J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
(UGC AUTONOMOUS)

Academic Regulations of MCA (Regular/Full Time) Programme
Under Choice Based Credit System (CBCS) (R18)

(Effective for the students admitted into I year from the Academic Year 2018-19 and onwards)

1.0 For Persuing three year program of study in Master of Computer Applications (MCA) offered by JB INSTITUTE OF ENGINEERING & TECHNOLOGY (JBIET), all the rules specified herein after and approved by the Academic council shall be in force and applicable to the students admitted from the academic year 2018-2019 onwards.

2.0 Eligibility for Admissions

2.1 Admission to the MCA programme shall be made subject to eligibility, qualification prescribed by the College from time to time.

2.2 Admission to the post graduate programme shall be made on the basis of either the merit rank or Percentile obtained by the qualified student in the relevant qualifying Examination/ the merit rank obtained by the qualified student in an entrance test conducted by Telangana State Government (ICET) for MCA programme.

2.3 The medium of instructions for MCA Programme will be **ENGLISH** only.

3.0 MCA Programme (Post-graduate Programme in Computer Applications) Structure

3.1 The MCA Programme is of Semester pattern, with Six Semesters consisting of Three 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), with a minimum of 90 instructional days per Semester.

3.2 A student shall be declared eligible for the award of the MCA Degree, if he pursues a course of study in not less than three and not more than six academic years. However, he is permitted to write the examinations for two more years after six academic years of course work, failing which he shall forfeit his seat in MCA programme.

3.3 **UGC / AICTE** specified definitions/descriptions are adopted appropriately for various terms and abbreviations used in these PG academic regulations, as listed below:

3.3.1 Semester Scheme

Each Semester shall have 'Continuous Internal Evaluation (CIE)' and 'Semester End Examination (SEE)'. Choice Based Credit System (CBCS) and Credit Based Semester System (CBSS) are taken as 'references' for the present set of Regulations. The terms 'SUBJECT' and 'COURSE' imply the same meaning here and refer to 'Theory Subject', or 'Lab Course', or 'Comprehensive Viva', or 'Project' as the case may be.

3.3.2 Credit Courses

All subjects/courses are to be registered by the student in a semester to earn credits which shall be assigned to each subject/course in an L: T: P: C (Lecture Periods: Tutorial Periods: Practical Periods: Credits) structure based on the following general pattern:

One credit for one hour/week/semester for theory/lecture (L) courses

One credit for two hours/ week/semester for laboratory/ practical (P) courses or tutorials (T)

Other student activities like study tour, guest lecture, conference/workshop participations, technical paper presentations, and identified mandatory courses, if any, will not carry credits.

3.3.3 Subject Course Classification

All subjects/courses offered for the MCA Degree Programme is broadly classified as follows. The Institute has followed in general the guidelines issued by AICTE/UGC.

S.No.	Broad Course Classification	Course Group/ Category	Course Description
1	Core Courses (CoC)	CC- Core Courses	Includes subjects related to the parent discipline/ department.
		Project Work	MCA Project or PG Project or Major Project.
		Comprehensive Viva-Voce	Viva-voce covering all the PG subjects studied during the course work and related Aspects.
2	Elective Courses (ElE)	PE - Professional Electives	Includes elective subjects related to the parent discipline/department.
		OE - Open Electives	Elective subjects which include inter-disciplinary subjects or subjects in an area outside the parent discipline/department.
Total number of Credits = 148			

4.0 Course Registration

4.1 A 'Faculty Advisor or Counselor' shall be assigned to students, who will advise on the MCA, its Course Structure and Curriculum, Choice/Option for Subjects/ Courses, based on his competence, progress, pre-requisites and interest.

4.2 The Academic Section of the College invites 'Registration Forms' from students within 15 days from the commencement of class work. The Registration Requests for any 'CURRENT SEMESTER' shall be completed BEFORE the commencement of SEEs (Semester End Examinations) of the 'PRECEDING SEMESTER'.

4.3 A Student can apply for Registration, ONLY AFTER obtaining the 'WRITTEN APPROVAL' from his Faculty Advisor, which should be submitted to the College Academic Section through the Head of Department (a copy of it being retained with Head of Department, Faculty Advisor and the Student).

4.4 If the Student 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.

4.5 Subject/ Course Options exercised through Registration are final and CANNOT be changed, nor can they be inter-changed; further, alternate choices also will not be considered. However, if the Subject/ Course that has already been listed for Registration by the College in a Semester could not be offered due to unforeseen or unexpected reasons, then the Student will be allowed to have alternate choice either for a new Subject, if it is offered, or for another existing Subject (subject to availability of seats). Such alternate arrangements will be made by the Head of Department, with due notification and time-framed schedule, within the FIRST WEEK from the commencement of Class-work for that Semester.

5.0 Attendance Requirements

The programmes are offered on the basis of a unit system with each subject being considered a unit.

5.1 A student is eligible to appear for the semester end examinations, if the student acquires a minimum of 75% of attendance in aggregate of all the subjects / courses (excluding attendance in mandatory courses) for that semester.

5.2 Shortage of attendance in aggregate up to 10% (65% and above, and below 75%) in each semester may be condoned on medical grounds by the committee comprising of HOD, Class incharge and 2 senior faculty members.

5.3 Shortage of attendance below 65% in aggregate shall in no case be condoned.

5.4 Students whose shortage of attendance is not condoned in any semester are not eligible to take their end examinations of that semester.

5.5 A student fulfils the attendance requirement in the present semester, shall not be eligible for readmission into the same class.

5.6 A stipulated condonation fee is payable for condoning of shortage of attendance. This fee will be informed time to time by the college administration.

6.0 Academic Requirements

The following academic requirements have to be satisfied, in addition to the attendance requirements mentioned in item no. 5. The performance of the candidate in each semester shall be evaluated subject-wise, with a maximum of 100 marks per subject / course (theory / laboratory), on the basis of Continuous Internal Evaluation and Semester End Examination.

6.1 A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course, if he secures not less than 40% of marks (28 out of 70 marks) in the Semester End Examination, and a minimum of 50% of marks in the sum total of CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together; in terms of Letter Grades and this implies securing 'B' Grade or above in a subject.

6.2 A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to comprehensive viva-voce, if he secures not less than 50% of the total marks. The student is deemed to have failed, if he does not attend the comprehensive viva-voce as per the schedule given. In such a case, he may reappear for comprehensive viva-voce in supplementary examinations, as and when scheduled.

6.3 A student shall register for all subjects for total of 148 credits as specified and listed in the course structure, put in required attendance and fulfil the academic requirements for securing 148 credits obtaining a minimum of 'B' Grade or above in each subject, and all 148 credits securing Semester Grade Point Average **(SGPA) 6.0** (in each semester) and final Cumulative Grade Point Average **(CGPA)** (i.e., CGPA at the end of PGP) **6.0**, to complete the PGP successfully.

6.4 Marks and Letter Grades obtained in all those subjects covering the above specified 148 credits alone shall be considered for the calculation of final CGPA, which will be indicated in the Grade Card /Marks Memo of second year second semester.

6.5 If a student registers for extra subject(s) (in the parent department or other departments/ branches of Engineering) other than those listed subjects totalling to 148 credits as specified in the course structure, the performance in extra subject(s) (although evaluated and graded using the same procedure as that of the required 148 credits) will not be taken into account while calculating the SGPA and CGPA. For such extra subject(s) registered, a certificate will be issued with a letter grade indicated as a performance measure, subject to completion of the attendance and academic requirements as stated in items 5 and 6.1 - 6.3.

6.6 A student detained in a semester due to shortage of attendance may be readmitted in the same semester in the next academic year for fulfillment of academic requirements. The academic regulations under which a student has been readmitted shall be applicable. However, no grade allotments or SGPA/ CGPA calculations will be done for the entire semester in which the student has been detained.

6.7 A student eligible to appear for the Semester End Examination in any subject, but absent from it or failed (failing to secure 'B' Grade or above), may reappear for that subject at the supplementary examination as and when conducted. In such cases, his Internal Marks assessed earlier for that subject will be carried over, and added to the marks secured in the supplementary examination, for the purpose of evaluating his performance in that subject.

6. 8 A student shall be declared eligible for the award of the MCA Degree by **JNTUH**, if he/she earns 148 credits as per the specified course structure and pursues a course of study in not less than three and not more than six academic years. However, he is permitted to write the examinations for two more years after six academic years of course work, failing which he shall forfeit his seat in MCA programme.

7.0 Evaluation - Distribution and Weightage of Marks

7.1 The performance of a student in every subject/course (including practicals) will be evaluated for 100 marks each, with 30 marks allotted for Continuous Internal Evaluation (CIE) and 70 marks for Semester End Examination (SEE).

7.2 For theory courses, during the semester there is 2 mid-term examinations (internal exams of 20 marks each), 5 Unit Tests of 5 marks each and 2 assignments carrying 5 marks each.

7.3 Each mid-term examination will be of 1 hour 20 minutes consisting of Part-A short answer questions) for 10 marks and Part-B (long answer) for 10 marks. Part-A consists of 5 two marks questions. Part- B consists of 2 questions from each unit carrying 5 marks each and student should answer either of the 2 questions.

Each Unit Test will be of 1 hour duration, consisting of 3 questions from that unit carrying 5 marks each and student should answer any two questions for 10 Marks. These 10 marks are scaled down to 5 for Unit Test marks calculation.

7.4 First mid-term examination is conducted from first 2 units of syllabus and second mid-term examination is conducted for remaining 3 Units of syllabus during the last week of instruction.

7.5 The Continuous Internal Evaluation for theory course shall be made as average of marks obtained in CIE – I and CIE –II as detailed in the table below.

CIE – I	Marks	CIE - II	Marks
MID – I	20	MID - II	20
Best of Unit Test - I and Unit Test - II	5	Average of the best two of Unit Test – III, Unit Test – IV and Unit Test V	5
Assignment – I	5	Assignment - II	5
Total	30	Total	30

7.6 If a student is absent for any test/assignment, he is awarded zero marks for that test/assignment. However a candidate may be permitted on genuine grounds provided he/she has taken permission before the mid-term examination from the Head of the Department. Moreover, he/she has to apply for makeup examinations within a week after completion of mid-term examinations. A subcommittee will be constituted with the following composition to look into such cases.

Subcommittee-composition:

S.No	Faculty Member	Designation
1	Concern Head of the Department	Chairman
2	Senior faculty nominate by Principal	Member
3	One Senior faculty member of the concern department	Member
4	One faculty member of the other department	Member
5	Additional Controller of Examinations	Convener

7.7 The details of the Question Paper pattern for Semester End Examination (Theory) are given below:

The Semester End Examination will be conducted for 70 marks. It consists of two parts. i).Part A for 20 marks, ii). Part B for 50 marks.

- Part A is compulsory and consists of 5 questions, one from each unit and carrying 4 marks each.
- Part-B consists of five questions carrying 10 marks each. Each of these questions is from one unit and may contain sub-questions. For each question there will be an "either" "or" choice, which means that there will be two questions from each unit and the student should answer either of the two questions.

7.8 For practical subjects, 70 marks shall be awarded for performance in the Semester End Examinations and 30 marks shall be awarded as Internal Marks. Out of the 30 marks for internal evaluation, day-to-day work in the laboratory is evaluated for 20 marks and internal practical examination is evaluated for 10 marks conducted by the laboratory teacher concerned.

7.9 The semester end examination is conducted with an external examiner and the laboratory teacher. The external examiner is selected and appointed by the Principal from the list submitted by Head of the Department.

7.10 There shall be a Comprehensive Viva-Voce in III year II Semester. The Comprehensive Viva-Voce is intended to assess the student's understanding of various subjects he has studied during the MCA course of study. A Committee consisting of Head of the Department, one senior faculty member and an external examiner shall evaluate the Comprehensive Viva-Voce for a maximum of 100 marks. There are no internal marks for the Comprehensive Viva-Voce. The external examiner is selected and appointed by the Principal from the list submitted by Head of the Department. A candidate has to secure a minimum of 50% of marks to be declared successful. If he fails to obtain the minimum marks, he has to reappear for the viva-voce during the supplementary examinations.

7.11 EVALUATION OF PROJECT/DISSERTATION WORK

Every candidate shall be required to submit a thesis or dissertation on a topic approved by the Project Review Committee.

7.11.1 A Project Review Committee (PRC) shall be constituted with Head of the Department as Chairperson, Project coordinator and one senior faculty member.

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

7.11.3 After satisfying 7.11.2, a candidate has to submit, in consultation with his Project Supervisor, the title, objective and plan of action of his project work to the PRC for approval. Only after obtaining the approval of the PRC the student can initiate the Project work.

7.11.4 If a candidate wishes to change his supervisor or topic of the project, he can do so with the approval of the PRC. However, the PRC shall examine whether or not the change of topic/supervisor leads to a major change of his initial plans of project proposal. If yes, his date of registration for the project work starts from the date of change of Supervisor or topic as the case may be.

