    a) Insert a sub-string into the given main string from a given position.
    b) Delete n characters from a given position in a given string.
18. Write a C++ program that uses a function to reverse the given character string in place, without any duplication of characters.
19. Write a C++ program to make the frequency count of letters in a given text.
20. Write a C++ program to count the lines, words and characters in a given text.
21. Write a C++ program to determine if the given string is a palindrome or not.
22. Write a C++ program to make frequency count of words in a given text.
23. Write a C++ program that displays the position or index in the string S where the string t begins , or –1 if S doesn't contain t.
24. 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.
25. Write a C++ program that counts the number of 1 bits in a given integer.
26. Write a C++ program to generate Pascal's triangle.
27. Write a C++ program to construct of pyramid of numbers.
28. Write a C++ program to compute the Sine series.
29. Write a C++ program that converts Roman numeral into an Arabic integer.
30. Write a C++ program which converts a positive Arabic integer into its corresponding Roman Numeral.
31. Write a C++ program to display the contents of a text file.
32. Write a C++ program which copies one file to another.
33. Write a C++ program to that counts the characters, lines and words in the text file.
34. Write a C++ program to change a specific character in a file.
    Note: Filename , number of the byte in the file to be changed and the new character are specified on the command line.
35. Write a C++ program to reverse the first n characters in a file.
36. Write a C++ program that uses a function to delete all duplicate characters in the given string.
37. Write a C++ program that uses a function to convert a number to a character string.
38. Write a C++ program that uses a recursive function to find the binary equivalent of a given non-negative integer n.
39. Write a C++ program to generate prime numbers up to n using Sieve of Eratosthenes method.
40. Write a C++ program
    a) To write an object to a file.
    b) To read an object from the file.
41. Write C++ programs that illustrate how the following forms of inheritance are supported:
    a) Single inheritance b) Multiple inheritance
    c) Multi level inheritance d) Hierarchical inheritance
42. Write a C++ program that illustrates the order of execution of constructors and destructors when new class is derived from more than one base class.
43. Write a C++ program that illustrates how run time polymorphism is achieved using virtual functions.
44. Write a C++ program that illustrates the role of virtual base class in building class hierarchy.
45. Write a C++ program that illustrates the role of abstract class in building class hierarchy.

TEXT BOOKS:


1. Mastering C++, K.R.Venu Gopal, Raj Kumar and T.Ravi Shankar, TMH.
2. C++ Programming, D.S.Malik, Cengage Learning.
3. Practical C++ Programming,S.Qualline,SPD.
4. Object Oriented Programming with C++, E.Balaguruswamy, 4th Edition, TMH,2008.
5. OOP with C++,S.Sahay,Oxford Higher Education.
6. C++ and OOP Paradigm,D.Jana,2nd Edition, PHI
7. Fundamentals of C++ Programming,S.Subramanian,Jaico Publishing House.

**MASTER OF COMPUTER APPLICATIONS**
**I Year I Semester**
**MC180 DATABASE SYSTEMS LAB**

**List of Sample Problems:**

1) Creation, altering and droping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
2) Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints.
   Example:- Select the roll number and name of the student who secured fourth rank in the class.
3) Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
4) Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)
5) i)Creation of simple PL/SQL program which includes declaration section, executable section and exception –handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
   ii)Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
6) Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
7) Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE- APPLICATION ERROR.
8) Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
9) Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
10) Program development using creation of package specification, package bodies, private objects, package variables and cursors and calling stored packages.
11) Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
12) Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers

**Example Problems:**

**1. Creating tables for various relations (in SQL):**

CLIENT_MASTER (CLIENTNO, NAME, ADDRESS1, ADDRESS2, CITY, PINCODE, STATE, BALDUE)

PRODUCT_MASTER( PRODUCTNO, DESCRIPTION, PROFITPERCENT, UNITMEASURE,

QTY_ON_HAND, REORDERLVL, SELLPRICE, COSTPRICE)

SALESMAN_MASTER (SALESMANNO, SALESMANNAME, ADDRESS1, ADDRESS2, CITY, PINCODE, STATE, SLAMT, TGTTOGET, YTDSALES, RESALES)

SALES_ORDER(OREDRENO, CLIENTNO, ORDERDATE, DELYADDR, SALESMANNO, DELYTYPE, BILLYN, DELYDATE, ORDERSTATUS)

SALES_ORDER_DETAILS(ORDERNO, PRODUCTNO, QTYORDERED, QTYDISP, PRODUCTRATE)

i) Insert data into their respective table.
ii) Exercise on retrieving records from table.
  a. Find out the names of all clients.
  b. Retrieve the entire contents of the client_master table.
  c. Retrieve the list of names, city and the state of all clients.
  d. List the various products available from the Product_master table.
  e. List all the clients who are located in Mumbai.
  f. Find the names of salesmen who have a salary equal to Rs.3000.
iii. Exercise on updating records in a table
  a. Change the city of client No 'C00005' to 'Bangalore'.
  b. Change the BalDue of ClientNo 'C00001' to Rs.1000.
  c. Change the cost price of 'Trousers' to Rs.950.00.
  d. Change the city of the salesman to Pune.
iv. Exercise on deleting records in a table
  a. Delete all sales men from the salesman_master whose salaries are equal to Rs.35000
  b. Delete all products from product_master where the quantity on hand is equal to 100.
  c. Delete from Client _Master where the column state holds the value 'Tamilnadu'.
v. Exercise on altering the table
  a. Add a column called 'Telephone' of data type 'number' and size='10' to the Client Master table.
  b. Change the size of Sell Price column in Product_Master to 10,2.
vi. Exercise on deleting the table structure along with the data
  a. Destroy the table Client_Master along with the data.
vii. Exercise on renaming the table
  a. Change the name of the Salesman_Master table to Sman_mast.
2. Using the tables created previously generate the SQL statements for the operations mentioned below. The tables in user are as follows:

Client_Master

Product_Master

Salesman_Master

Sales_Order

Sales_Order_Details

i) Perform the following computations on table data:

a. List the names of all clients having 'a' as the second letter in their names.
b. List the clients who stay in a city whose first letter is 'M'.
c. List all clients who stay in 'Bangalore' or 'Mangalore'.
d. List all clients whose BalDue is greater than value 10000.
e. List all information from the Sales_Order table for orders placed in the month of June.
f. List the order information for Client No'C00001' and 'C00002'.
g. List products whose selling price is greater than 500 and less than or equal to 750.
h. List products whose selling price is more than 500.Calculate a new selling price as, original selling price * .15. Rename the new column in the output of the above query as new_price.
i. List the names, city and state of clients who are not in the state of 'Maharashtra'.
j. Count the total no of orders.
k. Calculate the average price of all the products.
l. Determine the maximum and minimum products prices. Rename the output as max_price and min_price respectively.
m. Count the no of products having price less than or equal to 500.
n. List all the products whose Qty On Hand is less than reorder level.

ii) Exercise on Date Manipulation

a. List the order number and day on which clients on placed their order.
b. List the months (in alphabets) and date when the orders must be delivered.
c. List the Order Date in the format 'DD-Month-YY'. e.g.12-February-02.
d. List the date, 15 days after today's date.

iii). Exercises on using Having and Group by Clauses:

a. Print the description and total qty sold for each product.
b. Find the value of each product sold
c. Calculate the average qty sold for each client that has a maximum order value of 15000.00.
d. Find out the total of all the billed orders for the month of June.

iv). Exercises on Joins and Correlation:

a) Find out the products, which have been sold to 'Ivan Bay Ross'.
b) Find out the products and their quantities that will have to be delivered in the current month.
c) List the product no and description of constantly sold products (i.e. rapidly moving products).
d) Find the names of clients who have purchased 'Trousers'.
e) List the products and orders from customers who have ordered less than 5 units of 'Pull Overs'.
f) Find the products and their quantities for the orders placed by 'Ivan Bay Ross' and 'Mamta Muzumdar'.
g) Find the products and their quantities for the orders placed by Client No 'C00001' and 'C00002'.

v). Exercise on Sub-queries:

a. Find the Product No and Description of non_moving products i.e. Products not being sold.
b. List the customer Name Address1, Address2,City and Pin Code for the client who has placed order no 'O19001'.
c. List the client names that have placed orders before the month of May'02.
d. List if the product 'Lycra Top' has been ordered by any client and print the Client_no ,Name to whom it was sold.

e. List the names of clients who have placed orders worth Rs.10,000 or more.

3) Creating Views

4) Writing Assertions

5) Writing Triggers

6) Implementing Operations on relations (tables) using PL/SQL

Ex: Write a PL/SQL code block to calculate the area of a circle for a value of radius varying from 5 to 9. Store the radius and the corresponding values of calculated area in an empty table named Areas(radius,area).

7) Creating FORMS.
8) Generating REPORTS.

**Additional Problems:**

**i) Databases :**

**Objective**: This lab enables the students to practice the concepts learnt in the subject Databases by developing a database for an example company named "Roadway Travels" whose description is as follows. The student is expected to practice the designing, developing and querying a database in the context of example database "Roadway travels". Students are expected to use "Mysql" database.

**Roadway Travels**

**"Roadway Travels"** is in business since 1997 with several buses connecting different places in India. Its main office is located in Hyderabad.

The company wants to ***computerize its operations*** in the following areas:

- Reservations and Ticketing
- Cancellations

**Reservations & Cancellation:**

Reservations are directly handled by booking office. Reservations can be made 30 days in advance and tickets issued to passenger. One Passenger/person can book many tickets (to his/her family).

Cancellations are also directly handed at the booking office.

In the process of ***computerization*** of **Roadway Travels** you have to design and develop a Database which consists the data of Buses, Passengers, Tickets, and Reservation and cancellation details. You should also develop query's using SQL to retrieve the data from the database.

The above process involves many steps like 1. Analyzing the problem and identifying the Entities and Relationships, 2. E-R Model 3. Relational Model 4. Normalization 5. Creating the database 6. Querying. ***Students are supposed to work on these steps week wise and finally create a complete "Database System" to Roadway Travels.*** Examples are given at every experiment for guidance to

students.

**Experiment 1: E-R Model**

Analyze the problem carefully and come up with the entities in it. Identify what data has to be persisted in the database. This contains the entities, attributes etc.

Identify the primary keys for all the entities. Identify the other keys like candidate keys, partial keys, if any.

Example: **Entities:**

1. BUS
2. Ticket
3. Passenger

**Relationships:**

1. Reservation
2. Cancellation

**PRIMARY KEY ATTRIBUTES:**

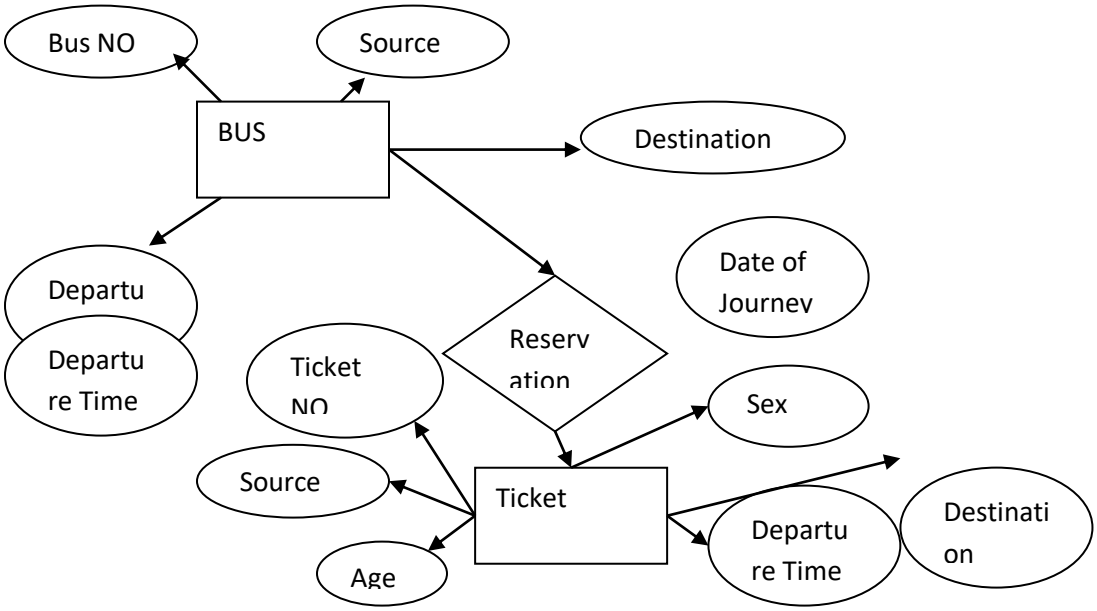
1. Ticket ID (Ticket Entity)
2. Passport ID (Passenger Entity)
3. Bus_NO(Bus Entity)

Apart from the above mentioned entities you can identify more. The above mentioned are few.
**Note:** *The student is required to submit a document by writing the Entities and Keys to the lab teacher.*

**Experiment 2: Concept design with E-R Model**

Relate the entities appropriately. Apply cardinalities for each relationship. Identify strong entities and weak entities (if any). Indicate the type of relationships (total / partial). Try to incorporate generalization, aggregation, specialization etc wherever required.

**Example: E-R diagram for bus**




**Note:** *The student is required to submit a document by drawing the* E-R Diagram *to the* lab teacher.

**Experiment 3: Relational Model**

Represent all the entities (Strong, Weak) in tabular fashion. Represent relationships in a tabular fashion. There are different ways of representing relationships as tables based on the cardinality. Represent attributes as columns in tables or as tables based on the requirement. Different types of attributes (Composite, Multi-valued, and Derived) have different way of representation.

Example: The passenger tables look as below. This is an example. You can add more attributes based on your E-R model. This is not a normalized table.

Passenger

| Name | Age | Sex | Address | Passport ID |
|---|---|---|---|---|
| | | | | Ticket_id |
| | | | | |
| | | | | |

**Note:** *The student is required to submit a document by Represent relationships in a tabular fashion to the lab teacher.*

**Experiment 4: Normalization**

Database normalization is a technique for designing relational database tables to minimize duplication of information and, in so doing, to safeguard the database against certain types of logical or structural problems, namely data anomalies. For example, when multiple instances of a given piece of information occur in a table, the possibility exists that these instances will not be kept consistent when the data within the table is updated, leading to a loss of data integrity. A table that is sufficiently normalized is less vulnerable to problems of this kind, because its structure reflects the basic assumptions for when multiple instances of the same information should be represented by a single instance only.

For the above table in the First normalization we can remove the multi valued attribute Ticket_id and place it in another table along with the primary key of passenger.

**First Normal Form: The above table can be divided into two tables as shown below.**

Passenger

| Name | Age | Sex | Address | Passport ID |
|---|---|---|---|---|
| | | | | |

| Passport ID | Ticket_id |
|---|---|
| | |

You can do the second and third normal forms if required. Any how Normalized tables are given at the end.

**Experiment 5: Installation of Mysql and practicing DDL commands**

Installation of MySql. In this week you will learn Creating databases, How to create tables, altering the database, dropping tables and databases if not required. You will also try truncate, rename commands etc.

Example for creation of a normalized "Passenger" table.

```
CREATE TABLE Passenger (
   Passport_id    INTEGER   PRIMARY KEY,
   Name   VARCHAR (50) Not NULL,
   Age    Integer Not NULL,
   Sex   Char,
   Address VARCHAR (50) Not NULL);
```

Similarly create all other tables.
**Note: Detailed creation of tables is given at the end.**

**Experiment 6: Practicing DML commands**

DML commands are used to for managing data within schema objects. Some examples:

- SELECT - retrieve data from the a database
- INSERT - insert data into a table
- UPDATE - updates existing data within a table
- DELETE - deletes all records from a table, the space for the records remain

**Inserting values into "Bus" table:**

Insert into Bus values (1234,'hyderabad', 'tirupathi');
Insert into Bus values (2345,'hyderabd','Banglore');
Insert into Bus values (23,'hyderabd','Kolkata');
Insert into Bus values (45,'Tirupathi,'Banglore');
Insert into Bus values (34,'hyderabd','Chennai');

**Inserting values into "Passenger" table:**

Insert into Passenger values (1, 45,'ramesh', 45,'M','abc123');
Insert into Passenger values (2, 78,'geetha', 36,'F','abc124');
Insert into Passenger values (45, 90,'ram', 30,'M','abc12');
Insert into Passenger values (67, 89,'ravi', 50,'M','abc14');
Insert into Passenger values (56, 22,'seetha', 32,'F','abc55');

**Few more Examples of DML commands:**

Select * from Bus; (selects all the attributes and display)
UPDATE BUS SET Bus No = 1 WHERE BUS NO=2;

**Experiment 7: Querying**

In this week you are going to practice queries (along with sub queries) using ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, Constraints etc.

**Practice the following Queries:**

1. Display unique PNR_no of all passengers.
2. Display all the names of male passengers.
3. Display the ticket numbers and names of all the passengers.
4. Find the ticket numbers of the passengers whose name start with 'r' and ends with 'h'.
5. Find the names of passengers whose age is between 30 and 45.
6. Display all the passengers names beginning with 'A'
7. Display the sorted list of passengers names

**Experiment 8 and Experiment 9: Querying (continued…)**

You are going to practice queries using Aggregate functions (COUNT, SUM, AVG, and MAX and MIN),

GROUP BY, HAVING and Creation and dropping of Views.

1. Write a Query to display the Information present in the Passenger and cancellation tables. **Hint:** Use UNION Operator.
2. Display the number of days in a week on which the 9W01 bus is available.
3. Find number of tickets booked for each PNR_no using GROUP BY CLAUSE. **Hint:** Use GROUP BY on PNR_No.
4. Find the distinct PNR numbers that are present.
5. Find the number of tickets booked by a passenger where the number of seats is greater than 1. **Hint:** Use GROUP BY, WHERE and HAVING CLAUSES.
6. Find the total number of cancelled seats.

**Experiment 10: Triggers**

In this week you are going to work on Triggers. Creation of insert trigger, delete trigger, update trigger. Practice triggers using the above database.

Eg: **CREATE TRIGGER updcheck BEFORE UPDATE ON passenger**

```
FOR EACH ROW
BEGIN
 IF NEW.TickentNO > 60 THEN
   SET New.Tickent no = Ticket no;
ELSE
        SET New.Ticketno = 0;
END IF;
END;
```

**Experiment 11: Procedures**

In this session you are going to learn Creation of stored procedure, Execution of procedure and modification of procedure. Practice procedures using the above database.

**Eg:CREATE PROCEDURE myProc()**
**BEGIN**
**SELECT COUNT(Tickets) FROM Ticket WHERE age>=40;**
**End;**

**Experiment 12: Cursors**

In this week you need to do the following: Declare a cursor that defines a result set.
Open the cursor to establish the result set. Fetch the data into local variables as needed from the cursor, one row at a time. Close the cursor when done

```
CREATE PROCEDURE myProc(in_customer_id INT)
 BEGIN
 DECLARE v_id   INT;
 DECLARE v_name VARCHAR(30);
```

**DECLARE c1 CURSOR FOR SELECT stdId,stdFirstname FROM students WHERE stdId=in_customer_id;**
**OPEN c1;**
**FETCH c1 into v_id, v_name;**
**Close c1;**
**END;**

**Tables**
**BUS**
Bus No: Varchar: PK (public key)
Source : Varchar
Destination : Varchar
**Passenger**
PPNO: Varchar(15)) : PK
Name: Varchar(15)
Age : int (4)
Sex:Char(10) : Male / Female
Address: VarChar(20)
**Passenger_Tickets**
PPNO: Varchar(15)) : PK
Ticket_No: Numeric (9)
**Reservation**
PNR_No: Numeric(9) : FK
Journey_date : datetime(8)
No_of_seats : int (8)
Address : Varchar (50)
Contact_No: Numeric (9) --> Should not be less than 9 and Should not accept any other character other than Integer
Status: Char (2) : Yes / No

Cancellation
PNR_No: Numeric(9) : FK
Journey_date : datetime(8)
No_of_seats : int (8)
Address : Varchar (50)
Contact_No: Numeric (9) --> Should not be less than 9 and Should not accept any other character other than Integer
Status: Char (2) : Yes / No

**Ticket**
Ticket_No: Numeric (9): PK
Journey_date : datetime(8)
Age : int (4)
Sex:Char(10) : Male / Female
Source : Varchar

Destination : Varchar
Dep_time : Varchar

**TEXT BOOKS:**

1.Introduction to SQL,Rick F.Vander Lans,Pearson education.
2.Oracle PL/SQL, B.Rosenzweig and E.Silvestrova,Pearson education.
3.Oracle PL/SQL Programming,Steven Feuerstein,SPD.
4.SQL & PL/SQL for Oracle 10g,Black Book,Dr.P.S.Deshpande,Dream Tech.
5.Oracle Database 11g PL/SQL Programming,M.Mc Laughlin,TMH.
6.SQL Fundamentals,J.J.Patrick,Pearson Education.

**MASTER OF COMPUTER APPLICATIONS**
**I Year I Semester**

**MC19A PROFESSIONAL COMMUNICATION SKILLS**
**(AUDIT-1)**

**Course Objectives:**

• To teach the four language skills - Listening, Speaking, Reading and Writing; critical thinking skills to students.

• To enable students comprehend the concept of communication.

• To help students cultivate the habit of Reading and develop their critical reading skills.

**Course Outcomes:**

• Students are trained to convert the conceptual understanding of communication into every day practice.

• Students are expected to be ready for placements.

• Students are prepared to communicate their ideas relevantly and coherently in professional writing.

**UNIT I INTRODUCTION**

Basics of Communication - Principles of Communication - Types of Communication – Stages of Communication – Verbal and Non-verbal Communication – Channels of Communication – Barriers to Effective Communication – Formal and Informal Expressions in Various Situations.

**UNIT II READING & STUDY SKILLS**

Reading Comprehension – Reading Strategies - Skimming and Scanning- Intensive and Extensive Reading– Unknown Passage for Comprehension - Critical Reading of Short Stories – Study Skills – Note Making – Summarizing – Articles and Prepositions – Synonyms and Antonyms

**UNIT III**
**WRITING SKILLS**

Difference between Spoken and Written Communication- Features of Effective Writing - Formation of a Sentence – SVOs and SVOC patterns – Types of sentences- Common errors in Writing - Writing coherent sentences using connectives and conjunctions- Written Presentation Skills – Tenses – Concord – Question Tags - Practice Exercises - One Word Substitutes – Words Often Confused and Misspelt.

**UNIT IV**
**PROFESSIONAL WRITING**

Letter writing – Types, Parts and Styles of Formal Letters – Language to be used in Formal Letters – Letters of Enquiry, Complaint, and Apology with Replies – Letter of Application -Resume – E-mail – Active and Passive Voice.

**UNIT V**
**REPORT WRITING**

Types of Reports – Formats of Reports – Memo Format – Letter Format and Manuscript Format- Parts of Technical Report – Informational, Analytical and Project Reports – Idioms and Phrases.

**REFERENCE BOOKS:**

1. Meenakshi Raman & Sangeetha Sharma. 2012. *Technical Communication.* New Delhi

*2*. Rizvi, M. A. 2005. *Effective Technical Communication.* New Delhi: Tata McGraw Hill

3. Sanjay Kumar & Pushp Latha. 2012. *Communication Skills.* New Delhi: OUP

4. Er. A. K. Jain, Dr. Pravin S. R. Bhatia & Dr. A. M. Sheikh. 2013. *Professional Communication Skills*. S. Chand Publishers. New Delhi.

5. Farhathullah, T.M. 2009. *English for Business Communication.* Bangalore: Prism

6. Bikram K Das. 2011. Functional Grammar and Spoken and Written Communication in English. Kolkata: Orient Blackswan

7. Kiranmai Dutt, P *et al. 2011. A Course in Communication Skills. New Delhi: CUP India*

8. Krishnaswamy, N. 2000. *Modern English – A Book of Grammar, Vocabulary and Usage.* Macmillan India Pvt. Ltd

9. Ramachandran, K K. *et al.* 2007. *Business Communication.* New Delhi: Macmillan

10. Taylor, Ken. 2011. *50 ways to improve your Business English*. Hyderabad: Orient Blackswan

**MASTER OF COMPUTER APPLICATIONS**
**I Year I Semester**

**MC19B PERSONALITY DEVELOPMENT THROUGH LIFE ENLIGHTENMENT SKILLS**
**(AUDIT-1)**

**Course Objectives**
1. To learn to achieve the highest goal happily
2. To become a person with stable mind, pleasing personality and determination
3. To awaken wisdom in students

**UNIT - I**
Neetisatakam-Holistic development of personality ,Verses- 19,20,21,22 (wisdom), Verses- 29,31,32 (pride & heroism) ,Verses- 26,28,63,65 (virtue) ,Verses- 52,53,59 (dont's) ,Verses- 71,73,75,78 (do's)

**UNIT - II**
• Approach to day to day work and duties

• Shrimad BhagwadGeeta : Chapter 2-Verses 41, 47,48,

• Chapter 3-Verses 13, 21, 27, 35, Chapter 6-Verses 5,13,17, 23, 35,

• Chapter 18-Verses 45, 46, 48.

**UNIT - III**
• Statements of basic knowledge.

• Shrimad BhagwadGeeta: Chapter2-Verses 56, 62, 68

• Chapter 12 -Verses 13, 14, 15, 16,17, 18

• Personality of Role model. Shrimad BhagwadGeeta:Chapter2-Verses 17, Chapter 3-Verses 36,37,42,

• Chapter 4-Verses 18, 38,39 , Chapter18 – Verses 37,38,63

**Suggested reading**
1. "Srimad Bhagavad Gita" by Swami SwarupanandaAdvaita Ashram (Publication Department), Kolkata
2. Bhartrihari's Three Satakam (Niti-sringar-vairagya) by P.Gopinath, 4. Rashtriya Sanskrit Sansthanam, New Delhi.

**MASTER OF COMPUTER APPLICATIONS**
**I Year I Semester**

**MC19C** **VALUE EDUCATION**
**(AUDIT-1)**

**Course Objectives**

Students will be able to

1. Understand value of education and self- development
2. Imbibe good values in students
3. Let the should know about the importance of character

**UNIT - I**

Values and self-development –Social values and individual attitudes. Work ethics, Indian vision of humanism. Moral and non- moral valuation. Standards and principles. Value judgements

**UNIT - II**

Importance of cultivation of values. Sense of duty. Devotion, Self-reliance. Confidence, Concentration. Truthfulness, Cleanliness. Honesty, Humanity. Power of faith, National Unity. Patriotism.Love for nature,Discipline

**UNIT - III**

Personality and Behavior Development - Soul and Scientific attitude. Positive Thinking. Integrity and discipline. Punctuality, Love and Kindness. Avoid fault Thinking. Free from anger, Dignity of labour. Universal brotherhood and religious tolerance.

**UNIT - IV**

True friendship. Happiness Vs suffering, love for truth. Aware of self-destructive habits. Association and Cooperation. Doing best for saving nature

**UNIT - V**

Character and Competence –Holy books vs Blind faith. Self-management and Good health. Science of reincarnation. Equality, Nonviolence,Humility, Role of Women. All religions and same message. Mind your Mind, Self-control.Honesty, Studying effectively

**TEXTBOOK:**

1. Chakroborty, S.K. “Values and Ethics for organizations Theory and practice”, Oxford University Press, New Delhi

**MASTER OF COMPUTER APPLICATIONS**
**I Year I Semester**

**MC19D** **CONSTITUTION OF INDIA**
**(AUDIT-1)**

**Course Objectives:**
Students will be able to:
1. Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
2. To address the growth of Indian opinion regarding modern Indian intellectuals' constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.
3. To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and its impact on the initial drafting of the Indian Constitution

**UNIT - I**
History of Making of the Indian Constitution: History Drafting Committee, ( Composition & Working) Philosophy of the Indian Constitution:Preamble Salient Features

**UNIT - II**
Contours of Constitutional Rights & Duties: Fundamental Rights Right to Equality Right to Freedom Right against Exploitation Right to Freedom of Religion Cultural and Educational Rights Right to Constitutional Remedies Directive Principles of State Policy Fundamental Duties.

**UNIT - III**
Organs of Governance: Parliament Composition Qualifications and Disqualifications Powers and Functions Executive President Governor Council of Ministers Judiciary, Appointment and Transfer of Judges, Qualifications Powers and Functions

**UNIT - IV**
Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal Corporation. Pachayati raj: Introduction, PRI: ZilaPachayat. Elected officials and their roles, CEO ZilaPachayat: Position and role. Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy

**UNIT - V**
Election Commission: Election Commission: Role and Functioning. Chief Election Commissioner and Election Commissioners. State Election Commission: Role and Functioning. Institute and Bodies for the welfare of SC/ST/OBC and women.

**Suggested reading**
1. The Constitution of India, 1950 (Bare Act), Government Publication.
2. Dr. S. N. Busi, Dr. B. R. Ambedkar framing of Indian Constitution, 1st Edition, 2015.
3. M. P. Jain, Indian Constitution Law, 7th Edn., Lexis Nexis, 2014.
4. D.D. Basu, Introduction to the Constitution of India, Lexis Nexis, 2015

**MASTER OF COMPUTER APPLICATIONS**
**I Year II Semester**

**MC210 PYTHON PROGRAMMING**

**Unit I**

Basic features of Python-Interactive execution, comments, types, variables, operators, expressions, Statements-assignment, input, print, Control flow-Conditionals, Loops, break statement, continue statement, pass statement, Functions, definition, call, scope and lifetime of variables, keyword arguments, default parameter values, variable length arguments, recursive functions, Sequences-Strings ,Lists and Tuples-basic operations and functions, iterating over sequences , Sets and Dictionaries- operations and functions, Python program examples.

**Unit II**

Files-operations-opening, reading, writing, closing,file positions. Exceptions – raising and handling exceptions, try/except statements, finally clause, standard exceptions, custom exceptions. Functional programming-mapping, filtering and reduction, Lambda functions, List comprehensions. Scope, namespaces and modules, import statement, creating own modules, avoiding namespace collisions when importing modules, iterators and generators, Python program examples.

**Unit III**

Object oriented programming- classes, constructors, objects, class variables, class methods, static methods, operator overloading. Inheritance-is-a relationship, composition, polymorphism, overriding, multiple inheritance, abstract classes, multithreaded programming, Python program examples.

**Unit IV**

GUI Programming with Tkinter, Widgets (Buttons, Canvas, Frame, Label, Menu, Entry, Text, Scrollbar, Combobox, Listbox, Scale), event driven programming-events, callbacks, binding, layout management-geometry managers: pack and grid, creating GUI based applications in Python.

**Unit V**

Introduction to Django Framework
Model Template View (MTV) framework, Creating a Project and Application,
Configuring database, Defining a model, Defining a view, Defining a template,
Defining a URL pattern, Enabling Admin site, Designing a RESTful API

TEXT BOOKS

1. Exploring Python, Timothy A. Budd, McGraw Hill Publications.
2. Introduction to Programming using Python, Y.Daniel Liang, Pearson.
3. Python Programming, R.Thareja, Oxford University Press.
4. PythonProgramming, Sheetal Taneja and Naveen Kumar, Pearson.
5. Internet of Things - A hands on approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015.

REFERENCE BOOKS

1. Introduction to Computer Science using Python, Charles Dierbach, Wiley India Edition.
2. Fundamentals of Python, K. A. Lambert, B.L. Juneja, Cengage Learning.
3. Think Python, how to think like a computer scientist, Allen B. Downey,SPD, O'Reilly.
4. www.python.org web site.
5. Official Django Document (https://buildmedia.readthedocs.org/media/pdf/django/1.5.x/django.pdf)

**MASTER OF COMPUTER APPLICATIONS**
**I Year II Semester**
**MC220 DATA STRUCTURES THROUGH JAVA**

**Unit I**

**Java Basics:** Java buzzwords, data types, variables, operators, expressions, **control flow statements-** if and switch statements, **loops-** for, while, do-while, jump statements, classes, objects, class modifiers, class members and access control, methods, simple input and output statements, string handling, **inheritance** and **polymorphism -** super and sub classes, member access rules, method overriding, dynamic method dispatch, abstract classes, Object class, **interfaces-** implementing interfaces, multiple inheritance in interfaces. **packages-** defining, creating and accessing a package, importing packages, file i/o in Java, **exception handling-** concepts of exception handling, types of exceptions, usage of try, catch, throw, throws and finally key words, creating own exception sub classes, **Java Collection frame work-** Commonly used Collection classes and interfaces in Java, java.util and java.io packages.

**Unit II**

Basic concepts-Data types, Abstract Data Types, Data structures, Algorithms, Performance analysis- time complexity and space complexity, Asymptotic Analysis-Big O, Omega and Theta notations.

Introduction to Linear and Non Linear data structures ,Linear data structures- Linear Lists, Sequential and Linked allocation , The list ADT, array and linked Implementations, Singly Linked Lists-Operations-Insertion, Deletion, Doubly Linked Lists- Operations- Insertion, Deletion, Circularly linked lists. Representation of single, two dimensional arrays, Sparse matrices and their representation, Using ArrayList, LinkedList classes, Iterator and ListIterator interfaces in Java.

**Unit III**

Stack ADT, definition, operations, array and linked implementations, applications-infix to postfix conversion, Postfix expression evaluation, recursion implementation, Queue ADT, definitions and operations ,array and linked implementations, Circular queues, Insertion and deletion operations, Deque (Double ended queue)ADT, array and linked implementations, Using ArrayDeque, ArrayList, LinkedList classes, Iterator and ListIterator interfaces, Queue and Deque interfaces in Java.

**Unit-IV**

Searching- Linear Search, Binary Search, Hashing- Introduction, hash tables, hash functions, collision resolution methods, Using HashSet, HashMap classes in Java, Comparison of Searching methods.

Sorting- Bubble Sort, Insertion Sort, Selection Sort, Radix Sort, Quick sort, Merge sort, Heap Sort, Comparison of Sorting methods.

**Unit-V**

Non Linear data structures- Trees – Basic Terminology, Binary tree ADT, array and linked representations, traversals, Priority Queue ADT- implementation, Heaps.

Graphs – Basic Terminology, Graph Representations- Adjacency matrix, Adjacency lists, Graph traversals- DFS and BFS

Search Trees-Binary Search Trees, Definition, ADT, Implementation, Operations- Searching, Insertion and Deletion, Balanced search trees-(Elementary treatment-only Definitions and Examples)AVL Trees, B-Trees, Red-Black trees, Comparison of Search Trees, Tries (examples only), Knuth-Morris-Pratt Pattern matching algorithm.