7.11.5 Candidate shall submit his project status report in two stages at least with a gap of 3 months between them.

7.11.6 The work on the project shall be initiated at the beginning of the III year II Semester and the duration of the project is one semester. A candidate is permitted to submit Project Thesis only after successful completion of all theory and practical courses with the approval of PRC not earlier than 20 weeks from the date of registration of the project work. For the approval of PRC the candidate shall submit the draft copy of thesis to the Head of the Department and make an oral presentation before the PRC.

7.11.7 After approval from the PRC, the soft copy of the thesis should be submiitted to the Department for ANTI-PLAGIARISM for the quality check and the plagiarism report should be included in the final thesis. If the copied information is less than 30%, then only thesis will be accepted for submission.

7.11.8 After satisfying 7.11.7, Three copies of the Project Thesis certified by the supervisor shall be submitted to the College.

7.11.9 For Project work Review in III Year II Sem. there is an internal marks of 50, the evaluation should be done by the PRC for 25 marks and Supervisor will evaluate for 25 marks. The Supervisor and PRC will examine the Problem Definition, Objectives, Scope of Work, Literature Survey. A candidate has to secure a minimum of 50% of marks to be declared successful for Project Work Review. If he fails to fulfill minimum marks, he has to reappear during the supplementary examination.

7.11.10 For Project Work Evaluation in III Year II Sem. there is an external marks of 100 and the same evaluated by the External examiner appointed by the principal from the panel 3 examiners submitted by the Head of the Department.The candidate has to secure minimum of 50% marks in Project Evaluation examination.

7.11.11 If candidate fails to fulfill as specified in 7.11.10, he/she will reappear for the Viva-Voce examination only after three months..

7.12 The Project Viva-Voce External examination marks must be submitted to the Exam Branch on the day of the examination.

8.0 Re-Admission/Re-Registration

8.1 Re-Admission for Discontinued Student

A student, who has discontinued the MCA degree programme due to any reason whatsoever, may be considered for '**readmission**' into the same degree programme with the academic regulations of the batch into which he gets readmitted, with prior permission from the authorities concerned, subject to item 6.6.

8.2 If a student is detained in a subject (s) due to shortage of attendance in any semester, he may be permitted to **re-register** for the same subject(s) in the same category (core or elective group) or equivalent subject, if the same subject is not available, as suggested by the Board of Studies of that department, as and when offered in the subsequent semester(s), with the

academic regulations of the batch into which he seeks re-registration, with prior permission from the authorities concerned, subject to item 3.2.

8.3 A candidate shall be given one chance to re-register for a maximum of two subjects, if the internal marks secured by a candidate are less than 50% and failed in those subjects. A candidate must re-register for failed subjects within four weeks of commencement of the class work and secure the required minimum attendance. In the event of the student taking this chance, his Continuous Internal Evaluation (internal) marks and Semester End Examination marks obtained in the previous attempt stand cancelled.

9.0 Examinations and Assessment - The Grading System

9.1 Grades will be awarded to indicate the performance of each student in each Theory Subject, or Lab/Practicals, or Project, etc., based on the % of marks obtained in CIE + SEE (Continuous Internal Evaluation + Semester End Examination, both taken together) as specified in Item 7 above, and a corresponding Letter Grade shall be given.

9.2 As a measure of the student's performance, a 10-point Absolute Grading System using the following Letter Grades (UGC Guidelines) and corresponding percentage of marks shall be followed:

% of Marks Secured in a subject/Course (Class Intervals)	Letter Grade (UGC Guidelines)	Grade Points
90% and above (≥90% , ≤ 100%)	O (Outstanding)	10
Below 90% but not less than 80% (≥80% , <90%)	A^+ (Excellent)	9
Below 80% but not less than 70% (≥ 70% , <80%)	A (Very Good)	8
Below 70% but not less than 60% (≥60% , <70%)	B^+ (Good)	7
Below 60% but not less than 50% (≥50% , <60%)	B (above Average)	6
Below 50% (< 50%)	F (FAIL)	0
Absent	**Ab**	**0**

9.3 A student obtaining F Grade in any Subject is deemed to have 'failed' and is required to reappear as 'Supplementary Candidate' for the Semester End Examination (SEE), as and when conducted. In such cases, his Internal Marks (CIE Marks) in those subjects will remain as obtained earlier.

9.4 If a student has not appeared for the examinations, 'Ab' Grade will be allocated to him for any subject and shall be considered 'failed' and will be required to reappear as 'Supplementary Candidate' for the Semester End Examination (SEE), as and when conducted.

9.5 A Letter Grade does not imply any specific marks percentage; it is only the range of percentage of marks.

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

9.7 A student earns Grade Point (GP) in each Subject/ Course, on the basis of the Letter Grade obtained by him in that Subject/ Course. The corresponding 'Credit Points' (CP) are computed by multiplying the Grade Point with Credits for that particular Subject/ Course.

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

9.8 The student passes the Subject/ Course only when he **gets GP ≥ 6 (B Grade or above).**

9.9 The Semester Grade Point Average (SGPA) is calculated by dividing the Sum of Credit Points (CP) secured from ALL Subjects/ Courses registered in a Semester, by the Total Number of Credits registered during that Semester. SGPA is rounded off to TWO Decimal Places. SGPA is thus computed as

SGPA = { $\sum_{i=1}^{N} C_i G_i$ } / { $\sum_{i=1}^{N} C_i$ } …. For each Semester,

where 'i' is the Subject indicator index (taking into account all Subjects in a Semester), 'N' is the no. of Subjects 'REGISTERED' for the Semester (as specifically required and listed under the Course Structure of the parent Department), 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.

9.10 The Cumulative Grade Point Average (CGPA) is a measure of the overall cumulative performance of a student over all Semesters considered for registration. The CGPA is the ratio of the Total Credit Points secured by a student in ALL registered Courses in ALL Semesters, and the Total Number of Credits registered in ALL the Semesters. CGPA is rounded off to TWO Decimal Places. CGPA is thus computed from the I Year Second Semester onwards, at the end of each Semester, as per the formula

$$\text{CGPA} = \left\{ \frac{\sum_{J=1}^{M} C_J G_J}{\sum_{J=1}^{N} C_J} \right\}$$ **..... FOR ALL "S" numbers of SEMESTERS REGISTERED**

(I.E., UP TO AND INCLUSIVE OF S SEMESTER, S ≥ 2),

where 'M' is the TOTAL no. of Subjects (as specifically required and listed under the Course Structure of the parent Department) the Student has 'REGISTERED' for from the 1^{st} Semester onwards upto and inclusive of the Semester S (obviously M > N), 'j' is the Subject indicator index (taking into account all Subjects from 1 to S Semesters), C_j is the no. of Credits allotted to the j^{th} Subject, and G_j represents the Grade Points (GP) corresponding to the Letter Grade awarded for that j^{th} Subject. After registration and completion of I Year I Semester however, the SGPA of that Semester itself may be taken as the CGPA, as there are no cumulative effects.

Illustration of calculation of SGPA

Course/Subject	Credits	Letter Grade	Grade points	Credit Points
Course 1	4	A	8	4*8 = 32
Course 2	4	O	10	4*10 = 40
Course 3	4	B	6	4*6 = 24
Course 4	3	B	6	3*6 = 18
Course 5	3	A+	9	3*9 = 27
Course 6	3	B	6	3*6 = 18
	21			159

SGPA = 159/21 = 7.57

Illustration of calculation of CGPA

Semester	Credits	SGPA	Credits * SGPA
Semester I	24	7	24*7 = 168
Semester II	24	6	24*6 = 144
Semester III	24	6.5	24*6.5 = 156
Semester IV	24	6	24*6 = 144
	96		612

CGPA = 612/96 = 6.37

10.0 Award of Degree and Class

10.1 If a student 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 **148** Credits (with CGPA 6.0), shall be declared to have 'QUALIFIED' for the award of the MCA Degree that he was admitted into.

10.2 Award of Class

After a student has earned the requirements prescribed for the completion of the programme and is eligible for the award of MCA Degree, he shall be placed in one of the following three classes based on the CGPA:

Class Awarded	CGPA
First Class with Distinction	≥ 7.75
First Class	$6.75 \leq$ CGPA < 7.75
Second Class	$6.00 \leq$ CGPA < 6.75

A student with final CGPA (at the end of the **PGP) < 6.00** shall not be eligible for the Award of Degree.

11.0 Withholding of Results

If the student has not paid the dues, if any, to the College or if any case of indiscipline is pending against him, the result and degree of the student will be withheld and he will not be allowed into the next semester.

12.0. Transitory Regulations

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

12.2 The candidate who fails in any subject will be given two chances to pass the same subject; otherwise, he has to identify an equivalent subject as per R18 Academic Regulations.

13.0 General

13.1 **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.

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

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

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

13.5 In case of any doubt or ambiguity in the interpretation of the above rules, the decision of the College is final.

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

MALPRACTICE RULES
DISCIPLINARY ACTION FOR CONDUCT IN EXAMINATION

	Nature of Malpractices/Improper conduct	**Punishment**
	if the candidate:	
1.(a)	Possesses or keeps accessible in examination hall, any paper, note book, programmable calculators, Cell phones, pager, palm computers or any other form of material concerned with or related to the subject of the examination (theory or practical) in which he is appearing but has not made use of (material shall include any marks on the body of the candidate which can be used as an aid in the subject of the examination)	Expulsion from the examination hall and cancellation of the performance in that subject only
1.(b)	Gives assistance or guidance or receives it from any other candidate orally or by any other body language methods or communicates through cell phones with any candidate or persons in or outside the exam hall in respect of any matter	Expulsion from the examination hall and cancellation of the performance in that subject only of all the candidates involved. In case of an outsider, he will be handed over to the police and a case is registered against him
2	Has copied in the examination hall from any 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.
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 practical's and project work) already appeared and shall not be allowed to appear for examinations of the remaining subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all End 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 End examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.

5	Uses objectionable, abusive or offensive language in the answer paper or in letters to the examiners or writes to the examiner requesting him to award pass marks	Cancellation of the performance in that subject
6	Refuses to obey the orders of the Chief Superintendent/Assistant – Superintendent / any officer on duty or misbehaves or creates disturbance of any kind in and around the examination hall or organizes a walk out or instigates others to walk out, or threatens the officer-in charge or any person on duty in or outside the examination hall of any injury to his person or to any of his relations whether by words, either spoken or written or by signs or by visible representation, assaults the officer-in-charge, or any person on duty in or outside the examination hall or any of his relations, or indulges in any other act of misconduct or mischief which result in damage to or destruction of property in the examination hall or any part of the College campus or engages in any other act which in the opinion of the officer on duty amounts to use of unfair means or misconduct or has the tendency to disrupt the orderly conduct of the examination.	In case of students of the college, they shall be expelled from examination halls and cancellation of their performance in that subject and all other subjects the candidate(s) has (have) already appeared and shall not be permitted to appear for the remaining examinations of the subjects of that semester/year. The candidates also are debarred and forfeit their seats. In case of outsiders, they will be handed over to the police and a police case is registered against them.
7	Leaves the exam hall taking away answer script or intentionally tears of the script or any part thereof inside or outside the examination hall.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all End examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
8	Possess any lethal weapon or firearm in the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat.
9	If student of the college, who is not a candidate for the particular examination or any person not connected with the college indulges in any malpractice or improper conduct mentioned in clause 6 to 8.	Student of the 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 Examination Result Processing Committee (ERPC) for further action to award suitable punishment.	