**TEXT BOOKS**

1. Data Structures, Algorithms and Applications in Java, 2nd Edition, S.Sahani, Universities Press.
2. Data Structures And Algorithms in Java, 2nd edition, M.T.Goodrich and R.Tamassia, John Wiley and Sons, Inc.
3. Data Structures and Algorithms in Java, 2nd edition, A.Drozdek, Cengage Learning
4. Data Structures with Java, J.R.Hubbard , 2nd edition, TMH.
5. Java: The Complete Reference , 10th Edition, Herbert Schildt, McGrawHill Education, Indian Edition.(For Unit I in the Syllabus)

**REFERENCE BOOKS**

1. Data Structures and Java Collections Frame Work, W.J.Collins, McGraw Hill.
2. Data Structures Using Java, Yedidyah Langsam, Moshe Augenstein, Aaron M. Tenenbaum, Pearson Education.
3. Data Structures and Software Development in an Object Oriented Domain, Java Edition, J.P.Tremblay and G.A.Cheston, Pearson Education
4. Data Structures and algorithms in Java, 2nd Edition, R.Lafore, Pearson Education.
5. Data Structures using Java, D.S.Malik and P.S. Nair,Cengage Learning.
6. Data Structures and algorithm analysis in Java, Mark Allen Weiss, Pearson education

**MASTER OF COMPUTER APPLICATIONS**
**I Year II Semester**

## MC230 DESIGN AND ANALYSIS OF ALGORITHMS

**UNIT I**

Introduction: Algorithm, Pseudo code for expressing algorithms, Performance Analysis-Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, Probabilistic analysis, Amortized complexity.
Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort, Strassen's Matrix Multiplication.

**UNIT II**

Greedy method: General method, applications-Job sequencing with dead lines, 0/1 knapsack problem, Disjoint set operations, union and find algorithms, Spanning trees, Minimum cost spanning trees, Prim's and Kruskal's algorithms, Single source shortest path problem.

**UNIT III**

Dynamic Programming: General method, applications-Multistage graphs, Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Traveling sales person problem, Reliability design.

**UNIT IV**

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

**UNIT V**

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

**TEXT BOOKS:**

1. Fundamentals of Computer Algorithms, 2nd Edition, Ellis Horowitz, Satraj Sahni and S.Rajasekharan, Universities Press,2008.
2. Foundations of Algorithms,4$^{th}$ edition,R.Neapolitan and K.Naimipour,Jones and Bartlett Learning.
3. Design and Analysis of Algorithms, P.H.Dave, H.B.Dave, Pearson Education,2008.

**REFERENCE BOOKS:**

1. Computer Algorithms, Introduction to Design and Analysis, 3rd Edition, Sara Baase, Allen, Van, Gelder, Pearson Education.
2. Algorithm Design: Foundations, Analysis and Internet examples, M.T.Goodrich and R.Tomassia, John Wiley and sons.

**MASTER OF COMPUTER APPLICATIONS**

**I Year II Semester**

**MC240 COMPUTER NETWORKS**

**UNIT I**

Introduction to Networks, internet, protocols and standards, the OSI model, layers in OSI model, TCP/IP suite, Addressing, Analog and digital signals.
Physical Layer: digital transmission, multiplexing, transmission media, circuit switched networks, Datagram networks, virtual circuit networks, switch and Telephone network.

**UNIT II**

Data link layer: Introduction, Block coding, cyclic codes, checksum, framing, flow and error control, Noiseless channels, noisy channels, HDLC, point to point protocols
Medium Access sub layer: Random access, controlled access, channelization, IEEE standards, Ethernet, Fast Ethernet, Giga-Bit Ethernet, wireless LANs.

**UNIT III**

Connecting LANs, backbone networks and virtual LANs, Wireless WANs, SONET, frame relay and ATM.
Network Layer: Logical addressing, internetworking, tunneling, address mapping, ICMP, IGMP, forwarding, uni-cast routing protocols, multicast routing protocols.

**UNIT IV**

Transport Layer: Process to process delivery, UDP and TCP protocols, SCTP, data traffic, congestion, congestion control, QoS, integrated services, differentiated services, QoS in switched networks.

**UNIT V**

Application Layer – Domain name space, DNS in internet, electronic mail, FTP, WWW, HTTP, SNMP, multi-media, network security.

**TEXT BOOKS:**

1.Data Communications and Networking , Behrouz A. Forouzan, Fourth Edition TMH.
2.Computer Networks, Andrew S Tanenbaum, 4th Edition. Pearson Education.

**REFERENCE BOOKS:**

1.An Engineering Approach to Computer Networks,S.Keshav,2nd Edition,Pearson Education.

**MASTER OF COMPUTER APPLICATIONS**
**I YEAR II SEMESTER**
**MC250 ACCOUNTING AND FINANCIAL MANAGEMENT**

**UNIT I**

**Introduction to Accounting:** Principles, concepts and conventions, double entry system of accounting, introduction to basic books of accounts, Journal, ledger- Trial Balance - Preparation of Final accounts: Trading Account, Profit and Loss Account and Balance Sheet. Use of spread sheet to solve the above problems.

**UNIT II**

**Financial Management** - Meaning and scope, role of Financial Manager, Objectives of time value of money - Goals of Financial Management, Leverages: Operating, Financial Leverage and Combined Leverage Cost of Capital: Cost of Equity, Preference Shares, Bonds- Weighted Average Cost of Capital – Capital Gearing- Overcapitalization and Undercapitalization, Sources of Finance. Use of spread sheet to solve the above problems.

**Unit III**

**Tools and Techniques for Financial Statement Analysis:** Ratio Analysis – Classification of Ratios – Short term solvency and long term solvency – Profitability ratios - Analysis and Interpretation of Financial Statements through ratios of Liquidity, Solvency and Profitability ratios. **Fund Flow Statement** - Meaning, Importance, Statement of changes in working capital and statement of Sources and application of funds. Cash flow Analysis: cash flow Statements: Preparation, Analysis and interpretation. Use of spread sheet to solve the above problems.

**UNIT IV**

**Break-even Analysis:** Concept of Break Even Point, Cost-Volume-Profit Analysis, Determination of Break Even Point, Margin of Safety and PV ratio, Impact of changes in Cost or selling price on BEP Practical applications of Break-even Analysis.
**Budgeting : B**udgeting – cash budget, sales budget – flexible Budgets and master budgets. Use of spread sheet to solve the above problems.

**Unit V**

**Capital Budgeting**: Capital and its significance, Types of Capital, Estimation of Fixed and Working capital requirements, Methods and sources of raising capital.. Capital Budgeting: features of capital budgeting proposals, Methods of Capital Budgeting: Payback Method, Accounting Rate of Return (ARR) and Net Present Value Method (simple problems). Use of spread sheet to solve the above problems.

TEXT BOOKS:

1. Aryasri: Accounting And Financial Management,, TMH, 2009
2. Van Horne, James, C: Financial Management and Policy, Pearson, 2009

REFERENCE BOOKS:


1. Dr. G. Vidyanath G. Lakshmi, Accounting and Financial Management.
2. Prasanna Chandra, Financial Management, TMH, 2009
3. S.N.Maheshwari, Financial Accounting, Sultan Chand, 2009.
4. Tulsian, Financial Accounting, S Chand, 2009.
5. Khan and Jain: Financial Management, TMH, 2009
6. Gokul Sinha: Financial Statement Analysis, PHI, 2009
7. Bhat Sundhindra:Financial Management, Excel:2009
8. Jawaharlal:Accounting for Management, Himalaya, 2009
9. Paresh Shah : Basic Financial Accounting for Management, Oxford 2010.
10. K. Scott Proctor, Building Financial Models with Microsoft Excel, 2nd edition, Wiley Finance.

**MASTER OF COMPUTER APPLICATIONS**

**I Year II Semester**

**MC260 PYTHON PROGRAMMING LAB**

**Note: The problems given below are only sample problems.**

1. Write a Python program that reads a list of names and ages, then prints the list sorted by age.
2. Write a Python program that will prompt the user for a file name, read all the lines from the file into a list, sort the list, and then print the lines in sorted order.
3. Write a Python program that asks the user for a file name, and then prints the number of characters, words, and lines in the file.
4. Write a Python program that will prompt the user for a string and a file name, and then print all lines in the file that contain the string.
5. Create a class Rectangle. The constructor for this class should take two numeric arguments, which are the length and breadth. Add methods to compute the area and perimeter of the rectangle, as well as methods that simply return the length and breadth. Add a method isSquare that returns a Boolean value if the Rectangle is a Square.
6. Write a class Complex for performing arithmetic with complex numbers. The constructor for this class should take two floating-point values. Add methods for adding, subtracting, and multiplying two complex numbers.
7. Write a Python program that converts a fully parenthesized arithmetic expression from infix to postfix.
8. Write a Python program that reads a postfix expression from standard input, evaluates it, and writes the value to standard output.
9. Write a Python program that implements binary search method to search for a key in a sorted list.

**Sample problems covering data structures:**

10.Write Python program to implement the List ADT using a linked list.

11.Write Python programs to implement the deque (double ended queue) ADT using

a) Array b) Singly linked list c) Doubly linked list.

12.Write a Python program that counts the occurrences of words in a text file and

displays the words in decreasing order of their occurrence counts.

13.Write a Python program that implements insertion sort for sorting a list of elements

in ascending order.

**GUI applications:**

1. Write a Python program that works as a simple calculator. Use a grid to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result.

2. Develop a Python GUI application that receives an integer in one text field, and computes its factorial Value and fills it in another text field, when the button named "Compute" is clicked.

3. Write a Python program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer Num2 is Zero, the program should Display an appropriate message in the result field in Red color.

4. Write a Python program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time.No light is on when the program starts.

**Advanced Problems:**

1. Create a table in MySQL that stores the status of devices in a house with the following data (Device ID, Device Name and Device State, last altered date and time). Now write a Python program that reads and alters the state of a given device. The date format is "YYYY-MM-DD:HH-mm-ss" where mm is minutes and ss is seconds.

2. Write a Python program that loads all the states of the devices into a dictionary from the table mentioned above.

3. Write a Python program that sorts the device states based on the last altered time

4. Write a Python program that reads a string from keyboard and prints the count of each alphabet in the string.

5. Write a Python program that reads a page from internet and prints it on the screen.

6. Write a Python program that reads and modifies an XML file

7. Write a Python program that reads and alters JSON data from a database table

8. Write a client-server Python program that uses socket connection to implement a time server. The client will connect to the server and the server sends the current time as "YYYY-MM-DD:HH-mm-ss" format. This value should be printed on the client side.

9. Write a Python program that generates 10 random numbers and stores them in a text file one per line. Now write another Python program that reads this data into a list and shows them

10. Write a program that reads key-value pair data from a file and stores them in a database table

11. Write a Python program that reads a time string in the format of "YYYY-MM-DD:HH-mm-ss" and prints its components separately.

12. Write a Python program that reads data from a table and writes it to a text file using tab as field separator and new line as record separator and vice versa.

13. Create a Django web application for a simple calculator with basic operations (+, -, * and /) with two numbers.

14. Create a Django web application that implements Library MIS, which has the features like

- Add/Delete a book
- Issue a book to a person
- Collect a book from a person
- Search for a title or author

15. Create a Django web application that implements a bus reservation system, where a new bus can be added/removed with a given source and destination. A user should be able to reserve or cancel a seat.

TEXT BOOKS

1. Exploring Python, Timothy A. Budd, McGraw Hill Publications.
2. Introduction to Programming using Python, Y.Daniel Liang, Pearson.
3. Python Programming, R.Thareja, Oxford University Press.
4. Python Programming, Sheetal Taneja and Naveen Kumar, Pearson.
5. Internet of Things - A hands on approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015.
6. www.python.org web site.
7. Official Django Document (https://buildmedia.readthedocs.org/media/pdf/django/1.5.x/django.pdf)

**MASTER OF COMPUTER APPLICATIONS**
**I Year II Semester**
**MC270 DATA STRUCTURES THROUGH JAVA LAB**

**List of Sample Problems/Experiments:**

1. Write a Java program that prints all real solutions to the quadratic equation ax2 + bx + c = 0. Read in a, b, c and use the quadratic formula. If the discriminant b2 -4ac is negative; display a message stating that there are no real solutions.
2. The Fibonacci sequence is defined by the following rule. The fist two values in the sequence are 1 and 1. Every subsequent value is the run of the two values preceding it. Write a Java program that uses both recursive and non-recursive functions to print the nth value in the Fibonacci sequence.
3. Write a Java program that prompts the user for an integer and then prints out all prime numbers up to that. Integer.
4. Write a Java program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome.
5. Write a Java program for sorting a given list of names in ascending order.
6. Write a Java program to multiply two given matrices.
7. Write a Java Program that reads a line of integers, and then displays each integer, and the sum of all the integers (use StringTokenizer class)
8. Write a Java program that reads a file name from the user then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.
9. Write a Java program that reads a file and displays a file and displays the file on the screen, with a line number before each line.
10. Write a Java program that displays the number of characters, lines and words in a text file.
11. Write a Java program for creating multiple threads

    a) Using Thread class

b) Using Runnable interface

12. Write a Java program that illustrates how run time polymorphism is achieved.
13. Write a Java program that illustrates the following

    a) Creation of simple package.

    b) Accessing a package.

    c) Implementing interfaces.
14. Write a Java program that illustrates the following

    a) Handling predefined exceptions

    b) Handling user defined exceptions
15. Write Java programs that use both recursive and non-recursive functions for implementing the following searching methods:

    a) Linear search b) Binary search
16. Write Java programs to implement the following using arrays and linked lists.

    a) List ADT
17. Write Java programs to implement the following using arrays.

    a) Stack ADT b) Queue ADT
18. Write a Java program that reads an infix expression, converts the expression to postfix form and then evaluates the postfix expression (use stack ADT).
19. Write a Java program that determines whether parenthetic symbols ( ), { } and < > are nested correctly in a string of characters(use stack ADT).
20. Write a Java program that uses both stack and queue to test whether the given string is a palindrome.
21. Write Java programs to implement the following using a singly linked list.

    a) Stack ADT b)Queue ADT
22. Write Java programs to implement the deque (double ended queue) ADT using

    a) Array b) Singly linked list c) Doubly linked list.

23. Write a Java program to implement priority queue ADT.

24. Write a Java program to perform the following operations:

a) Insert elements into a binary search tree.

b) Search for a key element in a binary search tree.

c) Delete an element from a binary search tree.

25. Write a Java program to implement all the functions of a dictionary (ADT) using Hashing.

26. Write a Java program to implement circular queue ADT using an array.

27. Write Java programs that use recursive and non-recursive functions to traverse the given binary tree in

a) Preorder b) Inorder and c) Postorder.

28. Write Java programs for the implementation of bfs and dfs for a given graph.

29. Write Java programs for implementing the following sorting methods:

a) Bubble sort d) Quick sort g) Radix sort

b) Selection sort e) Merge sort h) Binary tree sort

c) Insertion sort f) Heap sort

* 30. Write a Java program to perform the following operations

a) Insertion into a B-tree b) Deletion from a B-tree

* 31. Write a Java program to perform the following operations

a) Insertion into an AVL-tree b) Deletion from an AVL-tree

* 32. Write a Java program for implementing KMP pattern matching algorithm.

33. Write a Java program that displays node values in a level order traversal (traverse the tree one level at a time, starting at the root node) for a binary tree.

34. Write a Java program that uses recursive functions

a)To create a binary search tree.

b)To count the number of leaf nodes.

c)To copy the above binary search tree.

35. Write a Java program for making frequency count of words in a text.

Note: i)Use java.util package collections framework to implement the problems with numbers from 15 to 23, 25 and 29 in Java.
ii)Use ArrayList, LinkedList, HashSet, HashMap, TreeSet, TreeMap, ArrayDeque, PriorityQueue Classes and Queue, Deque, Iterator, ListIterator and other interfaces.
iii)Problems marked with * symbol are complex. It is not mandatory for the students to solve these problems.

*TEXT BOOKS*

1. Data Structures and algorithms in Java, 2nd Edition, R.Lafore, Pearson Education.
2. Data Structures, Algorithms and Applications in Java, 2nd Edition, S.Sahani, Universities Press.
3. Data Structures using Java, D.S.Malik and P.S. Nair,Cengage Learning.
4. Data Structures and algorithms in Java, 3rd edition, M.T. Goodrich and R.Tamassia, Wiley Student Edition.
5. Java: The Complete Reference, Herbert Schildt,10th edition, Mc GrawHill Education, Indian edition.
6. Data Structures with Java,John R Hubbard, Schaum's outlines,2nd edition,TMH.

**MASTER OF COMPUTER APPLICATIONS**
**I Year II Semester**
**MC280 COMPUTER NETWORKS LAB**

**Sample Problems on Computer Networks (Use C/C++ Programming Language):**

1. Implement the data link layer framing methods such as character stuffing and bit stuffing.
2. Implement the Aloha protocols.
3. Implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP .
4. Implement Dijkstra's algorithm to compute the Shortest path through a graph.
5. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table at each node using distance vector routing algorithm.
6. Take an example subnet of hosts . Obtain broadcast tree for it.
7. Take a 64 bit plain text and encrypt the same using DES algorithm.
8. Write a program to break the above DES coding.
9. Using RSA algorithm, Encrypt text data and Decrypt the same

**MASTER OF COMPUTER APPLICATIONS**
**I Year II Semester**

**MC29A** **DISASTER MANAGEMENT**
**(AUDIT – 2)**

**Course Objectives: -**
Students will be able to:

- learn to demonstrate a critical understanding of key concepts in disaster risk reduction and humanitarian response.
- critically evaluate disaster risk reduction and humanitarian response policy and practice from multiple perspectives.
- develop an understanding of standards of humanitarian response and practical relevance in specific types of disasters and conflict situations.
- critically understand the strengths and weaknesses of disaster management approaches, planning and programming in different countries, particularly their home country or the countries they work in

**UNIT-I**
**Introduction** Disaster: Definition, Factors And Significance; Difference Between Hazard And Disaster; Natural And Manmade Disasters: Difference, Nature, Types And Magnitude.

**UNIT-II**
**Repercussions Of Disasters And Hazards:** Economic Damage, Loss Of Human And Animal Life, Destruction Of Ecosystem. Natural Disasters: Earthquakes, Volcanisms, Cyclones, Tsunamis, Floods, Droughts And Famines, Landslides And Avalanches, Man-made disaster: Nuclear Reactor Meltdown, Industrial Accidents, Oil Slicks And Spills, Outbreaks Of Disease And Epidemics, War And Conflicts.

**UNIT-III**
**Disaster Prone Areas In India** Study Of Seismic Zones; Areas Prone To Floods And Droughts, Landslides And Avalanches; Areas Prone To Cyclonic And Coastal Hazards With Special Reference To Tsunami; Post-Disaster Diseases And Epidemics

**UNIT-IV**
**Disaster Preparedness And Management** Preparedness: Monitoring Of Phenomena Triggering A Disaster Or Hazard; Evaluation Of Risk: Application Of Remote Sensing, Data From Meteorological And Other Agencies, Media Reports: Governmental And Community Preparedness.

**UNIT-V**
**Risk Assessment** Disaster Risk: Concept And Elements, Disaster Risk Reduction, Global And National Disaster Risk Situation. Techniques Of Risk Assessment, Global Co-Operation In Risk Assessment And Warning, People's Participation In Risk Assessment. Strategies for Survival.

**UNIT-VI**
**Disaster Mitigation** Meaning, Concept And Strategies Of Disaster Mitigation, Emerging Trends In Mitigation. Structural Mitigation And Non-Structural Mitigation, Programs Of Disaster Mitigation In India.

**SUGGESTED READINGS:**

1. R. Nishith, Singh AK, “Disaster Management in India: Perspectives, issues and strategies “’New Royal book Company.
2. Sahni, PardeepEt.Al. (Eds.),” Disaster Mitigation Experiences And Reflections”, Prentice Hall Of India, New Delhi.
3. Goel S. L., Disaster Administration And Management Text And Case Studies”,Deep &Deep Publication Pvt. Ltd., New Delhi.

**MASTER OF COMPUTER APPLICATIONS**
**I Year II Semester**

**MC29B** **SOFT SKILLS**
**(AUDIT – 2) - (ACTIVITY - BASED)**

**Course Objectives**

- To improve the fluency of students in English
- To facilitate learning through interaction
- To illustrate the role of skills in real-life situations with case studies, role plays etc.
- To train students in group dynamics, body language and various other activities which boost their confidence levels and help in their overall personality development
- To encourage students develop behavioral skills and personal management skills
- To impart training for empowerment, thereby preparing students to become successful professionals

**Learning Outcomes**

- Developed critical acumen and creative ability besides making them industry- ready.
- Appropriate use of English language while clearly articulating ideas.
- Developing insights into Language and enrich the professional competence of the students.
- Enable students to meet challenges in job and career advancement.

**Unit 1 : INTRODUCTION**

Definition and Introduction to Soft Skills – Hard Skills vs Soft Skills – Significance of Soft/Life/Self Skills – Self and SWOT Analysis ***and***

1. **Exercises on Productivity Development**
   - Effective/ Assertive Communication Skills (Activity based)
   - Time Management (Case Study)
   - Creativity & Critical Thinking (Case Study)
   - Decision Making and Problem Solving (Case Study)
   - Stress Management (Case Study)
2. **Exercises on Personality Development Skills**
   - Self-esteem (Case Study)
   - Positive Thinking (Case Study)
   - Emotional Intelligence (Case Study)
   - Team building and Leadership Skills (Case Study)
   - Conflict Management (Case Study)
3. **Exercises on Presentation Skills**
   - Netiquette

- Importance of Oral Presentation – Defining Purpose- Analyzing the audience- Planning Outline and Preparing the Presentation- Individual & Group Presentation- Graphical Organizers- Tools and Multi-media Visuals
- One Minute Presentations (Warming up)
- PPT on Project Work- Understanding the Nuances of Delivery- Body Language – Closing and Handling Questions – Rubrics for Individual Evaluation (Practice Sessions)

**4. Exercises on Professional Etiquette and Communication**

- Role-Play and Simulation- Introducing oneself and others, Greetings, Apologies, Requests, Agreement & Disagreement….etc.
- Telephone Etiquette
- Active Listening
- Group Discussions (Case study)- Group Discussion as a part of Selection Procedure- Checklist of GDs
- Analysis of Selected Interviews (Objectives of Interview)
- Mock-Interviews (Practice Sessions)
- Job Application and Preparing Resume
- Process Writing (Technical Vocabulary) – Writing a Project Report- Assignments

**5. Exercises on Ethics and Values**

Introduction — Types of Values - Personal, Social and Cultural Values - Importance of Values in Various Contexts

- Significance of Modern and Professional Etiquette – Etiquette (Formal and Informal Situations with Examples)
- Attitude, Good Manners and Work Culture (Live Examples)
- Social Skills - Dealing with the Challenged (Live Examples)
- Professional Responsibility – Adaptability (Live Examples)
- Corporate Expectations

☞ Note: Hand-outs are to be prepared and given to students.

☞ Training plan will be integrated in the syllabus.

☞ Topics mentioned in the syllabus are activity-based.

**SUGGESTED SOFTWARE:**

☞ The following software from **'train2success.com'**

- Preparing for being Interviewed
- Positive Thinking
- Interviewing Skills
- Telephone Skills
- Time Management
- Team Building
- Decision making

**SUGGESTED READING**

1. Alex, K. 2012. *Soft Skills.* S. Chand Publishers
2. *Management Shapers*. 2011. Collection of 28 Books by different Authors. Universities Press.

3. Sherfield, Robert M. *2005. et al Cornerstone: Developing Soft Skills*. Pearson
4. Suresh Kumar,E; Sreehari, P. & Savithri, J. 2011. *Communication Skills and Soft Skills- An Integrated Approach.* New Delhi: Pearson
5. The ACE of Soft Skills by Gopalaswamy Ramesh & Mahadevan Ramesh. 2013. Pearson Publishers. New Delhi.
6. Patnaik, P. 2011. *Group Discussion and Interview Skills.* New Delhi: Foundation
7. Sudhir Andrews. 2009. *How to Succeed at Interviews.* New Delhi: Tata McGraw Hill
8. Sasikumar, V & Dhamija, P.V. 1993. *Spoken English - A Self-Learning Guide to Conversation Practice.* New Delhi: Tata McGraw-Hill
9. *Dixson, Richard J. Everyday Dialogues in English. Prentice Hall India Pvt Ltd*
10. Mukhopadhyay. L *et al. 2012. Polyskills. New Delhi: CUP India Pvt Ltd*
11. Rizvi, M. A. 2005. *Effective Technical Communication.* New Delhi: Tata McGraw Hill
12. *The Hindu Speaks on Education* by the Hindu Newspaper
13. Naterop, B. Jean and Revell, Rod. 2004. *Telephoning in English*. Cambridge: CUP

**MASTER OF COMPUTER APPLICATIONS**
**I Year II Semester**

**MC29C** **STRESS MANAGEMENT BY YOGA**
**(AUDIT – 2)**

Course Objectives
1. To achieve overall health of body and mind
2. To overcome stress

**UNIT - I**
Definitions of Eight parts of yog. ( Ashtanga )

**UNIT - II**
Yam and Niyam.
Do`s and Don't's in life.
a) Ahinsa, satya, astheya, bramhacharya and aparigraha

ii) Shaucha, santosh, tapa, swadhyay, ishwarpranidhan

**UNIT - III**
Asan and Pranayam
b) Various yog poses and their benefits for mind & body

ii)Regularization of breathing techniques and its effects-Types of pranayam

**Suggested Reading:**
1. 'Yogic Asanas for Group Tarining-Part-I" :Janardan Swami Yogabhyasi Mandal, Nagpur
2. "Rajayoga or conquering the Internal Nature" by Swami Vivekananda, AdvaitaAshrama (Publication Department), Kolkata.

**MASTER OF COMPUTER APPLICATIONS**
**I Year II Semester**

**MC29D SANSKRIT FOR TECHNICAL KNOWLEDGE**
**(AUDIT – 2)**

**Course Objectives:**
1. To get a working knowledge in illustrious Sanskrit, the scientific language in the world
2. Learning of Sanskrit to improve brain functioning
3. Learning of Sanskrit to develop the logic in mathematics, science & other subjects
4. enhancing the memory power
5. The engineering scholars equipped with Sanskrit will be able to explore the
6. huge knowledge from ancient literature

**UNIT - I**
Alphabets in Sanskrit,
Past/Present/Future Tense
Simple Sentences

**UNIT - II**
Order
Introduction of roots
Technical information about Sanskrit Literature

**UNIT - III**
Technical concepts of Engineering-Electrical, Mechanical, Architecture, Mathematics

**Suggested Reading:**
1. "Abhyaspustakam" – Dr.Vishwas, Samskrita-Bharti Publication, New Delhi
2. "Teach Yourself Sanskrit" Prathama Deeksha-VempatiKutumbshastri, Rashtriya Sanskrit Sansthanam, New Delhi Publication
3. "India's Glorious Scientific Tradition" Suresh Soni, Ocean books (P) Ltd., New Delhi

**MASTER OF COMPUTER APPLICATIONS**
**II Year I Semester**
**MC310 INTERNET TECHNOLOGIES**

**Unit I:**
HTML:Common tags- List, Tables, Images, Forms, Frames and IFrames, Cascading Style Sheets; Introduction to Java Scripts, Dynamic HTML with Java Script.
XML:Defining XML tags, their attributes and values, Document Type Definition, XML Schemas, Document Object Model, DOM and SAX Parsers, XHTML

**Unit II:**
Introduction to PHP: Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads, Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies

**Unit III:**
Object Oriented Programming with PHP: Creating classes and objects, public, private and protected access, constructor and destructor, Inheritance, Overriding and overloading methods, calling base class methods, static members, interfaces, abstract classes

File Handling: File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories
XML Processing: creating an XML file, using simple XML functions, extracting elements and attributes, modifying XML elements and attributes, adding and deleting elements and attributes, parsing XML file with parser functions .Simple Ajax implementation with PHP

**Unit IV:**
Introduction to Servlets: Lifecycle of a Servlet, The Servlet API, The javax.servlet Package, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions.
Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, implicit objects, Using Beans in JSP Pages. Using Cookies-Session Tracking, Security Issues ,Database Access: Using JDBC to access Database from JSPs and Servlets

**Unit V:**

Introduction to MVC architecture, Anatomy of a simple struts application, struts-config.xml file, Presentation layer with JSP, Struts Controller class, JSP bean, html and logic tag libraries, ActionForms, DynaActionForm, Actions, Forwarding, Error Handling, Database Connection Pooling, validation frame work and examples for simple data types, Internationalization

**TEXT BOOKS**:

1. Web Programming, building internet applications, Chris Bates 2nd edition,

WILEY Dreamtech (UNIT 1)

2. The Complete Reference PHP – Steven Holzner, Tata McGraw-Hill (Unit 2,3)

3. Java Server Pages –Hans Bergsten, SPD O'Reilly (UNITs 3,4,5)

4. The World of Scripting Languages , David Barron,Wiley Publications.

5. Professional Jakarta Struts - James Goodwill, Richard Hightower, Wrox Publishers.

**REFERENCE BOOKS**:

1. Programming world wide web,R.W.Sebesta,Fourth edition,Pearson.
2. Core SERVLETS ANDJAVASERVER PAGES VOLUME 1: CORE TECHNOLOGIES , Marty Hall and Larry Brown Pearson
3. Internet and World Wide Web – How to program , Dietel and Nieto,Pearson.
4. Jakarta Struts Cookbook , Bill Siggelkow, S P D O'Reilly.
5. Professional Java Server Programming,S.Allamaraju and othersApress(dreamtech).
6. Java Server Programming ,Ivan Bayross and others,The X Team,SPD
7. Web Warrior Guide to Web Programmming-Bai/Ekedaw-Thomas
8. Beginning Web Programming-Jon Duckett WROX.
9. Java Server Pages, Pekowsky, Pearson.
10. Java Script,D.Flanagan,O'Reilly,SPD.

**MASTER OF COMPUTER APPLICATIONS**
**II Year I Semester**

**MC320 DATA WAREHOUSING AND DATA MINING**

**Objectives:**

- To understand data mining concepts.
- To learn about various data preprocessing techniques.
- To learn about data warehousing.
- To learn about various data mining functionalities such as association rule mining, clustering, classification and outlier analysis.

**UNIT I**

**Introduction:** Fundamentals of data mining, Data Mining Functionalities, Classification of Data Mining systems, Data Mining Task Primitives, Integration of a Data Mining System with a Database or a Data Warehouse System, Issues in Data Mining.
**Data Preprocessing:** Need for Preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

**UNIT II**

**Data Warehouse and OLAP Technology for Data Mining:** Data Warehouse, Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Usage of Data Warehousing Online Analytical Processing and Mining
**Data Cube Computation**: Efficient Methods for simple Data Cube Computation (Full Cube, Iceberg Cube, Closed Cube and Shell Cube), Discovery Driven exploration of data cubes, Attribute-Oriented Induction for data characterization and its implementation

**UNIT III**

**Mining Frequent Patterns, Associations and Correlations:** Basic Concepts, The Apriori algorithm for finding frequent itemsets using candidate generation, Generating association rules from frequent itemsets, Mining frequent itemsets without candidate generation, Mining various kinds of Association Rules, Correlation Analysis

**UNIT IV**

**Classification and Prediction:** Description and comparison of classification and prediction, preparing data for Classification and Prediction
Classification by Decision Tree Induction, Bayesian Classification, Rule-Based Classification, Classification by Backpropagation
Prediction, linear and non-linear regression, evaluating accuracy of a Classifier or a Predictor

**UNIT V**

**Cluster Analysis:** Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, k-means and k-mediods methods, CLARANS, Agglomerative and divisive hierarchical clustering, chameleon dynamic modeling, DBSCAN, Grid based clustering method: STING, Conceptual Clustering, Constraint-Based Cluster Analysis, Outlier Analysis.

**TEXT BOOKS:**

1. Data Mining – Concepts and Techniques - Jiawei Han, Micheline Kamber and Jian Pei,3rd edition, Morgan Kaufmann Publishers, ELSEVIER.
2. Introduction to Data Mining – Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Pearson education.

**REFERENCES:**

1. Data Warehousing in the Real World – Sam Aanhory & Dennis Murray Pearson Edn Asia.
2. Insight into Data Mining,K.P.Soman,S.Diwakar,V.Ajay,PHI,2008.
3. Data Warehousing Fundamentals – Paulraj Ponnaiah Wiley student Edition
4. The Data Warehouse Life cycle Tool kit – Ralph Kimball Wiley student edition
5. Building the Data Warehouse By William H Inmon, John Wiley & Sons Inc, 2005.
6. Data Mining Introductory and advanced topics –Margaret H Dunham, Pearson education
7. Data Mining Techniques – Arun K Pujari,2nd edition, Universities Press.
8. Data Mining,V.Pudi and P.Radha Krishna,Oxford University Press.
9. Data Mining:Methods and Techniques,A.B.M Shawkat Ali and S.A.Wasimi, Cengage Learning.
10. Data Warehouse 2.0,The Architecture for the next generation of Data Warehousing, W.H.Inmon,D. Strauss,G.Neushloss, Elsevier, Distributed by SPD.

**MASTER OF COMPUTER APPLICATIONS**
**II Year I Semester**
**MC330 SOFTWARE ENGINEERING**

**UNIT I**

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

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.

Process models: The waterfall model, Incremental process models, Evolutionary process models, Specialized process models, The Unified process.

**UNIT II**

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

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

**UNIT III**

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

Design Engineering: Design process and Design quality, Design concepts, the design model, pattern based software design.

Creating an architectural design: software architecture, Data design, Architectural styles and patterns, Architectural Design, mapping data flow into a software architecture.

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.

**UNIT IV**
Introduction to UML: Importance of modeling, object oriented modeling, conceptual model of the UML, Architecture,
Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams. Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams.

**UNIT V**

Basic Behavioral Modeling-I: Interactions, Interaction diagrams.
Basic Behavioral Modeling-II: Use cases, Use case Diagrams, Activity Diagrams.

TEXT BOOKS: Software Engineering

1.Software Engineering :A practitioner's Approach, Roger S Pressman, sixth edition.

McGrawHill International Edition, 2005

2. Software Engineering, Ian Sommerville, seventh edition, Pearson education,2004.

3. Grady Booch, James Rumbaugh, Ivar Jacobson : The Unified Modeling Language User Guide, Pearson Education 2nd Edition
4. Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado: UML 2 Toolkit, WILEY-Dreamtech India Pvt. Ltd.

REFERENCE BOOKS:

1. Software Engineering,A Precise Approach,Pankaj Jalote,Wiley India,2010.
2. Software Engineering : A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
3. Fundamentals of Software Engineering,Rajib Mall,PHI, 2005
4. Software Engineering, Principles and Practices,Deepak Jain,Oxford University Press.
5. Software Engineering1: Abstraction and modeling, Diner Bjorner, Springer International edition, 2006.
6. Software Engineering2: Specification of systems and languages, Diner Bjorner, Springer International edition , 2006.
7. Software Engineering Foundations,Yingxu Wang,Auerbach Publications,2008.
8. Software Engineering Principles and Practice,Hans Van Vliet,3rd edition,John Wiley &Sons Ltd.
9. Software Engineering 3:Domains,Requirements,and Software Design,D.Bjorner,Springer International Edition.
10. Introduction to Software Engineering,R.J.Leach,CRC Press.
11. **Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.**
12. Pascal Roques: Modeling Software Systems Using UML2, WILEY-Dreamtech India Pvt. Ltd.
13. Atul Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.
14. Mark Priestley: Practical Object-Oriented Design with UML,TMH.
15. Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.
16. Object-Oriented Analysis and Design with the Unified Process By John W. Satzinger, Robert B Jackson and Stephen D Burd, Cengage Learning.
17. UML and C++,R.C.Lee, and W.M.Tepfenhart,PHI.
18. Object Oriented Analysis,Design and Implementation,B.Dathan,S.Ramnath,Universities Press.
19. OODesign with UML and Java,K.Barclay,J.Savage,Elsevier.
20. Learning UML 2.0,Russ Miles and Kim Hamilton,O'Reilly,SPD.