J.B INSTITUTE OF ENGINEERING AND TECHNOLOGY

UGC AUTONOMOUS

MASTER OF COMPUTER APPLICATIONS (MCA) R18
Effective from Academic Year 2018- 19 admitted batch
COURSE STRUCTURE AND SYLLABUS

I Year – I Semester

CODES	Category	Course Title	L	P	C
GF11A	Core Course - I	Mathematical Foundations of Computer Science	4	--	4
GF11B	Core Course - II	Computer Organization	4	--	4
GF11C	Core Course - III	Computer Programming and Problem Solving	4	--	4
GF11D	Core Course - IV	Probability and Statistics	4	--	4
GF11E	Core Course - V	Accounting and Financial Management	4	--	4
GF11F	Laboratory - I	IT Workshop	--	2	2
GF11G	Laboratory - II	Computer Programming Using C and C++ Lab	--	2	2
GF11H	Laboratory - III	English Language Communication Skills Lab	--	2	2
	Total Credits		20	6	26

I Year – II Semester

CODES	Category	Course Title	L	P	C
GF12A	Core Course - I	Data Structures and Algorithms	4	--	4
GF12B	Core Course - II	Operating Systems	4	--	4
GF12C	Core Course - III	Scripting Languages	4	--	4
GF12D	Core Course - IV	Operations Research	4	--	4
GF12E	Core Course - V	Software Engineering	4	--	4
GF12F	Laboratory - I	Data Structures Lab Using C++	--	2	2
GF12G	Laboratory - II	Scripting(Python) Lab	--	2	2
GF12H	Laboratory - III	Operation System Lab	--	2	2
	Total Credits		20	6	26

II Year - I Semester

CODES	Category	Course Title	L	P	C
GF21A	Core Course - I	Data Base Management Systems	4	--	4
GF21B	Core Course - II	Computer Networks	4	--	4
GF21C	Core Course - III	Statistical Programming with 'R'	4	--	4
GF21D	Core Course - IV	Object Oriented Analysis and Design using UML	4	--	4
GF21E	Core Course - V	Java Programming	4	--	4
GF21F	Laboratory - I	Data Base Management Systems Lab	--	2	2
GF21G	Laboratory - II	Java Programming Lab	--	2	2
	Total Credits		20	4	24

II Year - II Semester

CODES	Category	Course Title	L	P	C
GF22A	Core Course - I	Web Technologies	4	--	4
GF22B	Core Course - II	Linux Programming	4	--	4
GF22C GF22D GF22E	Core Elective – I	1. Machine Learning 2. Data Warehousing and Data Mining 3. Information Security	4	--	4
GF22F GF22G GF22H	Core Elective – II	1. Distributed Systems 2. Software Process and Project Management 3. Soft Computing	4	--	4
	Open Elective – I	Open Elective	4	--	4
GF55I	Laboratory - I	Web Technologies Lab	--	2	2
GF22J	Laboratory - II	Linux Programming Lab	--	2	2
	Total Credits		20	4	24

III Year - I Semester

CODES	Category	Course Title	L	P	C
GF31A	Core Course - I	Android Application Development	4	--	4
GF31B	Core Course - II	Software Testing Methodologies	4	--	4
GF31C GF31D GF31E	Core Elective – III	1. Web Services & Service Oriented Architecture (SOA) 2. Distributed Databases 3. Information Retrieval Systems	4	--	4
GF31E GF31F GF31G	Core Elective – IV	1. Ethical Hacking 2. Multi Media and Rich InternetApplications 3. Semantic web and Social Networks	4	--	4
	Open Elective – II	Open Elective	4	--	4
GF31H	Laboratory - I	Android Application Development Lab	--	2	2
GF31I	Laboratory - II	Software Engineering Methodologies and UML Lab	--	2	2
	Total Credits		20	4	24

III Year - II Semester

CODES	Course Title	L	P	C
GF32A	Comprehensive Viva Voce	--		4
GF32B	Project Work Review	--	8	4
GF32C	Project Viva Voce	--	20	16
		--	28	24

Note: All end Examination (Theory and Practical) are of three hours' duration.

L-Theory, P-Practical, C –Credits

Total No of Credits = 148

- Basic Computer Programming skills is required for all open electives. Additionally, knowledge on the specified area mentioned in prerequisites is required for opting the open elective
- Note: A student can register for any open elective subject provided that he has not already registered for the same subject.

S. No.	Open Electives	Prerequisites
1.	Algorithmic	--
2.	Big Data Analytics	Data Bases , Maths
3.	Bioinformatics	Data Structures
4.	Biometrics	--
5.	Computer Forensics	Maths, Data Structures
6.	Cyber Security	Internet Technologies
7.	Distributed Systems Security	Information Security
8.	E-Commerce	Internet Technologies
9.	Embedded Systems	Digital logic
10.	Intellectual Property Rights	--
11.	Internet of Things	Java
12.	Internet Technologies and Services	Java
13.	Mobile Computing	Java
14.	Mobile Application Security	Mobile Application Development
15.	Open Stack Cloud Computing	Linux Programming
16.	Principles Of Information Security	--
17.	Social Media Intelligence	--
18.	Storage Area Networks	Computer Networks
19.	Web Usability	--
20.	Organizational Structure and Personal Management	

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
I YEAR I SEMESTER	**4**	**0**	**4**

MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

Objectives:

- To develop problem-solving techniques and explore topics in a variety of areas of discrete mathematics, including but not limited to logic, graph theory, set theory, recursions, combinatorics, and algorithms.
- Students will learn to express statements in the language of formal logic and draw conclusions, model situations in terms of graph and set theory, find and interpret recursive definitions for mathematical sequences, use combinatorial methods to approach counting problems.
- Ability to Illustrate by examples the basic terminology of functions, relations, and sets and demonstrate knowledge of their associated operations.
- Ability to Demonstrate in practical applications the use of basic counting principles of permutations, combinations, inclusion/exclusion principle and the pigeonhole methodology.
- Ability to represent and Apply Graph theory in solving computer science problems.

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: Properties of binary Relations, equivalence, transitive closure, compatibility and partial ordering relations, Lattices, Hasse diagram. Functions: Composition of functions, Inverse Function, Hashing functions, Natural numbers, recursive functions.
Algebraic structures: Algebraic systems, Examples and general properties, Semi groups and monoids, groups and sub groups, homomorphism. Lattice as partially ordered sets, Boolean algebra.

UNIT III

Elementary Combinatorics: Basics of counting, Combinations & Permutations, with repetitions, Constrained repetitions,
Binomial Coefficients, Binomial and Multinomial theorems, the principles of Inclusion – Exclusion, Pigeon hole principles and its application.

UNIT IV

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 V

Graphs: Basic Concepts, Isomorphism and Sub graphs, Trees and their properties, Spanning Trees, Directed trees, Binary trees,
Planar Graphs, Multi graphs and Euler circuits, Hamiltonian graphs, Chromatic Numbers.

TEXT BOOKS:

1. Mathematical Foundations of Computer Science, Dr. D.S.Chandrasekharaiah, Third Edition, PRISM.

2. Discrete Mathematics with Applications to Computer Science, J P Trembley and R Manohar, TMH, 2008. (UNITs I and II)

REFERENCE BOOKS:

1 Discrete Mathematics for Computer Scientists and Mathematicians, second edition, J.L.Mott, A. Kandel, T.P. Baker, PHI
2 Elements of Discrete Mathematics- A Computer Oriented Approach, C.L.Liu, D.P. Mohapatra, 3rd Edition, TMH.
3. Discrete and Combinatorial Mathematics- An Applied Introduction-5th Edition–Ralph. P.Grimaldi, Pearson Education.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
I Year - I Semester	**4**	**0**	**4**

COMPUTER ORGANIZATION

Objectives:

- 1.To understand how computers are constructed out of a set of functional units.
- 2.To understand how these functional units operate, interact and communicate.
- 3.To understand the factors and trade-offs that affect computer performance.
- 4.To understand concrete representation of data at the machine level.
- 5.To understand how computations are actually performed at the machine level.
- 6.To understand how problems expressed by humans are expressed as binary strings in a machine.
- 7.Understand the system interconnection and the different I/O techniques.
- 8.Explain the functioning and programming of the INTEL-8086.
- 9.Understand the design of processors, the structure and operation of memory and virtual memory, cache, storage, and pipelining, system integration, and peripherals.
- Identify the different architectural and organizational design issues that can affect the performance of a computer such as Instruction Sets design, Pipelining, RISC architecture, and Superscalar architecture.
- Design an interconnection networks and multiprocessors.

Outcomes:

- After this course students understand in a better way the I/O and memory organization in depth.
- They should be in a position to write assembly language programs for various applications.

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/O 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 of Multiprocessors.
Interconnection Structures, Interprocessor Arbitration, InterProcessor Communication and Synchronization, Cache Coherence, 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, 5th Edition, McGraw Hill.

2. Fundamentals of Computer Organization and Design, Sivarama P. Dandamudi,

Springer Int. Edition.

3. Computer Organization and Architecture, William Stallings, 8th Edition, Pearson, 2007.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
I Year - I Semester	**4**	**0**	**4**

COMPUTER PROGRAMMING AND PROBLEM SOLVING

Objectives:

- To understand the various steps in Program development.
- To understand the basic concepts in C Programming Language.
- To learn how to write modular and readable C Programs
- To learn to write programs (using structured programming approach) in C to solve problems.
- To introduce the students to basic data structures such as lists, stacks and queues.
- To make the student understand simple sorting and searching methods.

Outcomes:

- Demonstrate the basic knowledge of computer hardware and software.
- Ability to apply solving and logical skills to programming in C language and also in other languages.

The First Three Units (I, II & III) are for 'C' Programming and the last Two Units (IV and V) are for C++.

UNIT –I

Introduction to Computers: Computer Systems, Computing Environments, Computer Languages, Creating and Running programs, System Development.
Introduction to Language: C programs, identifiers, Types, Variables, Constants, Operators, Input/Output, Expressions:- precedence and associativity, evaluating expressions, type conversion, statements.Selection:-making decisions (Conditional statements), repetition (control statements).

UNIT-II

Functions: Designing Structured Programs, Functions in C, User-defined functions, Inter-function communication (Parameter passing mechanism), Standard Functions, Scope, Recursion.**Arrays:** Concepts, Using Arrays in C, Inter-function communication (Passing arrays to functions), Two Dimensional arrays, Multidimensional arrays, Application**:-**selection, bubble and insertion sort.**Pointers:** Introduction, Pointers for Inter-function communication, Pointers to P{pointers, Compatibility, Arrays and Pointers(w.r.t arithmetic, passing arrays to function),memory allocation functions.**Strings:** String concepts, C String Input/Output Functions, Array of Strings, String Manipulation functions.

UNIT –III

Enumerated, Structure and Union Types: The Type Definition (typedef), Enumerated types, Structure, Unions.Enumarations, Unions, and Pre-Prpocessor Directives.

Text Input/Output Files in C: Files, Streams, Standard Library Input/Output functions, Formatting Input/Output functions, and Character Input/Output functions.

Binary Input/Output: Text versus Binary streams, Standard Library functions in files.

Converting File Type.

UNIT-IV

Introduction to C++: Introduction and Basic elements of C++, **Functions in C++:** User-defined functions, Types of functions, Parameter passing mechanisms, Overloading, Inline functions, Function Templates.

UNIT –V

Classes: Introductions to classes, Inheritance, Operator Overloading, Dynamic Polymorphism using Virtual functions, Abstract Classes. Templates and Exceptional Handling.

TEXT BOOKS:

1) Behrouz A. Forouzan, Richard F Grilberg,"Computer Science- A Structured Approach using C",3rd Edition ,Cengage Learning 2007.(For Units I,II,III)

2)D.S.Malik,"C++ Programming Language", Cengage Learning 2009.(For Units IV, V)

References:

1) Harry.H. Cheng,"C/C++ for Engineers and Scientist-An Interpretive Approach", McGraw-Hill International Edition, 2009.
2) 2. Owen L.Astrachan,"Programming with C++", Tata McGraw Hill, 2007.

3) Bjarne Stroustrup,"The C++ Programming Language", 3rd Edition, Addison-Wesley,1918.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	L	P	C
I YEAR I SEMESTER	4	0	4

PROBABILITY AND STATISTICS

UNIT I

Probability: Sample space and events – Probability – The axioms of probability - Some elementary theorems - Conditional probability – Baye‟s theorem.

UNIT II

Random variables – Discrete and continuous – Distribution – Distribution function.Distribution - Binomial, Poisson and normal distribution – related properties.

UNIT III

Sampling distribution: Populations and samples - Sampling distributions of mean (known and unknown) proportions, sums and differences.Estimation: Point estimation – interval estimation - Bayesian estimation.

UNIT IV

Test of Hypothesis-Means and proportions- Hypothesis concerning one and two means- Type 1 and Type II errors,. One tail, two tail tests. Tests of significance- Student‟s t-test, test F-test Estimation of proportions

UNIT V

Curve fitting: The method of least squares – Inferences based on the least squares estimations

- Curvilinear regression – multiple regressions – correlation for univariate and bivariate distributions.

TEXT BOOKS:

1. Probability and statistics for engineers (Erwin Miller and John E.Freund), R A Johnson and C.B.Gupta.7th edition, PHI.
2. Introduction to Probability and Statistics, 12th edition, W.Mendenhall, R.J.Beaver and B.M.Beaver, Cengage Learning.

REFERENCE BOOKS:

1. Text book of Probability and Statistics Dr.Shahnaz Bathul, V.G.S.Publishers 2003.
2. Probability and Statistics in Engineering, 4th Edition, William W.Hines, Douglas C.Montgomery, David M.Goldsman Connie M.Borror, Wiley Student Edition.
3. Probability, Statistics and Queuing Theory, 2nd Edition, Trivedi, John Wiley and Sons

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
I Year - I Semester	**4**	**0**	**4**

ACCOUNTING AND FINANCIAL MANAGEMENT

This course is designed to introduce students to the principles, concepts, and applications of financial accounting and management.

An introduction to accounting emphasizing how general purpose financial statements communicate information about the business corporation‟s performance and position for users internal and external to management. Approximately one third of the course emphasizes how the accountant processes and presents the information and includes exposure to recording transactions, adjusting balances and preparing financial statements for service and merchandise firms according to established rules and procedures.

This course presents the underlying framework and concepts of Financial Accounting in the context of how accounting fits into the overall business environment of contemporary society. Financial accounting is the basic means of recording and reporting financial information in a business. Students will learn how accounting functions as an information development and communication system that supports economic decision making and provides value to entities and society. Students will discover the uses and limitations of financial statements and related information and apply analytical tools in making both business and financial decisions. Topics examined include those related to corporate financial position, operating results, cash flows, and financial strength. Students will study the basic accounting system and will be shown how the various accounting alternatives for recording financial transactions impact on the usefulness of the information provided for decision-making. During coverage of relevant topics reference will be made to recent lapses in ethical reporting and the resulting impact on the financial markets and society.