**MASTER OF COMPUTER APPLICATIONS**
**II Year I Semester**
**MC341 EMBEDDED SYSTEMS**
**( ELECTIVE – 1 )**

**Learning Objectives:**

• Design embedded computer system hardware

• Design, implement, and debug multi-threaded application software that operates under real-time constraints on embedded computer systems

• Use and describe the implementation of a real-time operating system on an embedded computer system

• Formulate an embedded computer system design problem incljuding multiple constraints, create a design that satisfies the constraints, *implement the design in hardware and software, and measure performance against the design constraints

• Create computer software and hardware implementations that operate according to well-known standards

• Organize and write design documents and project reports

• Organize and make technical presentations that describe a design.

**UNIT I**

**Introduction to Embedded Systems:** Embedded Systems, Processor Embedded into a System, Embedded Hardware Units and Devices in a System, Embedded Software, Complex System Design, Design Process in Embedded System, Formalization of System Design, Classification of Embedded Systems

**UNIT II**

**8051 and Advanced Processor Architecture :** 8051 Architecture, 8051 Micro controller Hardware, Input/Output Ports and Circuits, External Memory, Counter and Timers, Serial data Input/Output, Interrupts, Introduction to Advanced Architectures, Real World Interfacing, Processor and Memory organization **Devices and Communication Buses for Devices Network:** Serial and parallel Devices & ports, Wireless Devices, Timer and Counting Devices, Watchdog Timer, Real Time Clock, Networked Embedded Systems, Internet Enabled Systems, Wireless and Mobile System protocols

**UNIT III**

**Embedded Programming Concepts :** Software programming in Assembly language and High Level Language, Data types, Structures, Modifiers, Loops and Pointers, Macros and Functions, object oriented Programming, Embedded Programming in C++ & JAVA.

**UNIT IV**

**Real – Time Operating Systems :** OS Services, Process and Memory Management, Real – Time Operating Systems, Basic Design Using an RTOS, Task Scheduling Models, Interrupt Latency, Response of Task as Performance Metrics **RTOS Programming :** Basic functions and Types of RTOSES, RTOS VxWorks, Windows CE

**UNIT V**

**Embedded Software Development Process and Tools :** Introduction to Embedded Software Development Process and Tools , Host and Target Machines, Linking and Locating Software, Getting Embedded Software into the Target System, Issues in Hardware-Software Design and Co-Design

**TEXT BOOKS:**

1. Embedded Systems, Raj Kamal, Second Edition TMH.

**REFERENCES:**

1. Embedded/Real-Time Systems, Dr.K.V.K.K.Prasad, dreamTech press
2. The 8051 Microcontroller and Embedded Systems, Muhammad Ali Mazidi, Pearson.
3. Embedded Systems, Shibu K V, Mc Graw Hill.
4. An Embedded Software Primer, David E. Simon, Pearson Education.
5. Micro Controllers, Ajay V Deshmukhi, TMH.
**6. Microcontrollers, Raj Kamal, Pearson Education.**

**MASTER OF COMPUTER APPLICATIONS**
**II Year I Semester**

**MC342 ARTIFICIAL INTELLIGENCE**
**( ELECTIVE – 1 )**

**Objectives:**

- To learn the difference between optimal reasoning Vs human like reasoning
- To understand the notions of state space representation, exhaustive search, heuristic search along with the time and space complexities
- To learn different knowledge representation techniques
- To understand the applications of AI: namely Game Playing, Theorem Proving, Expert Systems, Machine Learning and Natural Language Processing

**UNIT-I**

Introduction: What is AI? Foundations of AI, History of AI, Agents and environments, The nature of the Environment, Problem solving Agents, Problem Formulation, Search Strategies

**UNIT-II**

Knowledge and Reasoning: Knowledge-based Agents, Representation, Reasoning and Logic, Prepositional logic, First-order logic, Using First-order logic, Inference in First-order logic, forward and Backward Chaining

**UNIT-III**

Learning: Learning from observations, Forms of Learning, Inductive Learning, Learning decision trees, why learning works, Learning in Neural and Belief networks

**UNIT-IV**

Practical Natural Language Processing: Practical applications, Efficient parsing, Scaling up the lexicon, Scaling up the Grammar, Ambiguity, Perception, Image formation, Image processing operations for Early vision, Speech recognition and Speech Synthesis

**UNIT-V**

Robotics: Introduction, Tasks, parts, effectors, Sensors, Architectures, Configuration spaces, Navigation and motion planning, Introduction to AI based programming Tools

**TEXT BOOKS**

1. Stuart Russell, Peter Norvig: “Artificial Intelligence: A Modern Approach”,2nd Edition, Pearson Education, 2007

**REFERENCES**

1. Artificial Neural Networks B. Yagna Narayana, PHI
2. Artificial Intelligence , 2nd Edition, E.Rich and K.Knight (TMH).
3. Artificial Intelligence and Expert Systems – Patterson PHI.
4. Expert Systems: Principles and Programming- Fourth Edn, Giarrantana/ Riley, Thomson.
5. PROLOG Programming for Artificial Intelligence. Ivan Bratka- Third Edition – Pearson Education.
6. Neural Networks Simon Haykin PHI

**MASTER OF COMPUTER APPLICATIONS**
**II Year I Semester**

**MC343** **DISTRIBUTED DATABASES**
(E**LECTIVE-1**)

**UNIT I**

Features of Distributed versus Centralized Databases, Principles of Distributed Databases, Levels Of Distribution Transparency, Reference Architecture for Distributed Databases, Types of Data Fragmentation, Integrity Constraints in Distributed Databases, Distributed Database Design

**UNIT II**

Translation of Global Queries to Fragment Queries, Equivalence transformations for Queries, Transforming Global Queries into Fragment Queries, Distributed Grouping and Aggregate Function Evaluation, Parametric Queries.
Optimization of Access Strategies, A Framework for Query Optimization, Join Queries, General Queries

**UNIT III**

The Management of Distributed Transactions, A Framework for Transaction Management, Supporting Atomicity of Distributed Transactions, Concurrency Control for Distributed Transactions, Architectural Aspects of Distributed Transactions
Concurrency Control, Foundation of Distributed Concurrency Control, Distributed Deadlocks, Concurrency Control based on Timestamps, Optimistic Methods for Distributed Concurrency Control.

**UNIT IV**

Reliability, Basic Concepts, Nonblocking Commitment Protocols, Reliability and concurrency Control, Determining a Consistent View of the Network, Detection and Resolution of Inconsistency, Checkpoints and Cold Restart, Distributed Database Administration, Catalog Management in Distributed Databases, Authorization and Protection

**UNIT V**

Architectural Issues, Alternative Client/Server Architectures, Cache Consistency, Object Management, Object Identifier Management, Pointer Swizzling, Object Migration, Distributed Object Storage, Object Query Processing, Object Query Processor Architectures, Query Processing Issues, Query Execution, Transaction Management, Transaction Management in Object DBMSs, Transactions as Objects

Database Integration, Scheme Translation, Scheme Integration, Query Processing Query Processing Layers in Distributed Multi-DBMSs, Query Optimization Issues Transaction Management Transaction and Computation Model, Multidatabase Concurrency Control, Multidatabase Recovery, Object Orientation and Interoperability, Object Management Architecture CORBA and Database interoperability, Distributed Component Object Model, COM/OLE and Database Interoperability, PUSH-Based Technologies

**TEXT BOOKS:**

1. Distributed Databases Principles & Systems, Stefano Ceri, Giuseppe Pelagatti,TMH.
2. Principles of Distributed Database Systems, M. Tamer Ozsu, Patrick Valduriez , Pearson Education, 2nd Edition.

**REFERENCE BOOKS:**

1. Distributed Database Systems,Chanda Ray,Pearson.
2. Distributed Database Management Systems,S.K.Rahimi and Frank.S.Haug, Wiley.

**MASTER OF COMPUTER APPLICATIONS**

**II Year I Semester**

**MC344 E-COMMERCE**

**( ELECTIVE - 1)**

**Objectives:**

Identify the major categories and trends of e-commerce applications.
Identify the essential processes of an e-commerce system.
Identify several factors and web store requirements needed to succeed in e-commerce.
Discuss the benefits and trade-offs of various e-commerce clicks and bricks alternatives.
Understand the main technologies behind e-commerce systems and how these technologies interact.
Discuss the various marketing strategies for an online business.
Define various electronic payment types and associated security risks and the ways to protect against them.

**UNIT - I**

Electronic Commerce-Frame work, anatomy of E-Commerce applications, E-Commerce Consumer applications, E-Commerce organization applications. Consumer Oriented Electronic commerce - Mercantile Process models.

**UNIT - II**

Electronic payment systems - Digital Token-Based, Smart Cards, Credit Cards, Risks in Electronic Payment systems. Inter Organizational Commerce - EDI, EDI Implementation, Value added networks.

**UNIT - III**

Intra Organizational Commerce - work Flow, Automation Customization and internal Commerce, Supply chain Management. Corporate Digital Library - Document Library, digital Document types, corporate Data Warehouses.

**UNIT- IV**

Advertising and Marketing - Information based marketing, Advertising on Internet, on-line marketing process, market research. Consumer Search and Resource Discovery - Information search and Retrieval, Commerce Catalogues, Information Filtering.

**UNIT - V**

Multimedia - key multimedia concepts, Digital Video and electronic Commerce, Desktop video processing, Desktop video conferencing.

**TEXT BOOK:**

1. Frontiers of electronic commerce – Kalakata, Whinston, Pearson.

**REFERENCES BOOKS:**

1. E-Commerce fundamentals and applications Hendry Chan, Raymond Lee, Tharam Dillon, Ellizabeth Chang, John Wiley.
2. E-Commerce, S.Jaiswal – Galgotia.
3. E-Commerce, Efrain Turbon, Jae Lee, David King, H.Michael Chang.
4. Electronic Commerce – Gary P.Schneider – Thomson.
5. E-Commerce – Business, Technology, Society, Kenneth C.Taudon, Carol Guyerico Traver.

**MASTER OF COMPUTER APPLICATIONS**
**II Year I Semester**
**MC351 ANDROID APPLICATION DEVELOPMENT**
**(ELECTIVE – 2)**

**Objectives:**

To demonstrate their understanding of the fundamentals of Android operating systems
To demonstrate their skills of using Android software development tools
To demonstrate their ability to develop software with reasonable complexity on mobile platform
To demonstrate their ability to deploy software to mobile devices
To demonstrate their ability to debug programs running on mobile devices

**UNIT I**

Introduction to Android Operating System:Android OS design and Features – Android development framework, SDK features, Installing and running applications on Eclipse platform, Creating AVDs, Types of Android applications, Best practices in Android programming, Android tools
Android application components – Android Manifest file, Externalizing resources like values, themes, layouts, Menus etc, Resources for different devices and languages, Runtime Configuration Changes
Android Application Lifecycle – Activities, Activity lifecycle, activity states, monitoring state changes

**UNIT II**

Android User Interface:Measurements – Device and pixel density independent measuring units Layouts – Linear, Relative, Grid and Table Layouts
User Interface (UI) Components – Editable and non editable TextViews, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers
Event Handling – Handling clicks or changes of various UI components
Fragments – Creating fragments, Lifecycle of fragments, Fragment states, Adding fragments to Activity, adding, removing and replacing fragments with fragment transactions, interfacing between fragments and Activities, Multi-screen Activities

**UNIT III**

Intents and Broadcasts:Intent – Using intents to launch Activities, Explicitly starting new Activity, Implicit Intents, Passing data to Intents, Getting results from Activities, Native Actions, using Intent to dial a number or to send SMS
Broadcast Receivers – Using Intent filters to service implicit Intents, Resolving Intent filters, finding and using Intents received within an Activity
Notifications – Creating and Displaying notifications, Displaying Toasts.

**UNIT IV**

Persistent Storage:Files – Using application specific folders and files, creating files, reading data from files, listing contents of a directory Shared Preferences – Creating shared preferences, saving and retrieving data using Shared Preference
Database – Introduction to SQLite database, creating and opening a database, creating tables, inserting retrieving and deleting data, Registering Content Providers, Using content Providers (insert, delete, retrieve and update)

**UNIT V**

**Advanced Topics:** Alarms – Creating and using alarms
Using Internet Resources – Connecting to internet resource, using download manager
Location Based Services – Finding Current Location and showing location on the Map, updating location

**Text Books:**

1. Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox) , 2012
2. Android Application Development for Java Programmers, James C Sheusi, Cengage Learning, 2013

**References:**

1. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013.

**MASTER OF COMPUTER APPLICATIONS**
**II Year I Semester**
**MC352 INFORMATION EXTRACTION AND INFORMATION RETRIEVAL SYSTEMS**
**(ELECTIVE – 2)**

**UNIT I**

Introduction to Information Retrieval Systems : Definition of Information Retrieval System, Objectives of Information Retrieval System, Functional Overview, Relationship to Database Management Systems, Digital Libraries and Data Warehouses; Boolean retrieval. The term vocabulary and postings lists. Dictionaries and tolerant retrieval. Index construction. Index compression.

**UNIT II**

Scoring, term weighting and the vector space model. Computing scores in a complete search system. Evaluation in information retrieval. Relevance feedback and query expansion.

**UNIT III**

XML retrieval. Probabilistic information retrieval. Language models for information retrieval. Text classification. Vector space classification.

**UNIT IV**

Support vector machines and machine learning on documents. Flat clustering. Hierarchical clustering. Matrix decompositions and latent semantic indexing.

**UNIT V**

Web search basics. Web crawling and indexes. Link analysis.

**TEXT BOOKS:**

1. Introduction to Information Retrieval , Christopher D. Manning and Prabhakar Raghavan and Hinrich Schütze, Cambridge University Press, 2008.
2. Information Storage and Retrieval Systems: Theory and Implementation, Kowalski, Gerald, Mark T Maybury,Springer.

**REFERENCE BOOKS :**

1. Modern Information Retrieval , Ricardo Baeza-Yates, Pearson Education, 2007.
2. Information Retrieval: Algorithms and Heuristics, David A Grossman and Ophir Frieder, 2nd Edition, Springer, 2004.
3. Information Retrieval Data Structures and Algorithms, William B Frakes, Ricardo Baeza-Yates, Pearson Education, 1992.
4. Information Storage & Retrieval , Robert Korfhage , John Wiley & Sons.

**MASTER OF COMPUTER APPLICATIONS**

**II Year I Semester**

**MC353 INFORMATION SECURITY**

**(ELECTIVE – 2)**

**Objectives:**

- Understand the basic categories of threats to computers and networks
- Understand various cryptographic algorithms.
- Describe public-key cryptosystem.
- Describe the enhancements made to IPv4 by IPSec
- Understand Intrusions and intrusion detection
- Discuss the fundamental ideas of public-key cryptography.
- Generate and distribute a PGP key pair and use the PGP package to send an encrypted e-mail message.
- Discuss Web security and Firewalls

**UNIT – I**

**Attacks on Computers and Computer Security:** Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security **Cryptography: Concepts and Techniques:** Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, steganography, key range and key size, possible types of attacks.

**UNIT – II**

**Symmetric key Ciphers:** Block Cipher principles & Algorithms(DES, AES,Blowfish), Differential and Linear Cryptanalysis, Block cipher modes of operation, Stream ciphers, RC4,Location and placement of encryption function, Key distribution **Asymmetric key Ciphers:** Principles of public key cryptosystems, Algorithms(RSA, Diffie-Hellman,ECC), Key Distribution

**UNIT – III**

**Message Authentication Algorithms and Hash Functions:** Authentication requirements, Functions, Message authentication codes, Hash Functions, Secure hash algorithm, Whirlpool, HMAC, CMAC, Digital signatures, knapsack algorithm.

**UNIT – IV**

**E-Mail Security:** Pretty Good Privacy, S/MIME **IP Security:** IP Security overview, IP Security architecture, Authentication Header, Encapsulating security payload, combining security associations, key management

**UNIT – V**

**Web Security:** Web security considerations, Secure Socket Layer and Transport Layer Security, Secure electronic transaction **Intruders, Virus and Firewalls:** Intruders, Intrusion detection, password management, Virus and related threats, Countermeasures, Firewall design principles, Types of firewalls

**Case Studies on Cryptography and security:** Secure Inter-branch Payment Transactions, Cross site Scripting Vulnerability, Virtual Elections.

**TEXT BOOKS:**

1. Cryptography and Network Security : William Stallings, Pearson Education,5th Edition
2. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 2nd Edition.
3. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning

**REFERENCES:**

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.
2. Cryptography and Network Security : Forouzan Mukhopadhyay, Mc Graw Hill, 2nd Edition
3. Information Security, Principles and Practice : Mark Stamp, Wiley India.
4. Principles of Computer Sceurity: WM.Arthur Conklin, Greg White, TMH
5. Introduction to Network Security: Neal Krawetz, CENGAGE Learning.
6. Principles of Information security by Michael E Whitman and Herbert J.Mattord.