In Financial Management students are introduced to concepts and tools that enable them to think critically about the financial opportunities and challenges faced by an organization. In working through the material, they build upon the understanding they already have of accounting, business, and mathematics. They learn how to use financial statements such as balance sheets, income statements, and statements of cash flow. They prepare budgets, analyze investment options, and determine the best means of financing business endeavors. They discover ways of assessing both the return and the risk involved in a firm's financial decisions. The focus of this course is on solving practical business problems similar to those encountered in the workplace.

Objectives

1 Explain financial accounting terminology and the recording process.

2 Discuss and apply the basic principles of accounting, the accounting model, and the accounting cycle.

3 Demonstrate and show how items such as assets, liabilities, share capital, etc., are reported in the financial statements.

4 Carry out journal entries and adjustments.
5 Prepare and analyze a trial balance and financial statements.

6 Analyze a cash flow statement.

7 Analyze financial statements using ratio analysis.

8 Understand the elements, uses, and limitations of each financial statement and the relationships among the statements

9 Understand how financial information, primarily that provided by the financial statements, can be used to analyze business operations and make economic decisions

10 Identify the basic principles used in safeguarding assets and insuring the accuracy of accounting records

11 Appreciate the role accounting software applications play in gathering, recording, reporting and interpreting financial accounting information

12 Interpret and create standard financial statements.

13 Perform financial statement analysis for the purposes of evaluating and forecasting in financial management.

14 Evaluate a firm's working capital position.
15 Estimate the components of cost of capital by applying time value of money principles.
16 Perform net present value analysis for capital budgeting purposes.
17 Evaluate risk in the capital budgeting process.
18 Evaluate the cost of debt, preferred stock, and common stock as sources of capital.

19 The Most important learning objective is how to do accounts in Computers.

Prerequisites

- Some basic accounting knowledge and mathematics level is recommended.
- Ability to understand the knowledge of cost estimation towards the business sceneries.
- Ability to analyze market competency in the global village
- To become a Master in 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.

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.

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.

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: Budgeting–cash budget, sales budget – flexible Budgets and master budgets.

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).

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.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
I Year - I Semester	**4**	**0**	**4**

IT WORKSHOP

PC Hardware introduces the students to a personal computer and its basic peripherals, the process of assembling a personal computer, installation of system software like MS Windows, Linux and the required device drivers. In addition hardware and software level troubleshooting process, tips and tricks would be covered. **The students should work on working PC to disassemble and assemble to working condition and install Windows and Linux on the same PC. Students are suggested to work similar tasks in the Laptop scenario wherever possible.**

Internet & World Wide Web module introduces the different ways of hooking the PC on to the Internet from home and workplace and effective usage of the internet. Usage of web browsers, email, newsgroups and discussion forums would be covered. In addition, awareness of cyber hygiene, i.e., protecting the personal computer from getting infected with the viruses, worms and other cyber attacks would be introduced.

Productivity tools module would enable the students in crafting professional word documents, excel spread sheets and power point presentations using the Microsoft suite of office tools and LaTeX. (**Recommended to use Microsoft office 2007 in place of MS Office 2003)**

PC Hardware

Task 1 : Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor.

Task 2 : Every student should disassemble and assemble the PC back to working condition. Lab instructors should verify the work and follow it up with a Viva. Also students need to go through the video which shows the process of assembling a PC. A video would be given as part of the course content.

Task 3 : Every student should individually install MS windows on the personal computer.

Lab instructor should verify the installation and follow it up with a Viva.

Task 4 : Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot with both windows and Linux. Lab instructors should verify the installation and follow it up with a Viva.

Task 5: Hardware Troubleshooting: Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva.

Task 6: Software Troubleshooting: Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva.

Internet & World Wide Web

Task 1: Orientation & Connectivity Boot Camp: Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IP setting. Finally students should demonstrate, to the instructor, how to access the websites and email. If there is no Internet connectivity preparations need to be made by the instructors to simulate the WWW on the LAN.

Task 2: Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.

Task 3: Search Engines & Netiquette: Students should know what search engines are and how to use the search engines. A few topics would be given to the students for which they need to search on Google. This should be demonstrated to the instructors by the student.

Task 4: Cyber Hygiene: Students would be exposed to the various threats on the Internet and would be asked to configure their computer to be safe on the Internet. They need to first install antivirus software, configure their personal firewall and windows update on their computer. Then they need to customize their browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

Task 5: Develop your home page using HTML Consisting of your photo, name, address and education details as a table and your skill set as a list.

Productivity tools:

LaTeX and Word:

Word Orientation: The mentor needs to give an overview of LaTeX and Microsoft (MS) office 2007/ equivalent (FOSS) tool word: Importance of LaTeX and MS office 2007/ equivalent (FOSS) tool Word as word Processors, Details of the three tasks and features that would be covered in each, using LaTeX and word – Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter.

Task 1: Using LaTeX and Word to create project certificate. Features to be covered:- Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and Word.

Task 2 : Creating project abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Task 3 : Creating a Newsletter : Features to be covered:- Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs and Mail Merge in word.

Excel:

Excel Orientation: The mentor needs to tell the importance of MS office 2007/ equivalent (FOSS) tool Excel as a Spreadsheet tool, give the details of the two tasks and features that would be covered in each. Using Excel – Accessing, overview of toolbars, saving excel files, Using help and resources.

Task 1: Creating a Scheduler - Features to be covered: - Gridlines, Format Cells, Summation, auto fill, Formatting Text.

Task 2 : Calculating GPA - .Features to be covered:- Cell Referencing, Formulae in excel – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, LOOKUP/VLOOKUP, Sorting, Conditional formatting.

LaTeX and MS/equivalent (FOSS) tool Power Point:

Task1: Students will be working on basic power point utilities and tools which help them create basic power point presentation. Topic covered during this week includes :- PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows in both LaTeX and Power point. Students will be given model power point presentation which needs to be replicated (exactly how it"s asked).

Task 2: Second week helps students in making their presentations interactive. Topic covered during this week includes: Hyperlinks, Inserting –Images, Clip Art, Audio, Video, Objects, Tables and Charts.

Task 3: Concentrating on the in and out of Microsoft power point and presentations in LaTeX. Helps them learn best practices in designing and preparing power point presentation. Topic covered during this week includes: - Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide slotter, notes etc), Inserting – Background, textures, Design Templates, Hidden slides.

REFERENCE BOOKS:

1.Introduction to Information Technology, ITL Education Solutions limited, Pearson Education.

2.LaTeX Companion – Leslie Lamport, PHI/Pearson.

3. Introduction to Computers, Peter Norton, 6/e Mc Graw Hill.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	L	P	C
I Year - I Semester	4	0	4

COMPUTER PROGRAMMING USING C & C++ LAB

Objectives:

- To write programs in C to solve the problems.
- To implement linear data structures such as lists, stacks, queues.
- To implement simple searching and sorting methods.

Outcomes:

- Ability to write programs for different kinds of problems in C.
- Work confidently in compilers like C and others.
- Ability to identify appropriate data type or data structure to given problem.

Recommended Systems/Software Requirements:

- Intel based desktop PC.
- ANSI C Compiler with Supporting Editors.

C-Programs:

1. Program to calculate simple formulae like area of Circle, Rectangle etc.
2. Program to find Maximum, Minimum and Sum of given set of numbers.
3. Program to find whether a number is Prime or not.
4. Program to find Sine(x) and Cosine(x) using series expansion.
5. Program to demonstrate Call-by-Reference parameter passing mechanism.
6. Program to Implement of Linear and Binary Search mechanisms.
7. Program to Implement Selection and Bubble sort.
8. Program to Implement Multiplication using pointers..
9. Program to find the number of letters, words and sentences in a given string.
10. Program to do String manipulation without using in-built library functions.
11. Programs to generate address labels using structures.
12. Program to demonstrate Sequential File Access.
13. Program to demonstrate Random File Access.

C++-Programs:

14. Recursive functions (Great common divisor, Tower of Hanoi, Fibonacci etc).
15. Classes for Bank Account, Student Information, Library Catalogue, Employee.
16. Creation of Complex, Vector classes using operator overloading.
17. Creation of Inheritance hierarchy (bank account, person)
18. Tempalte functions for Min () and Max () for finding minimum and maximum in a List.
19. Program on Class Templates.
20. Programs demonstrating virtual, pure virtual functions using abstract base class"Shape".
21. Program to demonstrate multiple inheritances and exception handling.
22. Program demonstrating Stream and File I/O using student and employee classes.

References:

1) Harry.H. Cheng,"C/C++ for Engineers and Scientist-An Interpretive Approach", McGraw-Hill International Edition, 2009.
2) Owen L.Astrachan,"Programming with C++", Tata McGraw Hill, 2007.
3) Bjarne Stroustrup,"The C++ Programming Language", 3rd Edition, Addison-Wesley,1918.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

	L	P	C
MCA			
I Year - I Semester	**4**	**0**	**4**

ENGLISH LANGUAGE COMMUNICATION SKILLS LAB

The **Language Lab** focuses on the production and practice of sounds of language and familiarizes the students with the use of English in everyday situations and contexts.

Objectives:

1. To facilitate computer-aided multi-media instruction enabling individualized and independent language learning

2. To sensitize the students to the nuances of English speech sounds, word accent, intonation and

3. rhythm

4. To bring about a consistent accent and intelligibility in their pronunciation of English by providing an opportunity for practice in speaking

5. To improve the fluency in spoken English and neutralize mother tongue influence

6. To train students to use language appropriately for interviews, group discussion and public speaking.

Syllabus: English Language Communication Skills Lab shall have two parts:

Computer Assisted Language Learning (CALL) Lab
Interactive Communication Skills (ICS) Lab

The following course content is prescribed for the **English Language Communication Skills Lab**

Exercise – I

CALL Lab: Introduction to Phonetics – Speech Sounds – Vowels and Consonants

ICS Lab: Ice-Breaking activity and JAM sessionArticles, Prepositions, Word formation-Prefixes & Suffixes, Synonyms & Antonyms.

Exercise – II

CALL Lab: Structure of Syllables - Past Tense Marker and Plural Marker – Weak Forms and Strong Forms - Consonant Clusters.

ICS Lab: Situational Dialogues – Role-Play- Expressions in Various Situations – Self-introduction and Introducing Others – Greetings – Apologies – Requests – Social and Professional Etiquette - Telephone Etiquette.

Concord (Subject in agreement with verb) and Words often misspelt- confused/misused

Exercise - III

CALL Lab: Minimal Pairs- Word accent and Stress Shifts- Listening Comprehension.
ICS Lab: Descriptions- Narrations- Giving Directions and guidelines.
Sequence of Tenses, Question Tags and One word substitutes.

Exercise – IV

CALL Lab: Intonation and Common errors in Pronunciation.
ICS Lab: Extempore- Public Speaking
Active and Passive Voice, –Common Errors in English, Idioms and Phrases

Exercise – V

CALL Lab: Neutralization of Mother Tongue Influence and Conversation Practice
ICS Lab: Information Transfer- Oral Presentation Skills
Reading Comprehension and Job Application with Resume preparation.

Minimum Requirement of infra structural facilities for ELCS Lab:

1. Computer Assisted Language Learning (CALL) Lab:

The Computer aided Language Lab for 40 students with 40 systems, one master console, LAN facility and English language software for self- study by learners.

System Requirement (Hardware component):

Computer network with LAN with minimum 60 multimedia systems with the following specifications: i) P – IV Processor

Speed – 2.8 GHZ
RAM – 512 MB Minimum
Hard Disk – 80 GB
1 Headphones of High quality

2. Interactive Communication Skills (ICS) Lab :

The Interactive Communication Skills Lab: A Spacious room with movable chairs and audio-visual aids with a Public Address System, a T. V., a digital stereo –audio & video system and camcorder etc.

Prescribed Lab Manual: A Manual entitled "***English Language Communication Skills (ELCS) Lab Manual- cum- Work Book***", published by Cengage Learning India Pvt. Ltd, New Delhi. 2013.

Suggested Software:

2 **Cambridge Advanced Learners' English Dictionary with CD.**
3 **Grammar Made Easy by Darling Kindersley**
4 **Punctuation Made Easy by Darling Kindersley**
5 Clarity Pronunciation Power – Part I
6 Clarity Pronunciation Power – part II

7 **Oxford Advanced Learner's Compass, 7th Edition**
8 ***DELTA's key to the Next Generation TOEFL Test: Advanced Skill Practice.***
9 Lingua TOEFL CBT Insider, by Dreamtech
10 TOEFL & GRE (KAPLAN, AARCO & BARRONS, USA, Cracking GRE by CLIFFS)

11 **English in Mind (Series 1-4), Herbert Puchta and Jeff Stranks with Meredith Levy, Cambridge**

12 English Pronunciation in Use, Cambridge University Press
13 Technical Communication, OUP
14 Communication Skills, OUP

SUGGESTED READING:

1. Rama Krishna Rao, A. *et al. English Language Communication Skills – A Reader cum Lab Manual Course Content and Practice.* Chennai: Anuradha Publishers

2. Suresh Kumar, E. & Sreehari, P. 2009. *A Handbook for English Language Laboratories.* New Delhi: Foundation

3. *Speaking English Effectively* 2nd Edition by Krishna Mohan and N. P. Singh, 2011. Macmillan Publishers India Ltd. Delhi.