**MASTER OF COMPUTER APPLICATIONS**

**II Year I Semester**

**MC354 MOBILE COMPUTING**

**(ELECTIVE – 2)**

**UNIT I**

**Mobile Computing:** Novel Applications, Limitations of Mobile Computing, **Mobile Computing Architecture**: Programming languages, Functions of Operating Systems, Functions of Middleware for mobile Systems, Mobile Computing Architectural layers, Protocols, Layers.

**UNIT II**

**Mobile Devices:** Handheld Mobile Smartphones with Multimedia Functionalities, Smartcards, Smart Sensors,**Mobile System Networks**: Cellular Network, WLAN Network and Mobile IP, Ad-hoc Networks, **Mobility Management**

**UNIT III**

**Global System For Mobile Communications (Gsm):** Mobile Services, System Architecture, Protocols, Localization & Calling, Handover, Security. **GPRS:** GPRS System Architecture, **UMTS**: UMTS System Architecture. **LTE**: Long Term Evolution

**UNIT IV**

**Mobile Network Layer:** Mobile IP: Goals, Assumptions, Entities and Terminology, IP Packet Delivery, Agent Discovery, Registration, Tunneling and Encapsulation, Optimizations, Dynamic Host Configuration Protocol (DHCP) **Mobile Transport Layer:** Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit/fast recovery, Transmission /time-out freezing, Selective retransmission, Transaction oriented TCP, TCP over 2.5G/3G Wireless Networks.

**UNIT V**

**Security Issues in Mobile Computing:** Introduction, Information Security, Security Techniques and Algorithms, Security Protocols, Security Models, Security Frameworks for mobile Environment

**TEXT BOOKS:**

1. Raj Kamal, “Mobile Computing”, OXFORD UNIVERSITY PRESS.
2. Asoke K Talukder, et al, “Mobile Computing”, Tata McGraw Hill, 2008.

**REFERENCES:**

1. Jochen Schiller, “Mobile Communications”, Pearson Education, Second Edition, 2008.
2. Dr. Sunilkumar, et al “Wireless and Mobile Networks: Concepts and Protocols”, Wiley India.
3. Matthew S.Gast, “802.11 Wireless Networks”, SPD O’REILLY.
4. Ivan Stojmenovic , “Handbook of Wireless Networks and Mobile Computing”, Wiley, 2007.
5. Kumkum Garg, “Mobile Computing”, Pearson.
6. Handbook of Security of Networks, Yang Xiao, Frank H Li, Hui Chen, World Scientific, 2011.

**MASTER OF COMPUTER APPLICATIONS**
**II Year I Semester**
**MC360 INTERNET TECHNOLOGIES LAB**

**List of Sample Problems**

1. Develop static pages (using Only HTML) of an online Book store. The pages should resemble: www.amazon.com The website should consist the following pages.
   Home page, Registration and user Login
   User Profile Page, Books catalog
   Shopping Cart, Payment By credit card
   Order Conformation
2. Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.
3. Create and save an XML document at the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.
4. Bean Assignments
   a. Create a JavaBean which gives the exchange value of INR(Indian Rupees) into equivalent American/Canadian/Australian Dollar value.
   b. Create a simple Bean with a label - which is the count of number of clicks. Than create a BeanInfo class such that only the "count" property is visible in the Property Window.
   c. Create two Beans-a)KeyPad .b)DisplayPad .After that integrate the two Beans to make it work as a Calculator.
   d. Create two Beans Traffic Light(Implemented as a Label with only three background colours-Red,Green,Yellow) and Automobile(Implemented as a TextBox which states its state/movement). The state of the Automobile should depend on the following Light Transition Table.

| Light Transition | Automobile State |
|---|---|
| Red ---> Yellow | Ready |
| Yellow ---> Green | Move |
| Green --> Red | Stopped |

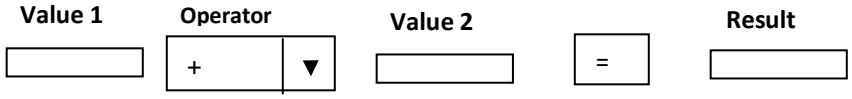
5. Install TOMCAT web server. Convert the static web pages of assignments 2 into dynamic web pages using Servlets and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml. Each user should have a separate Shopping Cart.

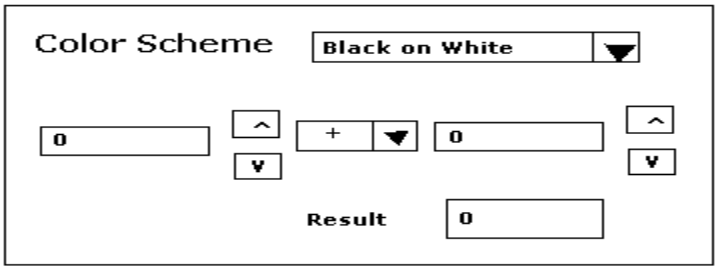
6. Redo the previous task using JSP by converting the static web pages of assignments 2 into dynamic web pages. Create a database with user information and books information. The books catalogue should be dynamically loaded from the database. Follow the MVC architecture while doing the website.

7. Implement the "Hello World!" program using JSP Struts Framework.

8.Redo the problem 5 using PHP.

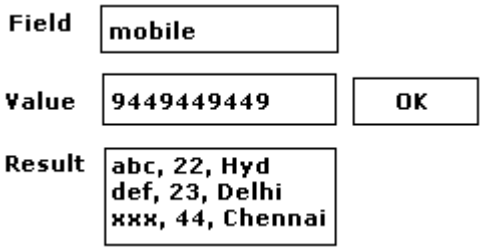
| **Additional Assignment Problems for the IT Lab.:** |
|---|
| Write an HTML page including any required Javascript that takes a number from one text field in |

| |
|---|
| the range of 0 to 999 and shows it in another text field in words. If the number is out of range, it should show “out of range” and if it is not a number, it should show “not a number” message in the result box. |
| |
| Write a java swing application that takes a text file name as input and counts the characters, words and lines in the file. Words are separated with white space characters and lines are separated with new line character. |
| |
| Write a simple calculator servlet that takes two numbers and an operator (+, -, /, * and %) from an HTML page and returns the result page with the operation performed on the operands. It should check in a database if the same expression is already computed and if so, just return the value from database. Use MySQL or PostgreSQL.(Do the same problem using PHP) |
| |
| Write an HTML page that contains a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (color, bold and font size). |
| |
| Write a servlet that takes name and age from an HTML page. If the age is less than 18, it should send a page with “Hello <name>, you are not authorized to visit this site” message, where <name> should be replaced with the entered name. Otherwise it should send “Welcome <name> to this site” message. (Do the same problem using PHP) |
| |
| Write a calculator program in HTML that performs basic arithmetic operations (+, -, /, * and %). Use CSS to change the foreground and background color of the values, buttons and result display area separately. Validate the input strings using JavaScript regular expressions. Handle any special cases like division with zero reasonably. The screen may look similar to the following:<br>  <br> |
| |
| Write a Java program that creates a calculator GUI, as shown in figure. Extra components may be added for convenience: |




The Color Scheme may be Black on White or Blue on Yellow (selectable) and accordingly all components colors must be changed. The values can be either entered or increased or decreased by a step of 10. The operators are +, -, / and * (selectable). Once any change takes place, the result must be automatically computed by the program.

Write a Java Application that will read an XML file that contains personal information (Name, Mobile Number, age and place. It reads the information using SAX parser. After reading the information, it shows two input Text Fields in a window, one for tag name and the other for value. Once these two values are given, it should list all the records in the XML file that match the value of the given field in a text area (result box). For example, if the two text boxes are entered with "name" and "ABCD" then it should show all the records for which name is "ABCD"? An Illustration is given below that takes a mobile number and lists all the records that have the same mobile number.




Consider the following web application for implementation:

The user is first served a login page which takes user's name and password. After submitting the details the server checks these values against the data from a database and takes the following decisions.

If name and password matches, serves a welcome page with user's full name.

If name matches and password doesn't match, then serves "password mismatch" page

If name is not found in the database, serves a registration page, where users full name, present user name (used to login) and password are collected. Implement this application using:

1. Pure JSP
2. Pure Servlets

3. Struts Framework
4. PHP

Implement a simple arithmetic calculator with +, -, /, *, % and = operations using Struts Framework The number of times the calculator is used should be displayed at the bottom (use session variable).

**MASTER OF COMPUTER APPLICATIONS**
**II Year I Semester**
**MC370 DATA WAREHOUSING AND DATA MINING LAB**

Task 1: Credit Risk Assessment

**Description:**

The business of banks is making loans. Assessing the credit worthiness of an applicant is of crucial importance. You have to develop a system to help a loan officer decide whether the credit of a customer is good, or bad. A bank's business rules regarding loans must consider two opposing factors. On the one hand, a bank wants to make as many loans as possible. Interest on these loans is the banks profit source. On the other hand, a bank cannot afford to make too many bad loans. Too many bad loans could lead to the collapse of the bank. The bank's loan policy must involve a compromise: not too strict, and not too lenient.

To do the assignment, you first and foremost need some knowledge about the world of credit. You can acquire such knowledge in a number of ways.
1. Knowledge Engineering. Find a loan officer who is willing to talk. Interview her and try to represent her knowledge in the form of production rules.
2. Books. Find some training manuals for loan officers or perhaps a suitable textbook on finance. Translate this knowledge from text form to production rule form.
3. Common sense. Imagine yourself as a loan officer and make up reasonable rules which can be used to judge the credit worthiness of a loan applicant.
4. Case histories. Find records of actual cases where competent loan officers correctly judged when, and when not to, approve a loan application.

**The German Credit Data**:
Actual historical credit data is not always easy to come by because of confidentiality rules. Here is one such dataset, consisting of 1000 actual cases collected in Germany. credit dataset (original) Excel spreadsheet version of the German credit data.
In spite of the fact that the data is German, you should probably make use of it for this assignment. (Unless you really can consult a real loan officer !)

A few notes on the German dataset
• DM stands for Deutsche Mark, the unit of currency, worth about 90 cents Canadian (but looks and acts like a quarter).
• owns_telephone. German phone rates are much higher than in Canada so fewer people own telephones.
• foreign_worker. There are millions of these in Germany (many from Turrkey). It is very hard to get German citizenship if you were not born of German parents.
• There are 20 attributes used in judging a loan applicant. The goal is the classify the applicant into one of two categories, good or bad.

**Subtasks : (Turn in your answers to the following tasks)**

1.List all the categorical (or nominal) attributes and the real-valued attributes seperately. (5 marks)

2.What attributes do you think might be crucial in making the credit assessment ? Come up with some simple rules in plain English using your selected attributes. (5 marks)

3. One type of model that you can create is a Decision Tree - train a Decision Tree using the complete dataset as the training data. Report the model obtained after training. (10 marks)

4. Suppose you use your above model trained on the complete dataset, and classify credit good/bad for each of the examples in the dataset. What % of examples can you classify correctly ? (This is also called testing on the training set) Why do you think you cannot get 100 % training accuracy ? (10 marks)

5. Is testing on the training set as you did above a good idea ? Why orWhy not ? (10 marks)

6. One approach for solving the problem encountered in the previous question is using cross-validation ? Describe what is cross-validation briefly. Train a Decistion Tree again using cross-validation and report your results. Does your accuracy increase/decrease ? Why ? (10 marks)

7. Check to see if the data shows a bias against "foreign workers" (attribute 20),or "personal-status" (attribute 9). One way to do this (perhaps rather simple minded) is to remove these attributes from the dataset and see if the decision tree created in those cases is significantly different from the full dataset case which you have already done. To remove an attribute you can use the preprocess tab in Weka's GUI Explorer. Did removing these attributes have any significant effect? Discuss. (10 marks)

8. Another question might be, do you really need to input so many attributes to get good results? Maybe only a few would do. For example, you could try just having attributes 2, 3, 5, 7, 10, 17 (and 21, the class attribute (naturally)). Try out some combinations. (You had removed two attributes in problem 7. Remember to reload the arff data file to get all the attributes initially before you start selecting the ones you want.) (10 marks)

9. Sometimes, the cost of rejecting an applicant who actually has a good credit (case 1) might be higher than accepting an applicant who has bad credit (case 2). Instead of counting the misclassifcations equally in both cases, give a higher cost to the first case (say cost 5) and lower cost to the second case. You can do this by using a cost matrix in Weka. Train your Decision Tree again and report the Decision Tree and cross-validation results. Are they significantly different from results obtained in problem 6 (using equal cost)? (10 marks)

10. Do you think it is a good idea to prefer simple decision trees instead of having long complex decision trees ? How does the complexity of a Decision Tree relate to the bias of the model ? (10 marks)

11. You can make your Decision Trees simpler by pruning the nodes. One approach is to use Reduced Error Pruning - Explain this idea briefly. Try reduced error pruning for training your Decision Trees using cross-validation (you can do this in Weka) and report the Decision Tree you obtain ? Also, report your accuracy using the pruned model. Does your accuracy increase ? (10 marks)

12. (Extra Credit): How can you convert a Decision Trees into "if-then-else rules". Make up your own small Decision Tree consisting of 2-3 levels and convert it into a set of rules. There also exist

different classifiers that output the model in the form of rules - one such classifier in Weka is rules.PART, train this model and report the set of rules obtained. Sometimes just one attribute can be good enough in making the decision, yes, just one ! Can you predict what attribute that might be in this dataset ? OneR classifier uses a single attribute to make decisions (it chooses the attribute based on minimum error). Report the rule obtained by training a one R classifier. Rank the performance of j48, PART and oneR. (10 marks)

**Task Resources:**

- Mentor lecture on Decision Trees
- Andrew Moore's Data Mining Tutorials (See tutorials on Decision Trees and Cross Validation)
- Decision Trees (Source: Tan, MSU)
- Tom Mitchell's book slides (See slides on Concept Learning and Decision Trees)
- Weka resources:
    - Introduction to Weka (html version) (download ppt version)
    - Download Weka
    - Weka Tutorial
    - ARFF format
    - Using Weka from command line

Task 2: Hospital Management System

Data Warehouse consists Dimension Table and Fact Table.

REMEMBER The following
Dimension

The dimension object (Dimension):

_ Name
_ Attributes (Levels) , with one primary key
_ Hierarchies

One time dimension is must.

About Levels and Hierarchies

Dimension objects (dimension) consist of a set of levels and a set of hierarchies defined over those levels. The levels represent levels of aggregation. Hierarchies describe parent-child relationships among a set of levels.

For example, a typical calendar dimension could contain five levels. Two hierarchies can be defined on these levels:

H1: YearL > QuarterL > MonthL > WeekL > DayL

H2: YearL > WeekL > DayL

The hierarchies are described from parent to child, so that Year is the parent of Quarter, Quarter the parent of Month, and so forth.

About Unique Key Constraints

When you create a definition for a hierarchy, Warehouse Builder creates an identifier key for each level of the hierarchy and a unique key constraint on the lowest level (Base Level)

Design a Hospital Management system data warehouse (TARGET) consists of Dimensions Patient, Medicine, Supplier, Time. Where measures are ' NO UNITS', UNIT PRICE.

Assume the Relational database (SOURCE) table schemas as follows

TIME (day, month, year),

PATIENT (patient_name, Age, Address, etc.,)

MEDICINE ( Medicine_Brand_name, Drug_name, Supplier, no_units, Uinit_Price, etc.,)

SUPPLIER :( Supplier_name, Medicine_Brand_name, Address, etc., )

If each Dimension has 6 levels, decide the levels and hierarchies, Assume the level names suitably.

Design the Hospital Management system data warehouse using all schemas. Give the example 4-D cube with assumption names.

**MASTER OF COMPUTER APPLICATIONS**
**II Year I Semester**

**MC380 SOFTWARE ENGINEERING LAB**

**Objectives:**

The student should be able to

understand how to analyze a problem

create requirement specifications.

design the software as a solution to the problem.

represent the design using Unified Modeling Language.

Create the views like use case view, logical view, component view, Deployment view,

Design Database

Do forward and Reverse Engineering, and Generation of documentation of the project.

2. Student has to take up another case study of his/her own interest and do the same whatever mentioned in first problem. Some of the ideas regarding case studies are given in reference books which were mentioned in theory syllabus can be referred for some idea.

**Problems on Software Engineering:**

Consider Bank ATM as example, write the SRS and design the system for Cash withdrawal, Balance inquiry and PIN change. Write possible test cases for the applications.

Consider a bus reservation application and develop the SRS and design the system for new reservation and cancellation (Ignore cash payment part). Write the test cases for this application.

Consider a library automation application and develop the SRS and design the system for add new book, issue a book, return a book and search for author/book title. Write the test cases for this application.

**Problems on UML:**

**Unified Modeling Language Lab:**

Students are divided into batches of 5 each and each batch has to draw the following diagrams using UML for an ATM system whose description is given below.

UML diagrams to be developed are:

1. Use Case Diagram.
2. Class Diagram.
3. Sequence Diagram.

4. Collaboration Diagram.
5. State Diagram
6. Activity Diagram.
7. Component Diagram
8. Deployment Diagram.
9. Test Design.

**Description for an ATM System**

The software to be designed will control a simulated automated teller machine (ATM) having a magnetic stripe reader for reading an ATM card, a customer console (keyboard and display) for interaction with the customer, a slot for depositing envelopes, a dispenser for cash (in multiples of Rs. 100, Rs. 500 and Rs. 1000), a printer for printing customer receipts, and a key-operated switch to allow an operator to start or stop the machine. The ATM will communicate with the bank's computer over an appropriate communication link. (The software on the latter is not part of the requirements for this problem.)