4. Sasi Kumar, V & Dhamija, P.V. *How to Prepare for Group Discussion and Interviews.* Tata McGraw Hill

5. 5.Hancock, M. 2009. *English Pronunciation in Use. Intermediate.* Cambridge: CUP
6. Spoken English: A Manual of Speech and Phonetics by R. K. Bansal & J. B. Harrison.

7. 7.Hewings, M. 2009. *English Pronunciation in Use. Advanced.* Cambridge: CUP
8. Marks, J. 2009. *English Pronunciation in Use. Elementary.* Cambridge: CUP
9. Nambiar, K.C. 2011. *Speaking Accurately. A Course in International Communication.* New Delhi : Foundation

10. Soundararaj, Francis. 2012. *Basics of Communication in English. New Delhi: Macmillan*
11. **Spoken English** (CIEFL) in 3 volumes with 6 cassettes, OUP.

12. **English Pronouncing Dictionary** Daniel Jones Current Edition with CD.

13. **A text book of English Phonetics for Indian Students** by T.Balasubramanian (Macmillan)

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
I Year - II Semester	**4**	**0**	**4**

DATA STRUCTURES AND ALGORITHMS

Objectives:

- To understand the basic concepts such as Abstract Data Types, Linear and Non Linear Data structures.
- To understand the notations used to analyze the Performance of algorithms.
- To understand the behavior of data structures such as stacks, queues, trees, hash tables, search trees, Graphs and their representations.
- To choose the appropriate data structure for a specified application. To understand and analyze various searching and sorting algorithms.
- To write programs in C to solve problems using data structures such as arrays, linked lists, stacks, queues, trees, graphs, hash tables, search trees.

Outcomes:

- Learn how to use data structure concepts for realistic problems.
- Ability to identify appropriate data structure for solving computing problems in respective language.
- Ability to solve problems independently and think critically.Be able to analyze algorithms and improve the efficiency of algorithms.
- Apply different designing methods for development of algorithms to realistic problems, such as divide and conquer, greedy and etc.
- Ability to understand and estimate the performance of algorithm

UNIT I

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. 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, Stack ADT, definitions, operations, array and linked implementations, applications-infix to postfix conversion, recursion implementation, Queue ADT, definitions and operations ,array and linked Implementations.

UNIT II

Non Linear data structures- Trees – Basic Terminology, Binary tree ADT, array and linked representations, traversals, threaded binary trees, Disjoint Sets, Union and Find algorithms, Priority Queues-Definition, ADT, Realizing a Priority Queue using Heap. Graphs – Introduction, Basic Terminology, Graph Representations- Adjacency matrix, Adjacency lists, Adjacency multi lists, Graph traversals- DFS and BFS.

UNIT III

Searching- Linear Search, Binary Search, Hashing-Introduction, hash tables, hash functions, collision resolution methods, Comparison of Searching methods. Sorting- Bubble Sort, Insertion Sort, Selection Sort, Heap Sort, Radix Sort. Algorithm Design method-Divide and Conquer method-applications- Quick sort, Merge sort, Comparison of Sorting methods, Model for External Sorting.

UNIT IV

Search Trees-Binary Search Trees, Definition, ADT, Implementation, Operations- Searching, Insertion and Deletion, AVL Trees, Definition, Operations – Insertion and Searching, B-Trees, Definition, B-Tree of order m, operations- insertion and deletion, Introduction to Red-Black and Splay Trees(Elementary treatment), Comparison of Search Trees.

UNIT V

Algorithm Design methods-Greedy method-applications-Kruskal‟s Algorithm for Minimum cost Spanning trees, Job Sequencing with deadlines, Single Source Shortest path problem, Dynamic Programming method-applications-Ordering matrix multiplications, Optimal Binary Search Trees, APSP problem.

Text Processing - Pattern matching algorithm- The Knuth-Morris-Pratt algorithm, Tries-Standard Tries, Compressed Tries, Suffix tries.

TEXT BOOKS:

1. Data structures and Algorithm Analysis in C++, Mark Allen Weiss, 3rd edition, Pearson Education. Ltd.

2. Data structures and Algorithms in C++, Michael T.Goodrich, R.Tamassia and D.Mount, Wiley Student edition , seventh edition, John Wiley and Sons.

REFERENCE BOOKS:

1. Data structures, Algorithms and Applications in C++, S.Sahani, Universities Press.
2. Data structures and algorithms in C++, 3rd Edition, Adam Drozdek, Cengage Learning.
3. Data structures using C and C++, Langsam, Augenstein and Tanenbaum, PHI.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
I Year - II Semester	**4**	**0**	**4**

OPERATING SYSTEMS

Objectives:

- Understand main components of OS and their working. To study the operations performed by OS as a resource manager. Understand the scheduling policies of OS.
- Understand the different memory management techniques.
- Understand process concurrency and synchronization. Understand the concepts of input/output, storage and file management.
- To study different OS and compare their features.

Outcomes:

- Apply optimization techniques for the improvement of system performance.
- Ability to understand the synchronous and asynchronous communication mechanisms in their respective OS.
- Learn about minimization of turnaround time, waiting time and response time and also maximization of throughput with keeping CPU as busy as possible.
- Ability to compare the different OS.

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 State, Process Control Block, Threads, Process Scheduling-Scheduling Queues, Schedulers, Context Switch, Preemptive Scheduling, Dispatcher, Scheduling Criteria, Scheduling algorithms, Multiple-Processor Scheduling, Real-Time Scheduling, Thread scheduling, Case studies: Linux, Windows.

Process Coordination – Process Synchronization, The Critical Section Problem, Peterson‟s solution, Synchronization Hardware, Semaphores, and Classic Problems of Synchronization, Monitors, Case Studies: Linux, Windows.

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, Performance of Demanding Paging, Page Replacement Page Replacement Algorithms, Allocation of Frames, Thrashing, Case Studies: Linux, Windows.

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, Allocation methods, Free-space Management, Directory Implementation, Efficiency and Performance, Case Studies: Linux, Windows.

Mass Storage Structure – Overview of Mass Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling, Disk Management, Swap space Management

UNIT V

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

Protection – System Protection, Goals of Protection, Principles of Protection, Domain of Protection, Access Matrix, Implementation of Access Matrix, Access Control, Revocation of Access Rights, Capability-Based Systems, Language-Based Protection, Case Studies: Linux, Windows.

TEXT BOOKS:

1. Operating System Principles, Abraham Silberschatz, 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.
7. Operating Systems, R.Elmasri, A.G.Carrick and D.Levine, Mc Graw Hill.
8. Operating Systems in depth, T.W. Doeppner, Wiley.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
I Year - II Semester	**4**	**0**	**4**

SCRIPTING LANGUAGES

UNIT I

Introduction to PERL and Scripting Scripts and Programs, Origin of Scripting , Scripting Today, Characteristics of Scripting Languages, Uses for Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines.

UNIT II

Advanced perl: Finer points of looping, pack and unpack, file system, eval, data structures, packages, modules, objects, interfacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Issues.

PHP Basics

PHP Basics- Features, Embedding PHP Code in your Web pages, Outputting the data to the browser, Data types, Variables, Constants, expressions, string interpolation, control structures, Function, Creating a Function, Function Libraries, Arrays, strings and Regular Expressions.

UNIT III

Advanced PHP Programming: PHP and Web Forms, Files, PHP Authentication and Methodologies - Hard Coded, File Based, Database Based, IP Based, Login Administration, Uploading Files with PHP, Sending Email using PHP, PHP Encryption Functions, the Mcrypt package, Building Web sites for the World.

UNIT IV

TCL : TCL Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures, strings, patterns, files, Advance TCL- eval, source, exec and uplevel commands, Name spaces, trapping errors, event driven programs, making applications internet aware, Nuts and Bolts Internet Programming, Security Issues, C Interface.Tk-Visual Tool Kits, Fundamental Concepts of Tk, Tk by example, Events and Binding , Perl-Tk.

UNIT V

Python: Introduction to Python language, python-syntax, statements, functions, Built-in-functions and Methods, Modules in python, Exception Handling.Integrated Web Applications in Python – Building Small, Efficient Python Web Systems, Web Application Framework.

TEXT BOOKS:

1. The World of Scripting Languages, David Barron, Wiley Publications.
2. Python Web Programming, Steve Holden and David Beazley, New Riders Publications.

REFERENCE BOOKS:

1. Open Source Web Development with LAMP using Linux, Apache, MySQL, Perl and PHP, J.Lee and B.Ware (Addison Wesley) Pearson Education.

2. Programming Python, M.Lutz, SPD.

3. PHP 6 Fast and Easy Web Development, Julie Meloni and Matt Telles, Cengage Learning Publications.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
I Year - II Semester	**4**	**0**	**4**

OPERATIONS RESEARCH

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. Model 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.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC autonomous

MCA	**L**	**p**	**C**
I Year – II Semester	**4**	**0**	**4**

SOFTWARE ENGINEERING

UNIT I

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, legacy software, Software myths. A Generic view of process: Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment, personal and team process models. 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. System models: Context Models, Behavioral models, Data models, Object models, structured methods.

UNIT III

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, assessing alternative architectural designs, mapping data flow into a software architecture.
Modeling component-level design: Designing class-based components, conducting component-level design, Object constraint language, designing conventional components. erforming User interface design: Golden rules, User interface analysis and design, interface analysis, interface design steps, Design evaluation.

UNIT IV

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging.
Product metrics: Software Quality, Frame work for Product metrics, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance. Metrics for Process and Products: Software Measurement, Metrics for software quality.

UNIT V

Risk management: Reactive Vs Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM Plan.Quality Management: Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software quality Assurance, Software reliability, The ISO 9000 quality standards.

TEXT BOOKS:

1. Software Engineering A practitioner‟s Approach, Roger S Pressman, sixth edition, McGraw Hill International Edition.
2. Software Engineering, Ian Sommerville, seventh edition, Pearson education.

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.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY

UGC autonomous

	L	p	C
MCA			
I Year – II Semester	**4**	**0**	**4**

DATA STRUCTURES LAB USING C++

List of Sample Problems:

1.Write a C++ program to perform the following operations :

1. Create a Singly linked list of elements.
2. Search for a given element in the above list.
3. Delete an element from the above Singly linked list.
4. Display the contents of the list.

1. Write a C++ program to perform the following operations:

Create a doubly linked list of elements.
Delete an element from the above doubly linked list
Display the contents of the list.

2. Write C++ programs to implement the following using an array.

a) Stack ADT b) Queue ADT

4. Write C++ programs to implement the following using a singly linked list.

a) Stack ADT b) Queue ADT

1. Write a C++ program to convert

a given infix expression into postfix form using stack and

evaluate the Postfix expression using stack.

2. Write C++ programs to implement the deque (double ended queue) ADT using

a)Singly linked list b) Doubly linked list c) an array.

2. Write C++ programs that use recursive functions to traverse the given binary tree in

Preorder b) Inorder and c) Postorder

3. Write a C++ program to perform the following operations:

Construct a binary search tree of elements.
Search for a key element in the above binary search tree.
Delete an element from the above binary search tree.

4. Write C++ programs for implementing the following Searching methods:

a) Linear Search b) Binary Search

10.Write C++ programs for implementing the following sorting methods:

a) Bubble Sort b) Selection Sort c) Insertion Sort

11. Write C++ programs for implementing the following sorting methods:
    a) Merge sort b)Quick sort c) Heap sort d)Radix sort.
12. Write a C++ program to perform the following operation: Insertion into a B-tree

13. Write a C++ program to implement all the functions of a dictionary (ADT) using ashing.

14. Write a C++ program for implementing Knuth-Morris- Pratt pattern matching algorithm.

15. Write C++ programs that use non-recursive functions to traverse the given binary tree in

    a)Preorder b) inorder and c) postorder.

16. Write C++ programs for the depth first and breadth first traversals of a graph.

(Note: You may use STL(Standard Template Libray) in writing the above programs)

TEXT BOOKS :

1. Data Structures A Pseudocode Approach with C++, India Edition, R.F.Gilberg and B.A.Forouzan,Cengage Learning.

2. Data structures with C++,J.R.Hubbard,Schaum‟s Outlines,TMH.
3. Data Structures and STL, W.J.Collins,Mc Graw Hill,International edition.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	L	P	C
I Year - II Semester	4	0	4

SCRIPTING LANGUAGE (PYTHON) LAB

Course Objectives:

1. To be able to introduce core programming basics and program design with functions using Python programming language.
2. To understand a range of Object-Oriented Programming, as well as in-depth data and information processing techniques.
3. To understand the high-performance programs designed to strengthen the practical expertise.

Course Outcomes:

1. Student should be able to understand the basic concepts scripting and the contributions of scripting language
2. Ability to explore python especially the object-oriented concepts, and the built-in objects of Python.
3. Ability to create practical and contemporary applications such as TCP/IP network programming, Web applications, discrete-event simulations

Prerequisites: Students should install Python on Linux platform.

List of Programs:

i) Write a program to demonstrate different number data types in Python.
ii) Write a program to perform different Arithmetic Operations on numbers in Python.

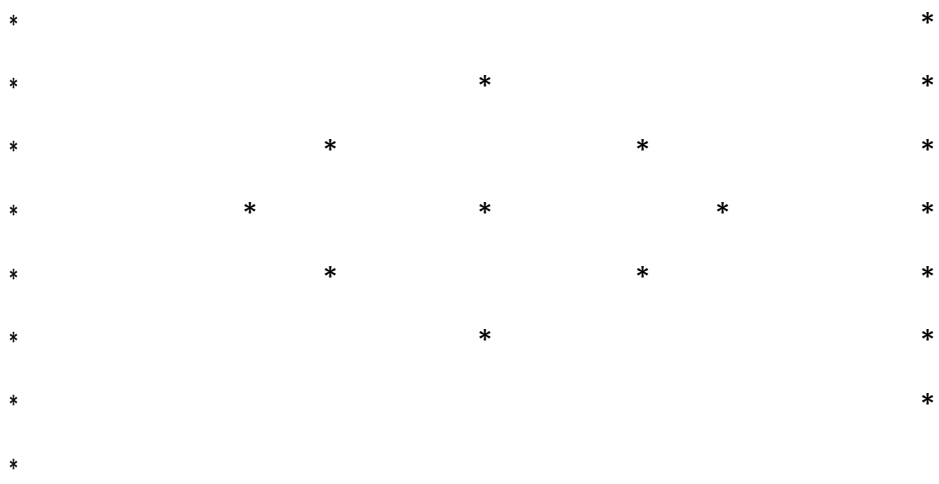
iii) Write a program to create, concatenate and print a string and accessing sub-string from a given string.

iv) Write a python script to print the current date in the following format “Sun May 29 02:26:23 IST 2017”
v) Write a program to create, append, and remove lists in python.
vi) Write a program to demonstrate working with tuples in python.
vii) Write a program to demonstrate working with dictionaries in python.

viii) Write a python program to find largest of three numbers.

ix) Write a Python program to convert temperatures to and from Celsius, Fahrenheit.
[Formula : c/5 = f-32/9]

10. Write a Python program to construct the following pattern, using a nested for loop



11. Write a Python script that prints prime numbers less than 20.
12. Write a python program to find factorial of a number using Recursion.
13. Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the

    Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).

14. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.
15. Write a python program to define a module and import a specific function in that module to another program.
16. Write a script named **copyfile.py**. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.
17. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.
18. Write a Python class to convert an integer to a roman numeral.

19. Write a Python class to implement pow(x, n)
20. Write a Python class to reverse a string word by word.

TEXT BOOKS:

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

2. Python Web Programming, Steve Holden and David Beazley, New Riders Publications.

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
I Year - II Semester	**4**	**0**	**4**

OPERATING SYSTEMS LAB

EXPERIMENT 1: Simulate the following CPU scheduling algorithms

a) Round Robin b) SJF.

EXPERIMENT 2: Simulate the following CPU Scheduling algorithms

a) FCFS b) Priority.

EXPERIMENT 3: Simulate all file allocation strategies.

a) Sequential b) Indexed c) Linked.

EXPERIMENT 4: Simulate MVT and MFT.

EXPERIMENT 5: Simulate the following File Organization Techniques

a) Single level directory b) Two level.

EXPERIMENT 6: Simulate the following File Organization Techniques

a) Hierarchical b) DAG.

EXPERIMENT 7: Simulate Disk scheduling algorithms a) FCFS b) SSTF c) SCAN d) C-SCAN e) LOOK.

EXPERIMENT 8: Simulate Bankers Algorithm for Dead Lock Avoidance.

EXPERIMENT 9: Simulate Bankers Algorithm for Dead Lock Prevention.

EXPERIMENT 10: Simulate all page replacement algorithms

a) FIFO b) LRU c) LFU Etc. …

EXPERIMENT 11: Simulate Paging Technique of memory management.

EXPERIMENT 12: Simulate on Allocation of Frames

REFERENCE BOOKS:

1. An Introduction to Operating Systems, P.C.P Bhatt, 2nd edition, PHI.
2. Unix System Programming Using C++, Terrence Chan, PHI/Pearson.
3. Modern Operating Systems, Andrew S Tanenbaum, 3rd Edition, PHI

J.B.INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - I Semester	**4**	**0**	**4**

DATA BASE MANAGEMENT SYSTEMS

Prerequisites

- A course on "Advanced Data Structures"

Course Objectives

- To understand the basic concepts and the applications of database systems.
  To master the basics of SQL and construct queries using SQL.
- Topics include data models, database design, relational model, relational algebra, transaction control, concurrency control, storage structures and access techniques.

Course Outcomes

- Gain knowledge of fundamentals of DBMS, database design and normal forms
  Master the basics of SQL for retrieval and management of data.
- Be acquainted with the basics of transaction processing and concurrency control.
  Familiarity with database storage structures and access techniques

UNIT I

Database System Applications, Purpose of Database Systems, View of Data – Data Abstraction, Instances and Schemas, Data Models – the ER Model, Relational Model, Other Models – Database Languages – DDL, DML, Database Access from Applications Programs, Transaction Management, Data Storage and Querying, Database Architecture, Database Users and Administrators, History of Data base Systems.

Introduction to Data base design, ER diagrams, Beyond ER Design, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model, Conceptual Design for Large enterprises. Relational Model: Introduction to the Relational Model – Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design, Introduction to Views – Destroying /altering Tables and Views.

UNIT II

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

Form of Basic SQL Query – Examples of Basic SQL Queries, Introduction to Nested Queries, Correlated Nested Queries, Set – Comparison Operators, Aggregate Operators, NULL values – Comparison using Null values – Logical connectives – AND, OR and NOT – Impact on SQL Constructs, Outer Joins, Disallowing NULL values, Complex Integrity Constraints in SQL Triggers and Active Data bases.

UNIT III

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

UNIT IV

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

Concurrency Control: Serializability, and recoverability – Introduction to Lock Management – Lock Conversions, Dealing with Deadlocks, Specialized Locking Techniques – Concurrency Control without Locking. Crash recovery: Introduction to Crash recovery, Introduction to ARIES, the Log, Other Recovery

related Structures, the Write-Ahead Log Protocol, Check pointing, recovering from a System Crash, Media recovery.

UNIT V

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

Storing data: Disks and Files -The Memory Hierarchy – Redundant Arrays of Independent Disks.Tree Structured Indexing: Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM). B+ Trees: A Dynamic Index Structure, Search, Insert, and Delete.Hash Based Indexing: Static Hashing, Extendable hashing, Linear Hashing, Extendible Vs. Linear Hashing.

TEXT BOOKS:

1. Data base Management Systems, Raghu Ramakrishnan, Johannes Gehrke, TMH, 3rd Edition, 2003.

2. Data base System Concepts, A.Silberschatz, H.F. Korth, S.Sudarshan, McGraw hill, VI edition, 2006.
3. Fundamentals of Database Systems 5th edition, Ramez Elmasri, Shamkant B.Navathe, Pearson Education, 2008.

REFERENCE BOOKS:

1. Database Management System Oracle SQL and PL/SQL, P.K.Das Gupta, PHI.
2. Database System Concepts, Peter Rob & Carlos Coronel, Cengage Learning, 2008.
3. Database Systems, A Practical approach to Design Implementation and Management Fourth edition, Thomas Connolly, Carolyn Begg, Pearson education.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - I Semester	**4**	**0**	**4**

COMPUTER NETWORKS

Prerequisites

A course on "Computer Programming and Data Structures"
A course on "Design and Analysis of Algorithms"

Course Objectives

The objective of the course is to equip the students with a general overview of the concepts and fundamentals of computer networks.
Familiarize the students with the standard models for the layered approach to communication between machines in a network and the protocols of the various layers.
The following topics are included: Reference models, the physical layer (transmission media); the data link layer (error detection and correction, point-to-point protocols); the medium access layer protocols; the network layer (routing algorithms, congestion control); internetworking (addressing, internetwork routing and protocols, quality of service); the transport layer (connection-oriented transport layer services and protocols); application layer protocols

Course Outcomes

Gain the knowledge of the basic computer network technology.
Gain the knowledge of the functions of each layer in the OSI and TCP/IP reference model.
Obtain the skills of subnetting and routing mechanisms.
Familiarity with the essential protocols of computer networks, and how they can be appled in network design and implementation.

UNIT I

Overview of the Internet: Protocol, Layering Scenario, TCP/IP Protocol Suite: The OSI Model, Internet history standards and administration; Comparison of the OSI and TCP/IP reference model

Data Link Layer – design issues, CRC Codes, Elementary Data link Layer protocols, sliding window protocol

UNIT II

Multiple Access Protocols –ALOHA, CSMA, Collision free protocols, Ethernet- Physical Layer, Ethernet Mac Sub layer.

Data link layer switching & use of bridges, learning bridges, spanning tree bridges, repeaters, hubs, bridges, switches, routers and gateways.

UNIT III

Network Layer: Network Layer Design issues, store and forward packet switching connection less and connection oriented networks-routing algorithms-optimality principle, shortest path, flooding, Distance Vector Routing, Count to Infinity Problem, Hierarchical Routing, Congestion control algorithms, admission control.

UNIT IV

Internetworking: Tunneling, Internetwork Routing, Packet fragmentation, IPv4, Ipv6 Protocol, IP addresses, CIDR, IMCP, ARP, RARP, DHCP.

Transport Layer: Services provided to the upper layers elements of transport protocol-addressing connection establishment, connection release, Crash Recovery.

UNIT V

The Internet Transport Protocols UDP-RPC, Real Time Transport Protocols, The Internet Transport Protocols- Introduction to TCP, The TCP Service Model, The TCP Segment Header, The Connection Establishment, The TCP Connection Release, The TCP Connection Management Modeling, The TCP

Sliding Window, The TCP Congestion Control, The future of TCP.
Application Layer-Introduction, providing services, Applications layer paradigms, Client server model, Standard client-server application-HTTP, FTP, electronic mail, TELNET, DNS, SSH.

TEXT BOOKS:
1. Computer Networks, Andrew S. Tanenbaum, David J Wetherall, Pearson Education, 5th Edition.
2. Computer Networks A Top-Down Approach, Behrouz A Forouzan, Firouz Mosharraf, TMH.

REFERENCE BOOKS:
1. An Engineering Approach to Computer Networks-S.Keshav, 2nd Edition, Pearson Education.
2. Understanding communications and Networks, 3rd Edition, W.A.Shay, Cengage Learning.
3. Computer Networks, L.L.Peterson and B.S.Davie, 4th edition, ELSEVIER.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - I Semester	**4**	**0**	**4**

Statistical Programming with 'R'

Objectives:

- To learn the fundamentals of R
- To understand performing operations on complex data types
- To understanding how to use the four object systems in R
- To enable the students to use existing functional programming tools
- Explains how to create functions that use non-standard evaluation in a principled way
- Shows how to use profiling to pinpoint performance bottlenecks and hot to convert slow R functions to fast C++ equivalents.

UNIT-I

Introduction, how to run R, R Sessions and Functions, Basic Math, Variables, Data Types, Vectors, Conclusion, Advanced Data Structures, Data Frames, Lists, Matrices, Arrays, Classes**.**

UNIT-II

R Programming Structures, Control Statements, Loops, - Looping Over Nonfactor Sets,-If-Else, Arithmetic and Boolean Operators and values, Default Values for Argument, Return Values,
Deciding Whether to explicitly call return- Returning Complex Objects, Functions are Objective, No Pointers in R, Recursion, A Quick sort Implementation-Extended Extended Example: A Binary Search Tree.

UNIT-III

Doing Math and Simulation in R, Math Function, Extended Example Calculating Probability- Cumulative Sums and Products-Minima and Maxima- Calculus, Functions Fir Statistical Distribution, Sorting, Linear Algebra Operation on Vectors and Matrices,
Extended Example: Vector cross Product- Extended Example: Finding Stationary Distribution of Markov Chains, Set Operation, Input /output, Accessing the Keyboard and Monitor, Reading and writer Files,

UNIT-IV

Graphics, Creating Graphs, The Workhorse of R Base Graphics, the plot () Function – Customizing Graphs, Saving Graphs to Files.

UNIT-V

Probability Distributions, Normal Distribution- Binomial Distribution- Poisson Distributions Other Distribution, Basic Statistics, Correlation and Covariance, T-Tests, -ANOVA.
Linear Models, Simple Linear Regression, -Multiple Regression Generalized Linear Models, Logistic Regression, - Poisson Regression- other Generalized Linear Models-Survival Analysis, Nonlinear Models, Spines- Decision- Random Forests.

TEXT BOOKS:

1) The Art of R Programming, Norman Matloff, Cengage Learning
2) R for Everyone, Lander, Pearson

REFERENCE BOOKS:

1) R Cookbook, PaulTeetor, Oreilly.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - I Semester	**4**	**0**	**4**

OBJECT ORIENTED ANALYSIS AND DESIGN USING UML

Prerequisites:

- Software Engineering

Course Objectives:

- To train students on object modelling To apply unified process phases
- To apply unified modelling language for software design of any applications To study case studies for OOAD

Course Outcomes:

- Will be able to use UML notations
- Can apply unified process in software development
- Will be able to perform analysis and design using object modelling

UNIT I

Introduction to UML: Importance of modelling, principles of modelling, object oriented modelling, conceptual model of the UML, Architecture, and Software Development Life Cycle.