The ATM will service one customer at a time. A customer will be required to insert an ATM card and enter a personal identification number (PIN) - both of which will be sent to the bank for validation as part of each transaction. The customer will then be able to perform one or more transactions. The card will be retained in the machine until the customer indicates that he/she desires no further transactions, at which point it will be returned - except as noted below.

The ATM must be able to provide the following services to the customer:

1. A customer must be able to make a cash withdrawal from any suitable account linked to the card, in multiples of Rs. 100 or Rs. 500 or Rs. 1000. Approval must be obtained from the bank before cash is dispensed.
2. A customer must be able to make a deposit to any account linked to the card, consisting of cash and/or checks in an envelope. The customer will enter the amount of the deposit into the ATM, subject to manual verification when the envelope is removed from the machine by an operator. Approval must be obtained from the bank before physically accepting the envelope.
3. A customer must be able to make a transfer of money between any two accounts linked to the card.
4. A customer must be able to make a balance inquiry of any account linked to the card.
5. A customer must be able to abort a transaction in progress by pressing the Cancel key instead of responding to a request from the machine.

The ATM will communicate each transaction to the bank and obtain verification that it was allowed by the bank. Ordinarily, a transaction will be considered complete by the bank once it has been approved. In the case of a deposit, a second message will be sent to the bank indicating that the customer has deposited the envelope. (If the customer fails to deposit the envelope within the timeout period, or presses cancel instead, no second message will be sent to the bank and the deposit will not be credited to the customer.)

If the bank determines that the customer's PIN is invalid, the customer will be required to re-enter the PIN before a transaction can proceed. If the customer is unable to successfully enter the PIN after three tries, the card will be permanently retained by the machine, and the customer will have to contact the bank to get it back.

If a transaction fails for any reason other than an invalid PIN, the ATM will display an explanation of the problem, and will then ask the customer whether he/she wants to do another transaction.

The ATM will provide the customer with a printed receipt for each successful transaction

The ATM will have a key-operated switch that will allow an operator to start and stop the servicing of customers. After turning the switch to the "on" position, the operator will be required to verify and enter the total cash on hand. The machine can only be turned off when it is not servicing a customer. When the switch is moved to the "off" position, the machine will shut down, so that the operator may remove deposit envelopes and reload the machine with cash, blank receipts, etc.

REFERENCE BOOKS

1.Learning UML 2.0,Russ Miles and Kim Hamilton,O'Reilly,SPD.
2.Mastering UML with Rational Rose,W.Boggs&M.Boggs,Wiley India.

**MASTER OF COMPUTER APPLICATIONS**
**II Year II Semester**

**MC411** **INTERNET OF THINGS**
**(ELECTIVE – 3)**

**Objectives:**

- To introduce the terminology, technology and its applications
- To introduce the concept of M2M (machine to machine) with necessary protocols
- To introduce the hardware and working principles of various sensors used for IoT
- To introduce the Python Scripting Language which is used in many IoT devices
- To introduce the Raspberry PI platform, that is widely used in IoT applications
- To introduce the implementation of web based services on IoT devices

**UNIT I**

Introduction to Internet of Things –Definition and Characteristics of IoT, Physical Design of IoT – IoT Protocols, IoT communication models, IoT Communication APIs, IoT enabled Technologies – Wireless Sensor Networks, Cloud Computing, Big data analytics, Communication protocols, Embedded Systems, IoT Levels and Templates, Domain Specific IoTs – Home, City, Environment, Energy, Retail, Logistics, Agriculture, Industry, health and Lifestyle

**UNIT II**

Introduction to Python - Language features of Python, Data types, data structures, Control of flow, functions, modules, packaging, file handling, data/time operations, classes, Exception handling Python packages - JSON, XML, HTTPLib, URLLib, SMTPLib

**UNIT III**

IoT Physical Devices and Endpoints - Introduction to Raspberry Pi- Installation, Interfaces (serial, SPI, I2C),Programming – Python program with Raspberry PI with focus on interfacing external gadgets, controlling output, reading input from pins.

**Unit IV**

Controlling Hardware- Connecting LED, Buzzer, Switching High Power devices with transistors, Controlling AC Power devices with Relays, Controlling servo motor, speed control of DC Motor, Using unipolar and bipolar Stepper motors

Digital input- Sensing push switch, pull-up and pull-down resistors, Rotary encoder, Using keypad, Using RTC

Sensors: Light sensor, temperature sensor with thermistor, voltage sensor, ADC and ADC, Temperature and Humidity Sensor DHT11, Read Switch, Distance Measurement with ultrasound sensor

**UNIT V**

IoT Physical Servers and Cloud Offerings – Introduction to Cloud Storage models and communication APIs Webserver – Web server for IoT, Cloud for IoT, Python web application framework Designing a RESTful web API

**TEXT BOOK:**

1. Internet of Things - A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015, ISBN: 9788173719547
2. Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759
3. Raspberry Pi Cookbook, Software and Hardware Problems and solutions, Simon Monk, O'Reilly (SPD), 2016, ISBN 7989352133895

**MASTER OF COMPUTER APPLICATIONS**
**II Year II Semester**
**MC412 SOFTWARE TESTING AND QUALITY ASSURANCE**
**(ELECTIVE – 3)**

**Objectives:**
The student should be able to:
• To understand software testing and quality assurance as a fundamental component of software life cycle
• To define the scope of SW T&QA projects
• To efficiently perform T&QA activities using modern software tools
• To estimate cost of a T&QA project and manage budgets
• To prepare test plans and schedules for a T&QA project
• To develop T&QA project staffing requirements

**UNIT I**
**Software Quality Assurance and Standards:** The Software Quality challenge, What is Software Quality, Software Quality factors, The components of Software Quality Assurance system, Software Quality Metrics, Costs of Software Quality, Quality Management Standards, Management and its role in Software Quality Assurance, SQA unit and other actors in SQA system. - **(Chapters: 1-4, 21-23, 25, 26) of T3 Quality Standards:** ISO 9000 and Companion ISO Standards, CMM, CMMI, PCMM, Malcom Balridge, 3 Sigma, 6 Sigma and other latest quality standards **(Refer Internet and R11, R12, R13).**

**UNIT II**
**Software Testing Strategy and Environment:** Minimizing Risks, Writing a Policy for Software Testing, Economics of Testing, Testing-an organizational issue, Management Support for Software Testing, Building a Structured Approach to Software Testing, Developing a Test Strategy **Building Software Testing Process:** Software Testing Guidelines, workbench concept, Customizing the Software Testing Process, Process Preparation checklist - **(Chapters: 2,3) of T1 Software Testing Techniques:** Dynamic Testing – Black Box testing techniques, White Box testing techniques, Static testing, Validation Activities, Regression testing **-(Chapters: 4, 5, 6, 7, 8) of T2**

**UNIT III**
**Software Testing Tools:** Selecting and Installing Software Testing tools – **(Chapter 4) of T1.** Automation and Testing Tools **- (Chapter 15) of T2** Load Runner, Win runner and Rational Testing Tools, Silk test, Java Testing Tools, JMetra, JUNIT and Cactus. **(Refer Internet and R9, R10)**

**UNIT IV**
**Testing Process Seven Step Testing Process – I:** Overview of the Software Testing Process, Organizing of Testing, Developing the Test Plan, Verification Testing, Validation Testing. **(Chapters 6, 7, 8, 9, 10) of T1**

**UNIT V**
**Seven Step Testing Process – II:** Analyzing and Reporting Test results, Acceptance and Operational Testing, Post-Implementation Analysis **Specialized Testing Responsibilities**: Software Development Methodologies, Testing Client/Server Systems **(Chapters 12, 13, 14, 15) of T1.**

**TEXT BOOKS:**

1. Effective Methods for Software Testing, Third edition, *William E. Perry*, Wiley India, 2009
2. Software Testing – Principles and Practices, *Naresh Chauhan*, Oxford University Press, 2010.
3. Software Quality Assurance – From Theory to Implementation, *Daniel Galin*, Pearson Education, 2009.

**REFERENCES:**

1. Testing Computer Software, Cem Kaner, Jack Falk, Hung Quoc Nguyen, Wiley India, rp2012.
2. Software Testing – Principles, Techniques and Tools, *M.G.Limaye*, Tata McGraw-Hill, 2009.
3. Software Testing - A Craftsman's approach, *Paul C. Jorgensen*, Third edition, Auerbach Publications, 2010.
4. Foundations of Software Testing, *Aditya P. Mathur*, Pearson Education, 2008.
5. Software Testing and Quality Assurance – Theory and Practice, *Kshirasagar Naik, Priyadashi Tripathy*, Wiley India, 2010.
6. Software Testing, *Ron Patton*, Second edition, Pearson Education, 2006.
7. Software Testing and Analysis – Process, Principles and Techniques, *Mauro Pezze, Michal Young*, Wiley India, 2008.
8. Software Testing Techniques, Boris Beizer, Second edition, Wiley India, 2006
9. Foundations of Software Testing, Dorothy Graham, et al., Cengage learning, 2007, rp 2010.
10. Software Testing - Effective Methods, Tools and Techniques, *Renu Rajani, Pradeep Oak*, Tata McGraw-Hill, rp2011.
11. Software Automation Testing Tools for Beginners, *Rahul Shende*, Shroff Publishers and Distributors, 2012.
12. Software Testing Tools, *K.V.K.K. Prasad*, Dream Tech Press, 2008.
13. Software Testing Concepts and Tools, *Nageswara Rao Pusuluri*, Dream Tech press, 2007.

**MASTER OF COMPUTER APPLICATIONS**
**II Year II Semester**
**MC413 NETWORK PROGRAMMING**
**(ELECTIVE – 3)**

**Objectives:**
- To understand Linux utilities
- To understand file handling, signals
- To understand IPC, network programming in Java
- To understand processes to communicate with each other across a Computer Network.

**UNIT – I**
Linux Utilities- File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking utilities, Filters, Text processing utilities and Backup utilities. Bourne again shell(bash) - Introduction, pipes and redirection, here documents, running a shell script, the shell as a programming language, shell meta characters, file name substitution, shell variables, command substitution, shell commands, the environment, quoting, test command, control structures, arithmetic in shell, shell script examples.Review of C programming concepts-arrays, strings (library functions), pointers, function pointers, structures, unions, libraries in C.

**UNIT - II**
Files- File Concept, File types File System Structure, Inodes, File Attributes, file I/O in C using system calls, kernel support for files, file status information-stat family, file and record locking-lockf and fcntl functions, file permissions- chmod, fchmod, file ownership-chown, lchown , fchown, links-soft links and hard links – symlink, link, unlink. File and Directory management – Directory contents, Scanning Directories- Directory file APIs. Process- Process concept, Kernel support for process, process attributes, process control – process creation, replacing a process image, waiting for a process, process termination, zombie process, orphan process.

**UNIT – III**
Signals- Introduction to signals, Signal generation and handling, Kernel support for signals, Signal function, unreliable signals, reliable signals, kill, raise , alarm, pause, abort, sleep functions.Interprocess Communication - Introduction to IPC mechanisms, Pipes- creation, IPC between related processes using unnamed pipes, FIFOs-creation, IPC between unrelated processes using FIFOs(Named pipes), differences between unnamed and named pipes, popen and pclose library functions, Introduction to message queues, semaphores and shared memory.Message Queues- Kernel support for messages, UNIX system V APIs for messages, client/server example.Semaphores-Kernel support for semaphores, UNIX system V APIs for semaphores.

**UNIT – IV**
Shared Memory- Kernel support for shared memory, UNIX system V APIs for shared memory, client/server example.Network IPC - Introduction to Unix Sockets, IPC over a network, Client-Server model ,Address formats(Unix domain and Internet domain), Socket system calls for Connection Oriented - Communication, Socket system calls for Connectionless-Communication, Example-Client/Server Programs- Single Server-Client connection, Multiple simultaneous clients, Socket options – setsockopt, getsockopt, fcntl.

**UNIT-V**

Network Programming in Java-Network basics, TCP sockets, UDP sockets (datagram sockets), Server programs that can handle one connection at a time and multiple connections (using multithreaded server), Remote Method Invocation (Java RMI)-Basic RMI Process, Implementation details-Client-Server Application.

**TEXT BOOKS:**

1. Unix System Programming using C++, T.Chan, PHI.(Units II,III,IV)
2. Unix Concepts and Applications, 4th Edition, Sumitabha Das, TMH.(Unit I)
3. An Introduction to Network Programming with Java, Jan Graba, Springer, 2010.(Unit V)
4. Unix Network Programming ,W.R. Stevens, PHI.(Units II,III,IV)
5. Java Network Programming,3rd edition, E.R. Harold, SPD, O'Reilly.(Unit V)

**REFERENCES:**

1. Linux System Programming, Robert Love, O'Reilly, SPD.
2. Advanced Programming in the UNIX environment, 2nd Edition, W.R.Stevens, Pearson Education.
3. UNIX for programmers and users, 3rd Edition, Graham Glass, King Ables, Pearson Education.
4. Beginning Linux Programming, 4th Edition, N.Matthew, R.Stones, Wrox, Wiley India Edition.
5. Unix Network Programming The Sockets Networking API, Vol.-I,W.R.Stevens, Bill Fenner, A.M.Rudoff, Pearson Education.
6. Unix Internals, U.Vahalia, Pearson Education.
7. Unix shell Programming, S.G.Kochan and P.Wood, 3rd edition, Pearson Education.
8. C Programming Language, Kernighan and Ritchie, PHI

**MASTER OF COMPUTER APPLICATIONS**

**II Year II Semester**

**MC414** **DATA SCIENCE**

**(ELECTIVE – 3)**

**Objectives:**

To understand about big data, to learn the analytics of Big Data
To understand how data is stored and processed in Hadoop
To learn about NoSQL databases
To learn R tool and understand how data is analyzed using R features
To learn about spark and to understand what features of it are making it to overtake hadoop

**UNIT I**

Types of Digital data: Classification of Digital Data, Introduction to Big Data: What is big data, Evolution of Big Data, Traditional Business Intelligence vs Big Data, Coexistence of Big Data and Data Warehouse. Big Data Analytics: What is Big Data Analytics, What Big Data Analytics Isn't, Why this sudden Hype Around Big Data Analytics, Classification of Analytics, Greatest Challenges that Prevent Business from Capitalizing Big Data, Top Challenges Facing Big Data, Why Big Data Analytics Important, Data Science, Terminologies used in Big Data Environments.

**UNIT II**

Hadoop: Features of Hadoop, Key advantages of hadoop, versions of hadoop, overview of hadoop ecosystem, Hadoop distributions.
Why hadoop? RDBMS vs Hadoop, Distribution computing challenges, History of hadoop, Hadoop overview,HDFS

**UNIT III**

Processing data with hadoop, interfacing with hadoop ecosystem.
NoSQL: Where it is used? What is it? Types of NoSQL Databases, Why NoSQL? Advantages of NoSQL, What we miss with NoSQL? Use of NoSQL in industry,SQL vs NoSQL.

**UNIT IV**

What is R? Why use R for analytics? How to run R? First R example, functions a short programming example, some important R data structures, vectors, matrices, lists, R programming structures.

**UNIT V**

Introduction to Spark, Scala language: values, data types, variables, expressions, conditional expressions, evaluation order, compound expressions, functions, tuple with functions, List, Length, ++, ::, sorted, reverse, sum. slice, mkString, contains, map, filter, leftfold, reduce, Map, Contains, getOrElse, WithDefault, Keys and Values, groupBy, set, mapValues, keys and values, Option(Some and None), Objects, classes, inheritance, traits

TEXT BOOKS:

1. BIG DATA and ANALYTICS, Seema Acharya, Subhashini Chellappan, Wiley publications (Unit I, II, III)
2. BIG DATA, Black Book™, DreamTech Press, 2015 Edition.
2. "The art of R programming" by Norman matloff, 2009.(Unit IV)
3. "Atomic Scala", 2nd edition, Bruce Eckel, Dianne Marsh. (Unit V)

REFERENCE BOOKS:


1. Rajiv Sabherwal, Irma Becerra- Fernandez," Business Intelligence –Practice, Technologies and Management", John Wiley 2011.
2. Lariss T. Moss,ShakuAtre, " Business Intelligence Roadmap", Addison-Wesley It Service.
3. Yuli Vasiliev, " Oracle Business Intelligence : The Condensed Guide to Analysis and Reporting", SPD Shroff, 2012.
4. "Hadoop: The definitive guide", by O'reilly, yahoo press, 2nd edition.
5. "Introduction to R" by Sandeep Rakshit, McGrawHill Education, 2016.

**MASTER OF COMPUTER APPLICATIONS**

**II Year II Semester**

**MC421 MACHINE LEARNING**

**(OPEN ELECTIVE – 1)**

**Objectives:**

- To be able to formulate machine learning problems corresponding to different applications.
- To understand a range of machine learning algorithms along with their strengths and weaknesses.
- To understand the basic theory underlying machine learning.

**UNIT – I**

**Introduction**: An illustrative learning task, and a few approaches to it. What is known from algorithms? Theory, Experiment. Biology. Psychology. Overview of Machine learning, related areas and applications. Linear Regression, Multiple Regression, Logistic Regression,logistic functions

**Concept Learning**: Version spaces. Inductive Bias. Active queries. Mistake bound/ PAC model. basic results. Overview of issues regarding data sources, success criteria.

**UNIT –II**

**Decision Tree Learning**: - Minimum Description Length Principle. Occam's razor. Learning with active queries Introduction to information theory, Decision Trees, Cross Validation and Over fitting.

**Neural Network Learning**: Perceptions and gradient descent back propagation, multilayer networks and back propagation.

**UNIT –III**

**Sample Complexity and Over fitting**: Errors in estimating means. Cross Validation and jackknifing VC dimension. Irrelevant features: Multiplicative rules for weight tuning.

**Support Vector Machines**: functional and geometric margins, optimum margin classifier, constrained optimization, Lagrange multipliers, primal/dual problems, KKT conditions, dual of the optimum margin classifier, soft margins, and kernels.

**Bayesian Approaches**: The basics Expectation Maximization. Bayes theorem, Naïve Bayes Classifier, Markov models,Hidden Markov Models

**UNIT—IV**

**Instance-based Techniques**: Lazy vs. eager generalization. K nearest neighbor, case- based reasoning.**Clustering and Unsupervised Learning:** K-means clustering, Gaussian mixture density estimation, model selection

**UNIT—V**

**Genetic Algorithms**: Different search methods for induction - Explanation-based Learning: using prior knowledge to reduce sample complexity.**Dimensionality reduction:** feature selection, principal component analysis, linear discriminant analysis, factor analysis, independent component analysis, multidimensional scaling, manifold learning

**TEXT BOOKS:**

1. Tom Michel, Machine Learning, McGraw Hill, 1997
2. Trevor Has tie, Robert Tibshirani & Jerome Friedman. The Elements of Statically Learning, Springer Verlag, 2001

**MASTER OF COMPUTER APPLICATIONS**
**II Year II Semester**
**MC422 SOFTWARE PROJECT MANAGEMENT**
**(OPEN ELECTIVE – 1)**

**UNIT I**

Conventional Software Management : The waterfall model, conventional software
Management performance. Evolution of Software Economics : Software Economics, pragmatic software cost estimation.

**UNIT II**

Improving Software Economics : Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.
The old way and the new : The principles of conventional software engineering, principles of modern software management, transitioning to an iterative process.

**UNIT III**

Life cycle phases : Engineering and production stages, inception, Elaboration, construction, transition phases.
Artifacts of the process : The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts. Model based software architectures : A Management perspective and technical perspective.

**UNIT IV**

Work Flows of the process : Software process workflows, Inter trans workflows.Checkpoints of the Process : Major Mile Stones, Minor Milestones, Periodic status assessments. Iterative Process Planning : Work breakdown structures, planning guidelines, cost and schedule estimating, Interaction planning process, Pragmatic planning.
Project Organizations and Responsibilities : Line-of-Business Organizations,
Project Organizations, evolution of Organizations.
Process Automation : Automation Building Blocks, The Project Environment.

**UNIT V**

Project Control and Process instrumentation : The server care Metrics, Management indicators, quality indicators, life cycle expectations pragmatic Software Metrics, Metrics automation. Tailoring the Process : Process dicriminants, Example.
Future Software Project Management : Modern Project Profiles Next generation
Software economics, modern Process transitions.
Case Study : The Command Center Processing and Display System-Replacement(CCPDS-R)

TEXT BOOKS:

1. Software Project Management, Walker Royce, Pearson Education, 1998
2. Software Project Management, Bob Hughes & Mike Cotterell, fourth edition,Tata Mc-Graw Hill,2006

REFERENCE BOOKS:

1. Applied Software Project Management, Andrew Stellman & Jennifer Greene,O'Reilly, 2006
2. Head First PMP, Jennifer Greene & Andrew Stellman, O'Reilly,2007
3. Software Engineering Project Managent, Richard H. Thayer & Edward Yourdon, second edition, Wiley India, 2004.
4. Agile Project Management, Jim Highsmith, Pearson education, 2004.

**MASTER OF COMPUTER APPLICATIONS**
**II Year II Semester**
**MC423 CLOUD COMPUTING**
**(OPEN ELECTIVE – 1)**

**Course Description:**

Cloud computing has evolved as a very important computing model, which enables information, software, and shared resources to be provisioned over the network as services in an on-demand manner. This course provides an insight into what is cloud computing and the various services cloud is capable.

**UNIT I:**

**Computing Paradigms,** High-Performance Computing, Parallel Computing.,Distributed Computing..,ClusterComputing.,Grid Computing, Cloud Computing., Biocomputing, Mobile Computing.,QuantumComputing, Optical Computing. Nanocomputing.

**UNIT II:**

**Cloud Computing Fundamentals**: Motivation for Cloud Computing,The Need for Cloud Computing,Defining Cloud Computing, Definition of Cloud computing,Cloud Computing Is a Service, Cloud Computing Is a Platform, Principles of Cloud computing,Five Essential Characteristics...Four Cloud Deployment Models

.

**UNIT III:**

Cloud Computing Architecture and Management: Cloud architecture, Layer,Anatomy of the Cloud,Network Connectivity in Cloud Computing, Applications, on the Cloud,Managing the Cloud,Managing the Cloud Infrastructure Managing the Cloud application, Migrating Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration.

**UNIT IV:**

**Cloud Service Models:** Infrastructure as a Service,Characteristics of IaaS.Suitability of IaaS,Pros and Cons of IaaS,Summary of IaaS Providers,Platform as a Service,Characteristics of PaaS,Suitability of PaaS,Pros and Cons of PaaS,Summary of PaaS Providers,Software as a Service,Characteristics of SaaS,Suitability of SaaS,Pros and Cons of SaaS,Summary of SaaS Providers.Other Cloud Service Models

**UNIT V:**

**Cloud Service Providers:**EMC,EMC IT,Captiva Cloud Toolkit,Google,Cloud Platform,Cloud Storage, Google Cloud Connect,Google Cloud Print,Google App Engine,Amazon Web Services, Amazon Elastic Compute Cloud,Amazon Simple Storage Service,Amazon Simple Queue ,ervice,Microsoft,Windows Azure, Microsoft Assessment and Planning Toolkit,SharePoint,IBM,Cloud Models,IBM Smart Cloud, SAP Labs,SAP HANA Cloud Platform,Virtualization Services Provided by SAP,Salesforce,Sales Cloud, Service Cloud: Knowledge as a Service,Rackspace,VMware,Manjrasoft,Aneka Platform

**TEXT BOOKS:**

1. **Essentials of cloud Computing : K.Chandrasekhran , CRC press, 2014**

**REFERENCE BOOKS:**

1. Cloud Computing: Principles and Paradigms by Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Wiley, 2011.
2. Distributed and Cloud Computing , Kai Hwang, Geoffery C.Fox, Jack J.Dongarra, Elsevier, 2012.
3. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Kumaraswamy, Shahed Latif, O'Reilly, SPD, rp2011.

**MASTER OF COMPUTER APPLICATIONS**
**II Year II Semester**

**MC424** **COMPUTER FORENSICS**
**(OPEN ELECTIVE – 1)**

**Objectives:**

To understand the cyberspace
To understand the forensics fundamentals
To understand the evidence capturing process.
To understand the preservation of digital evidence.

**UNIT I**

**Computer Forensics Fundamentals:** Introduction to Computer Forensics, Use of Computer Forensics in Law Enforcement, Computer Forensics Assistance to Human Resources/Employment Proceedings, Computer Forensics Services, Benefits of Professional Forensics Methodology, Steps Taken by Computer Forensics Specialists, Who Can Use Computer Forensic Evidence?.**Types of Computer Forensics Technology :** Types of Military Computer Forensic Technology, Types of Law Enforcement Computer Forensic Technology, Types of Business Computer Forensics Technology.

**UNIT II**

**Computer Forensics Evidence and Capture: Data Recovery:** Data Recovery Defined, Data Backup and Recovery, The Role of Backup in Data Recovery, The Data-Recovery Solution, Case Histories.**Evidence Collection and Data Seizure:** Why Collect Evidence?, Collection Options, Obstacles, Types of Evidence, The Rules of Evidence, Volatile Evidence, General Procedure, Collecting and Archiving, Methods of Collection, Artifacts, Collection Steps, Controlling Contamination: The Chain of Custody.

**UNIT III**

**Duplication and Preservation of Digital Evidence:** Preserving the Digital Crime Scene, Computer Evidence Processing Steps, Legal Aspects of Collecting And Preserving Computer Forensic Evidence. **Computer Image Verification and Authentication :** Special Needs of Evidential Authentication, Practical Considerations, Practical Implementation.

**UNIT IV**

**Computer Forensics Analysis: Discovery of Electronic Evidence:** Electronic Document Discovery: A Powerful New Litigation Tool, **Identification of Data:** Timekeeping, Time Matters,Forensic Identification and Analysis of Technical Surveillance Devices. **Reconstructing Past Events**: How to Become a Digital Detective, Useable File Formats, Unusable File Formats, Converting Files. **Networks:** Network Forensics Scenario, A Technical Approach, Destruction of Email, Damaging Computer Evidence, International Principles Against Damaging of Computer Evidence, Tools Needed for Intrusion Response to the Destruction of Data, Incident Reporting and Contact Forms.

**UNIT V**

**Current Computer Forensics Tools:** Evaluating Computer Forensics Tool Needs, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software.

**TEXT BOOKS:**


1. "Computer Forensics : Computer Crime Scene Investigation", JOHN R. VACCA, Firewall Media.
2. "Guide to Computer Forensics and Investigations"4e, Nelson, Phillips Enfinger, Steuart, Cengage Learning.


**REFERENCES:**


1. "Computer Forensics and Cyber Crime", Marjie T Britz, Pearson Education.
2. "Computer Forensics", David Cowen, Mc Graw Hill.
3. Brian Carrier , "File System Forensic Analysis" , Addison Wesley, 2005
4. Dan Farmer & Wietse Venema ,"Forensic Discovery", Addison Wesley, 2005
5. Eoghan Casey , —Digital Evidence and Computer Crime —, Edition 3, Academic Press, 2011
6. Chris Pogue, Cory Altheide, Todd Haverkos ,Unix and Linux Forensic Analysis DVD ToolKit, Syngress Inc. , 2008
7. Harlan Carvey ,Windows Forensic Analysis DVD Toolkit, Edition 2, Syngress Inc. , 2009
8. Harlan Carvey ,Windows Registry Forensics: Advanced Digital Forensic Analysis of the Windows Registry , Syngress Inc, Feb 2011
9. Eoghan Casey, Handbook of Digital Forensics and Investigation, Academic Press, 2009
10. Gonzales/ Woods/ Eddins, Digital Image Processing using MATLAB, 2nd edition, Gatesmark Publishing, ISBN 9780982085400
11. N.Efford, Digital Image Processing, Addison Wesley 2000, ISBN 0-201-59623-7
12. M Sonka, V Hlavac and R Boyle, Image Processing, Analysis and Machine Vision, PWS
13. 1999, ISBN 0-534-95393-
14. Pratt.W.K., Digital Image Processing, John Wiley and Sons, New York, 1978

**MASTER OF COMPUTER APPLICATIONS**
**II Year II Semester**
**MC425 ORGANIZATION STRUCTURE AND PERSONNEL MANAGEMENT**
**(OPEN ELECTIVE – 1)**

## Objectives:

One of the main objectives of this course is to provide the students with an in depth understanding of organizations. Organizations are a main tool that modern society applies to meet the challenges of innovation and resolution of complex tasks. To understand how to organize to resolve the challenges organizations meet is a main focus of this course. The course reviews the main contributions to organization research.

This is an introductory course in Organization Theory. In this course we will explore the basic theories and principles around which contemporary organizations are structured in complex, dynamic, uncertain, and competitive environments. The course examines the effects of both the internal and external environment on managerial choices for the structuring of organizations, as well as the implications that stem from these choices.

This course also covers Personnel Management and Communication. In this course we will explore the basic theories of Personnel Management and its functions. The object of the study is to do the analysis of various functions of the organization in functional area of Personnel Management so that the student should be in a position to create data bases and programmes in the content of Personnel Department.

**Outcomes:**

The course will help each student to better:

- Understand the importance of organizational structure and design on internal organizational processes and overall effectiveness.
- Understand the relationships between organization structure and the behavior of those who work in them or otherwise interact with them.
- Recognize the managerial implications of organization design and change and how these are informed by the relevant theories.
- Appreciate the impact of advanced technologies on the strategy and structure of organizations and how to address the changes implied by the adoption of these technologies.
- Acquire the knowledge and skills needed to analyze the design and structure of organizations through a combination of lectures, discussions, and cases.
- Understand the Personnel Functions like position of the personnel department in the organization.
- Understand manpower planning, job description, interviewing techniques, transfers, promotion and its policies.

- Understand the training and development and career planning and Performance Appraisal.
- Obtain and practice effective written and oral business communications skills.

**UNIT I**

Classical Theories of organization : Functional approach, classical theories of organization, division of labour, levels of authority, span of control, authority & responsibility, efficiency of management. Behavioral theories of organization, limitations of formal organization, human relation, group behavior, committee and group making, motivation and morale.

**UNIT II**

Personnel Function: Evaluation, objectives, principles, philosophies and policies, duties & responsibilities of the manager, position of the personnel department in the organization, line and staff relationship & the changing concept of personnel management in India.

**UNIT III**

Manpower planning : Uses benefits problems and limitations, manpower inventory, manpower forecasting, job description, recruitment, Job specification and job selection, interviewing techniques, transfers, promotion and its policies.

Training and development : Objectives and policies planning, organizing the training department, training manager and his job, on and off the job training, techniques, career planning, objectives of performance appraisal.

**UNIT IV**

Strategic management: Objectives, importance policies, concept of core competence capability of organizational learning, strategic levels and planning, business level strategy and functional level, PHASES OF PLANNING,SWOT, develop strategies and prepare strategic plan.

**UNIT V**

Communication : Importance of communication, inter personnel communication barriers of communication, communication in organizations, using communication skills to manage conflicts. Impact of informational technology and fostering effective communication

**TEXT BOOKS:**

1. L.M.Prasad, Principles and Practice of Management, Sultan Chand & Sons.
2. A.R.Aryasri, Organizational Structure and Personnel Management, TMH, 2009

**REFERENCE BOOKS:**

1. Hellriegel, Jackson and Slocum, Edition 9, Management-A competency – Based Approach
2. L.M.Prasad, Human Resource Management.

**MASTER OF COMPUTER APPLICATIONS**
**II Year II Semester**
**MC426 OPERATIONS RESEARCH**
**(OPEN ELECTIVE – 1)**

**UNIT I**

Introduction to Operations Research: Basics definition, scope, objectives, phases, models and limitations of Operations Research. Linear Programming Problem – Formulation of LPP, Graphical solution of LPP. Simplex Method, Artificial variables, big-M method, two-phase method, degeneracy and unbound solutions.

**UNIT II**

Transportation Problem: Formulation, solution, unbalanced Transportation problem. Finding basic feasible solutions – Northwest corner rule, least cost method and Vogel's approximation method. Optimality test: the stepping stone method and MODI method.

Assignment model: Formulation. Hungarian method for optimal solution. Solving unbalanced problem. Traveling salesman problem as assignment problem.

**UNIT III**

Sequencing models: Solution of Sequencing Problem – Processing n Jobs through 2 Machines – Processing n Jobs through 3 Machines – Processing 2 Jobs through m machines – Processing n Jobs through m Machines.

Replacement Models: Replacement of Items that Deteriorate whose maintenance costs increase with time without change in the money value. Replacement of items that fail suddenly: individual replacement policy, group replacement policy.

**UNIT IV**

Dynamic programming: Characteristics of dynamic programming. Dynamic programming approach for Priority Management employment smoothening, Stage Coach/Shortest Path and Reliability problems.

Games Theory: Competitive games, rectangular game, saddle point, minimax (maximin) method of optimal strategies, value of the game. Solution of games with saddle points, dominance principle. Rectangular games without saddle point – mixed strategy for 2 X 2 games.

**UNIT V**

Inventory models: Inventory costs. Models with deterministic demand – model (a) demand rate uniform and production rate infinite, model (b) demand rate non-uniform and production rate infinite, model (c) demand rate uniform and production rate finite.

Queuing Theory: Essential Features of a queuing system. Performance measures of a queuing system. Moddel 1: {(M/M/1) : (∞/FCFS)} Single server, Unlimited Queue model. Model 2: {(M/M/1) : (∞/SIRO)} Single server, Unlimited Queue model. Model III: {(M/M/1) : (N/FCFS)} Single server, Finite Queue model.

**TEXT BOOKS:**

1. J K Sharma., "Operations Research Theory & Applications 4e", Macmillan India Ltd.
2. P. K. Gupta and D. S. Hira, "Operations Research", S. Chand & co., 2007.

**REFERENCE BOOKS:**

1. Pradeep Prabhakar Pai, Operations Research – principles and Practice, Oxford University Press, 2012.
2. A.M. Natarajan, P. Balasubramani, A. Tamilarasi, "Operations Research", Pearson Education.
3. P Sankara Iyer, "Operations Research", Tata McGraw-Hill, 2008.
4. N.V.S. Raju, "Operations Research", HI-TECH, 2002.
5. Col. D. S. Cheema, "Operations Research", Laxmi Publications Ltd., 2005.
6. F.S. Hillier, G.J. Lieberman, "Introduction to Operations Research – 8ed", TMH.
7. H.S. Kasana & K.D. Kumar, "Introductory Operations Research – Theory and applications", Springer, 2003, rp2005.
8. Billy E. Gillett, "Introduction to Operations Research – A Computer-Oriented Algorithmic Approach", Tata McGraw-Hill, 1979, rp2004.
9. A.B.Rao, Operations Research, Jaico .
10. Ravindran,Phillips,Solberg, Operations Research, 2nd edition,Wiley India.
11. W.L.Winston, Operations Research, 4th edition,Cengage Learning.
12. R. Panneerselvam, "Operations Research", PHI-2e, 2006, rp2008.
13. ANITHA H S, "Operations Research", EXEL books, 2011.