UNIT II

Basic Structural Modelling: Classes, Relationships, common Mechanisms, diagrams.

Advanced Structural Modelling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. Class & Object Diagrams: Terms, concepts, modelling techniques for Class & Object Diagrams.

UNIT III

Basic Behavioural Modelling-I: Interactions, Interaction diagrams.

Basic Behavioural Modelling-II: Use cases, Use case Diagrams, Activity Diagrams.

UNIT IV

Advanced Behavioural Modelling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams.

Architectural Modelling: Component, Deployment, Component diagrams and Deployment diagrams.

UNIT V

Patterns and Frameworks, Artifact Diagrams. Case Study: The Unified Library application

TEXT BOOKS:

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:

3. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.
5. Pascal Roques: Modeling Software Systems Using UML2, WILEY-Dreamtech India Pvt. Ltd.

6. Atul Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - I Semester	**4**	**0**	**4**

JAVA PROGRAMMING

Prerequisites

- A course on "Computer Programming & Data Structures"

Course Objectives

- Introduces object oriented programming concepts using the Java language.
- Introduces the principles of inheritance and polymorphism; and demonstrates how they relate to the design of abstract classes
- Introduces the implementation of packages and interfaces
- Introduces exception handling, event handling and multithreading
- Introduces the design of Graphical User Interface using applets and swings

Course Outcomes

- Develop applications for a range of problems using object-oriented programming techniques Design simple Graphical User Interface applications

UNIT I

Java Basics - History of Java, Java buzzwords, comments, data types, variables, constants, scope and life time of variables, operators, operator hierarchy, expressions, type conversion and casting, enumerated types, control flow-block scope, conditional statements, loops, break and continue statements, simple java program, arrays, input and output, formatting output,
Review of OOP concepts, encapsulation, inheritance, polymorphism, classes, objects, constructors, methods, parameter passing, static fields and methods, access control, this reference, overloading methods and constructors, recursion, garbage collection, building strings, exploring string class, Enumerations, autoboxing and unboxing, Generics.

UNIT II

Inheritance – Inheritance concept, benefits of inheritance, Super classes and Sub classes, Member access rules, Inheritance hierarchies, super uses, preventing inheritance: final classes and methods, casting, polymorphism- dynamic binding, method overriding, abstract classes and methods, the Object class and its methods.
Interfaces – Interfaces vs. Abstract classes, defining an interface, implementing interfaces, accessing implementations through interface references, extending interface. **Inner classes**– Uses of inner classes, local inner classes, anonymous inner classes, static inner classes, examples.
Packages-Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages.

UNIT III

Data structures creation and manipulation in java – Introduction to Java Collections, Overview of Java Collection frame work, commonly used Collection classes– ArrayList, LinkedList, HashSet, HashMap, TreeMap, Collection Interfaces – Collection, Set, List, Map, Legacy Collection classes – Vector, Hashtable, Stack, Dictionary(abstract), Enumeration interface, Iteration over Collections – Iterator interface, ListIterator interface. Other Utility classes – String Tokenizer, Formatter, Random, Scanner, Observable, java.util.
Files – streams- byte streams, character streams, text Input/output, binary input/output, random

access file operations, File management using File class, java.io. **Networking** – Introduction, Manipulating URLs, Ex. Client/Server Interaction with Stream Socket Connections, Connectionless Client/Server Interaction with Datagrams, java.net.

UNIT IV

Exception handling – Dealing with errors, benefits of exception handling, the classification of exceptions- exception hierarchy, checked exceptions and unchecked exceptions, usage of try, catch, throw, throws and finally, rethrowing exceptions, exception specification, built in exceptions, creating own exception sub classes. Guide lines for proper use of exceptions.

Multithreading - Differences between multiple processes and multiple threads, thread states, creating threads, interrupting threads, thread priorities, synchronizing threads, interthread communication, thread groups, daemon threads.

UNIT V

GUI Programming with Java - The AWT class hierarchy, Introduction to Swing, Swing vs. AWT, MVC architecture, Hierarchy for Swing components, Containers – Top-level containers – JFrame, JApplet, JWindow, JDialog, Light weight containers – JPanel, A simple swing application, Overview of several swing components- Jbutton, JToggleButton, JCheckBox, JRadioButton, JLabel, JTextField, JTextArea, JList, JComboBox, JMenu, Java's Graphics capabilities – Introduction, Graphics contexts and Graphics objects, color control, Font control, Drawing lines, rectangles and ovals, Drawing arcs, Layout management - Layout manager types – border, grid, flow, box.

Event Handling - Events, Event sources, Event classes, Event Listeners, Relationship between Event sources and Listeners, Delegation event model, Semantic and Low-level events, Examples: handling a button click, handling mouse and keyboard events, Adapter classes.

Applets – Inheritance hierarchy for applets, differences between applets and applications, life cycle of an applet - Four methods of an applet, Developing applets and testing, passing parameters to applets, applet security issues.

TEXT BOOKS:

1. Java: the complete reference, 8th edition, Herbert Schildt, TMH.

2. Java for Programmers, P.J.Deitel and H.M.Deitel, Pearson education / Java: How to Program P.J.Deitel and H.M.Deitel, 8th edition, PHI.

REFERENCE BOOKS:

1. Java Programming, D.S.Malik, Cengage Learning.
   Core Java, Volume 1-Fundamentals, eighth edition, Cay S.Horstmann and Gary Cornell, Pearson Education.
2. An introduction to Java programming and object oriented application development, R.A. Johnso Cengage Learning.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - I Semester	**0**	**2**	**2**

DATABASE MANAGEMENT SYSTEMS LAB

Pre-requisites:

- Co-requisite of course "Database Management Systems"

Course Objectives:

- Introduce ER data model, database design and normalization Learn SQL basics for data definition and data manipulation

Course Outcomes:

- Design database schema for a given application and apply normalization
- Acquire skills in using SQL commands for data definition and data manipulation.
- Develop solutions for database applications using procedures, cursors and triggers

List of Sample Problems:

1. Creation, altering and dropping 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 : (CLENTNO, 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)

x) Insert data into their respective table.

xi) 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 Client No '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 master.

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
SalesOrderDetails

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'.
Count the total no of orders.
Calculate the average price of all the products.

j. Determine the maximum and minimum products prices. Rename the output as max_price and min_price respectively.Count the no of products having price less than or equal to 500.

k. 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'.

h) Exercice 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.

1. **Creating Views**
2. **Writing Assertions**
3. **Writing Triggers**
4. **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).

TEXT BOOKS:

1. Introduction to SQL, Mastering the relational DB languages, IV th edition, Rick F. Vander Lans, Pearson ed., 2007.

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)

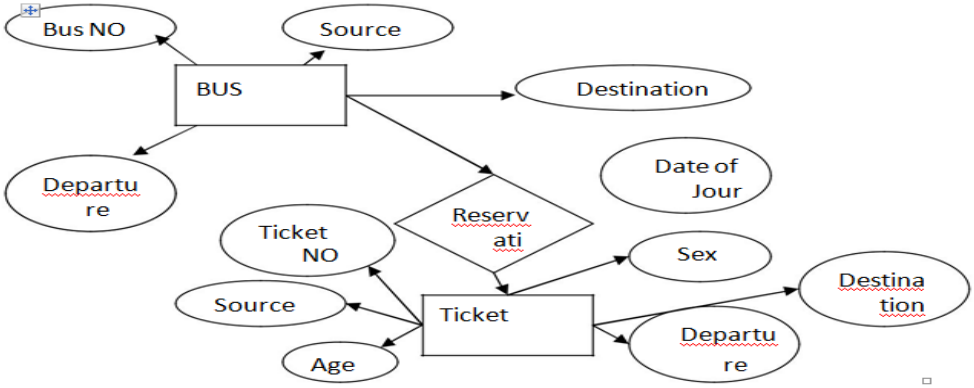
Apart from the above mentioned entities you can identify more. The above mentioned are few.

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

Name	Age	Sex	Address	Passport ID	Ticket id

You can do the second and third normal forms if required. Anyhow 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 an 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.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - I Semester	**0**	**2**	**2**

JAVA PROGRAMMING LAB

Prerequisites

- A course on "Computer Programming & Data Structures"

Co-requisite

- A Course on "Object-Oriented Programming Through Java"

Course Objectives

- Introduces object oriented programming concepts using the Java language.
- Introduces the principles of inheritance and polymorphism; and demonstrates how they relate to the design of abstract classes
- Introduces the implementation of packages and interfaces
- Introduces exception handling, event handling and multithreading
- Introduces the design of Graphical User Interface using applets and swings

List of Sample Problems:

1. a) 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.
b) 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 sum 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.
2. a) Write a Java program that prompts the user for an integer and then prints out all prime numbers up to that integer.

b) Write a Java program to multiply two given matrices.
c) 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 of java.util)
3. Write a Java program to find both the largest and smallest number in a list of integers.
4. Write a Java program to illustrate method overloading.
5. Write a Java program that implements the Sieve of Eratosthenes to find prime numbers.
6. Write a Java program to sort a list of names in ascending order.
7. Write a Java program to implement the matrix ADT using a class. The operations supported by this ADT are:
a) Reading a matrix. c) Addition of matrices.
b) Printing a matrix. d) Multiplication of matrices.
8. Write a Java Program to solve Towers of Hanoi problem .
9. Write a Java Program that uses a recursive function to compute ncr. (Note: n and r values are given.)
10. Write a Java program to perform the following operations:
a) Concatenation of two strings.
b) Comparison of two strings.

11. Implement the complex number ADT in Java 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) Multiplication of complex numbers.
d) Addition of Complex numbers.

12. Write a Java program that makes frequency count of letters in a given text.
13. Write a Java program that uses functions to perform the following operations :
    a) Inserting a sub-string in to the given main string from a given position.
    b) Deleting n characters from a given position in a given string.
14. a) Write a Java program that checks whether a given string is a palindrome or not.
    Ex: MADAM is a palindrome.
    b) Write a Java program to make frequency count of words in a given text.
15 .a) 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.
    c) Write a Java program that reads a file and displays the file on the screen, with a line number before each line.
    c) Write a Java program that displays the number of characters, lines and words in a text file.

    d) Write a Java 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.

16. Write a Java program that illustrates the following
    a) Creation of simple package. b) Accessing a package. c) Implementing interfaces.
17. a) Write a Java program that creates three threads. First thread displays “Good Morning” every one second, the second thread displays “Hello” every two seconds and the third thread displays “Welcome” every three seconds.

    b) Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.
18. Write Java programs that illustrates the following
    a) Handling predefined exceptions
    b) Handling user defined exceptions
19. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result.
20. a) Develop an applet in Java that displays a simple message.
    b) Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named “Compute” is clicked.

21. Write a Java 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 and integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.

22. Write a Java program that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result, and then sends the result back to the client. The client displays the result on the console. For ex: The data sent from the client is the radius of a circle, and the result produced by the server is the area of the circle. (Use java.net)

23. a) Write a Java 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.

b) Write a Java program that allows the user to draw lines, rectangles and ovals.

24 . a) Write a Java program to create an abstract class named Shape that contains an empty method named numberOfSides ( ).Provide three classes named Trapezoid, Triangle and Hexagon such that each one of the classes extends the class Shape. Each one of the classes contains only the method numberOfSides ( ) that shows the number of sides in the given geometrical figures.

b) Suppose that a table named Table.txt is stored in a text file. The firstline in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Jtable component.

25. Write a Java program for handling Key events.
26. Write a Java program for handling mouse events. (Use Adapter classes).

27. Write Java programs that use both recursive and non-recursive functions for implementing the following searching methods:

    a) Linear search b) Binary search

28. Write Java programs to implement the List ADT using arrays and linked lists.
29. Write Java programs to implement the Stack ADT and Queue ADT using arrays.
30. Write Java programs to implement the following using a singly linked list.
    a) Stack ADT b) Queue ADT .
31. Write Java programs for implementing the following sorting methods: a) Bubble sort c) Quick sort
    b) Selection sort

32. Write a Java Program to perform the following:
    a) Create a binary search tree of elements.
    b) Search the above binary search tree for a key value.
    c) Traverse the above binary search tree in inorder.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - II Semester	**4**	**0**	**4**

WEB TECHNOLOGIES

Course Objectives:

- To introduce PHP language for server side scripting
- To introduce XML and processing of XML Data with Java
- To introduce Server side programming with Java Servlets and JSP To introduce Client side scripting with Java script and AJAX.

Course Outcomes:

- The Student is expected to gain knowledge of client side scripting, validation of forms and AJAX programming have understanding of server side scripting with PHP language have understanding of what is XML and how to parse and use XML Data with Java To introduce Server side programming with Java Servlets and JSP

UNIT I

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.

File Handling in PHP: File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories.

UNIT II

XML: Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemas, Document Object Model, XHTML.

Parsing XML Data - DOM and SAX Parsers in java.

UNIT III

Introduction to Servlets: Common Gateway Interface (CGI), Lifecycle of a Servlet, deploying a servlet, The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions, connecting to a database using JDBC.

UNIT IV

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 and session for session tracking, connecting to database in JSP.

UNIT V

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

TEXT BOOKS:

1. Web Technologies, Uttam K Roy, Oxford University Press
2. The Complete Reference PHP – Steven Holzner, Tata McGraw-Hill .

REFERENCE BOOKS:

1. Web Programming, building internet applications, Chris Bates 2nd edition, Wiley Dreamtech.
2. Java Server Pages –Hans Bergsten, SPD O'Reilly.
3. Java Script, D.Flanagan, O'Reilly, SPD.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - II Semester	**4**	**0**	**4**

LINUX PROGRAMMING

Prerequisites

- Familiarity with using Unix Programming environment and having a good working knowledge of the C programming language.

Objectives:

- To understand the LINUX system structure.
- To understand and use command line shell.
- To make effective use of UNIX utilities and Shell scripting language such as bash.
- To produce programs similar to standard UNIX utilities such as ls, mv, cp etc. using Unix system calls.
- To develop the skills necessary for Unix systems programming including file system programming, process and signal management, and inter process communication.
- To develop the basic skills required to write network programs using Sockets.

Outcomes:

- Work confidently in Linux environment.
- Work with shell script to automate different tasks as Linux administration

UNIT I

Linux Utilities-File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking commands, Filters, Text processing utilities and Backup utilities. Sed-Scripts, Operation, Addresses, Commands, Applications, awk- Execution, Fields and Records, Scripts, Operation, Patterns, Actions, Associative Arrays, String and Mathematical functions, System commands in awk, Applications..
Introduction, shell responsibilities, 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, interrupt processing, functions, debugging shell scripts. Review of C programming concepts-arrays, strings (library functions), pointers, function pointers, structures, unions, libraries in C.

UNIT II

Files and Directories- File Concept, File types, File System Structure, file metadata-Inodes, kernel support for files, system calls for file I/O operations- open, create, read, write, close, lseek, dup2, 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.
Directories-Creating, removing and changing Directories-mkdir, rmdir, chdir, obtaining current working directory-getcwd, Directory contents, Scanning Directories-opendir, readdir, closedir, rewinddir, seekdir, telldir functions.

UNIT III

Process – Process concept, Layout of a C program image in main memory, Process environment-environment list, environment variables, getenv, setenv, Kernel support for process, process identification, process hierarchy, process states, process control - process creation, replacing a process image, waiting for a process, process termination, zombie process, orphan process,
system call interface for process management-fork, vfork, exit, wait, waitpid, exec family, system, I/O redirection, Process Groups, Sessions and Controlling Terminal, Differences between threads and processes. 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.

UNIT IV

Interprocess Communication - Introduction to IPC, IPC between processes on a single computer system, IPC between processes on different systems, 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. Message Queues- Kernel support for messages, APIs for message queues, client/server example. Semaphores-Kernel support for semaphores, APIs for semaphores, file locking with semaphores.

UNIT V

Shared Memory- Kernel support for shared memory, APIs for shared memory, shared memory example. Sockets- Introduction to Berkeley Sockets, IPC over a network, Client-Server model, Socket address structures (Unix domain and Internet domain),

Socket system calls for connection oriented protocol and connectionless protocol, example-client/server programs-Single Server-Client connection, Multiple simultaneous clients, Comparison of IPC mechanisms.

TEXT BOOKS:

1. Unix System Programming using C++, T.Chan, PHI.
2. Unix Concepts and Applications, 4th Edition, Sumitabha Das, TMH, 2006.

REFERENCE BOOKS:

1. Linux System Programming, Robert Love, O'Reilly, SPD, rp-2007.
2. Unix for programmers and users, 3rd Edition, Graham Glass, King Ables, Pearson Education, 2003.
3. Advanced Programming in the Unix environment, 2nd Edition, W.R.Stevens, Pearson Education.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - II Semester	**4**	**0**	**4**

Machine Learning
(Core Elective – I)

Prerequisites:

- Students are expected to have knowledge in linear signals and systems, Fourier Transform, basic linear algebra, basic probability theory and basic programming techniques; knowledge of Digital Signal Processing is desirable.
- A course on "Computational Mathematics"
- A course on "Computer Oriented Statistical Methods"

Course Objectives:

- This course introduces fundamental concepts, theories, and algorithms for pattern recognition and machine learning.
- Topics include: Pattern Representation, Nearest Neighbour Based Classifier, Bayes Classifier, Hidden Markov Models, Decision Trees, Support Vector Machines, Clustering, and an application of hand-written digit recognition.

Course Outcomes:

- Understand the theory, benefits, inadequacies and possible applications of various machine learning and pattern recognition algorithms
- Identify and employ suitable machine learning techniques in classification, pattern recognition, clustering and decision problems.

UNIT – I

Introduction - Well-posed learning problems, designing a learning system Perspectives and issues in machine learning **Concept learning and the general to specific ordering** – Introduction, A concept learning task, concept learning as search, Find-S: Finding a Maximally Specific Hypothesis, Version Spaces and the Candidate Elimination algorithm, Remarks on Version Spaces and Candidate Elimination, Inductive Bias.

Decision Tree Learning – Introduction, Decision Tree Representation, Appropriate Problems for Decision Tree Learning, The Basic Decision Tree Learning Algorithm Hypothesis Space Search in Decision Tree Learning, Inductive Bias in Decision Tree Learning, Issues in Decision Tree Learning.

UNIT - II

Artificial Neural Networks Introduction, Neural Network Representation, Appropriate Problems for Neural Network Learning, Perceptions, Multilayer Networks and the Back propagation Algorithm. Discussion on the Back Propagation Algorithm, An illustrative Example: Face Recognition

Evaluation Hypotheses – Motivation, Estimation Hypothesis Accuracy, Basics of Sampling Theory, A General Approach for Deriving Confidence Intervals, Difference in Error of Two Hypotheses, Comparing Learning Algorithms.

UNIT - III

Bayesian learning - Introduction, Bayes Theorem, Bayes Theorem and Concept Learning Maximum Likelihood and Least Squared Error Hypotheses, Maximum Likelihood Hypotheses for Predicting Probabilities, Minimum Description Length Principle , Bayes Optimal Classifier, Gibs Algorithm,

Naïve Bayes Classifier, An Example: Learning to Classify Text, Bayesian Belief Networks, EM Algorithm.

Computational Learning Theory – Introduction, Probably Learning an Approximately Correct Hypothesis, Sample Complexity for Finite Hypothesis Space, Sample Complexity for Infinite Hypothesis Spaces, The Mistake Bound Model of Learning. **Instance-Based Learning –** Introduction, k-Nearest Neighbor Learning, Locally Weighted Regression, Radial Basis Functions, Case-Based Reasoning, Remarks on Lazy and Eager Learning.

UNIT - IV

Pattern Comparison Techniques, Temporal patterns, Dynamic Time Warping Methods, Clustering, Codebook Generation, Vector Quantization.

Pattern Classification: Introduction to HMMS, Training and Testing of Discrete Hidden Markov Models and Continuous Hidden Markov Models, Viterbi Algorithm, Different Case Studies in Speech recognition and Image Processing

UNIT - V

Analytical Learning – Introduction, Learning with Perfect Domain Theories: PROLOG-EBG Remarks on Explanation-Based Learning, Explanation-Based Learning of Search Control Knowledge, Using Prior Knowledge to Alter the Search Objective, Using Prior Knowledge to Augment Search Operations.

Combining Inductive and Analytical Learning – Motivation, Inductive-Analytical Approaches to Learning, Using Prior Knowledge to Initialize the Hypothesis.

TEXT BOOKS:

1. Machine Learning – Tom M. Mitchell, MGH

2. Fundamentals of Speech Recognition By Lawrence Rabiner and Biing – Hwang Juang.

REFERENCE BOOKS:

1. Machine Learning : An Algorithmic Perspective, Stephen Marsland, Taylor & Francis

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - II Semester	**4**	**0**	**4**

DATA WAREHOUSING AND DATA MINING
(Core Elective – I)

Prerequisites:

- Students are expected to have knowledge in linear signals and systems, Fourier Transform, basic linear algebra, basic probability theory and basic programming techniques; knowledge of Digital Signal Processing is desirable.
- A course on "Computational Mathematics"
- A course on "Computer Oriented Statistical Methods"

Course Objectives:

- This course introduces fundamental concepts, theories, and algorithms for pattern recognition and machine learning.
- Topics include: Pattern Representation, Nearest Neighbour Based Classifier, Bayes Classifier, Hidden Markov Models, Decision Trees, Support Vector Machines, Clustering, and an application of hand-written digit recognition.

Course Outcomes:

- Understand the theory, benefits, inadequacies and possible applications of various machine learning and pattern recognition algorithms
- Identify and employ suitable machine learning techniques in classification, pattern recognition, clustering and decision problems.

UNIT- I

Data Mining Data-Types of Data-, Data Mining Functionalities- Interestingness Patterns-Classification of Data Mining systems-
Data mining Task primitives -Integration of Data mining system with a Data warehouse-Major issues in Data Mining-Data Preprocessing.

UNIT - II

Data Warehouse And Business Analysis
Data Warehouse-Data Warehouse Architecture- Multidimensional Data Model-
Data cube and OLAP Technology-Data Warehouse Implementation -DBMS schemas for Decision support - Efficient methods for Data cube computation.

UNIT- III

Association Rule Mining And Classification Mining Frequent Patterns-Associations and correlations-Mining Methods- Mining Various kinds of Association Rules- Correlation Analysis- Constraint based Association mining.
Classification and Prediction- Basic concepts-Decision tree induction-Bayesian classification, Rule-based classification classification by Back propagation,-Support vector machines-.Associative Classification, Lazy learners-Other classification methods – Prediction.

UNIT- IV

Clustering and Applications Cluster analysis-Types of Data in Cluster Analysis-Categorization of Major Clustering Methods-Partitioning Methods,-Hierarchical Methods- Density-Based Methods,-Grid-Based Methods,-Model- Based Clustering Methods- Clustering high dimensional data-Constraint- Based cluster analysis-Outlier Analysis

UNIT - V

Mining Data Streams, Time-Series And Sequence Data Basic concepts- Mining data streams-Mining Time-series data--Mining sequence patterns in Transactional databases-.Mining Object- Spatial-Multimedia-Text and Web data- Spatial Data mining- Multimedia Data mining--Text Mining- Mining the World Wide Web.

TEXT BOOKS:

1. Data Mining – Concepts and Techniques - JIAWEI HAN & MICHELINE KAMBER, Elsevier.

2. Data Warehousing, Data Mining &OLAP- Alex Berson and Stephen J. Smith- Tata McGraw-Hill Edition, Tenth reprint 2007

REFERENCES:

1. Building the Data Warehouse- W. H. Inmon, Wiley Dreamtech India Pvt. Ltd.
2. Data Mining Introductory and Advanced topics –MARGARET H DUNHAM, PEA.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - II Semester	**4**	**0**	**4**

INFORMATION SECURITY
(Core Elective-I)

Prerequisites

- A Course on "Computer Networks, Mathematics

Course Objectives

- Understand information security's importance in our increasingly computer-driven world.. Master the key concepts of information security and how they "work."
- To understand the fundamentals of Cryptography
- To understand the various key distribution and management schemes
- To understand how to deploy encryption techniques to secure data in transit across data networks
- To apply algorithms used for secure transactions in real world applications

Course Outcomes

- Ability to demonstrate the knowledge of cryptography and network security concepts and applications.
- Apply security principles in system design.
- Ability to identify and investigate vulnerabilities and security threats and mechanisms to counter them.

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

Authentication Applications: Kerberos, X.509 Authentication Service, Public – Key Infrastructure, Biometric Authentication.

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,4th Edition.
2. Cryptography and Network Security : Atul Kahate, Mc Graw Hill, 2nd Edition.

REFERENCE BOOKS:

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1stEdition.
2. Cryptography and Network Security : Forouzan Mukhopadhyay, Mc Graw Hill, 2nd Edition.
3. Information Security, Principles and Practice: Mark Stamp, Wiley India.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - II Semester	**4**	**0**	**4**

DISTRIBUTED SYSTEMS
(Core Elective II)

Prerequisites:

- A course on " Operating Systems"

Course Objectives:

- This course provides an insight into Distributed systems.
- Topics include- Peer to Peer Systems, Transactions and Concurrency control, Security and Distributed shared memory

Course Outcomes:

- Ability to understand Transactions and Concurrency control.
  Ability to understand Security issues.
- Understanding Distributed shared memory.

UNIT I

Characterization of Distributed Systems-Introduction, Examples of Distributed systems, Resource sharing and web, challenges, System models-Introduction, Architectural and Fundamental models, Networking and Internetworking, Interprocess Communication.

Distributed objects and Remote Invocation-Introduction, Communication between distributed objects, RPC, Events and notifications, Case study-Java RMI.

UNIT II

Operating System Support- Introduction, OS layer, Protection, Processes and Threads, Communication and Invocation, Operating system architecture, Distributed File Systems-Introduction, File Service architecture, case study- SUN network file systems.

Name Services-Introduction, Name Services and the Domain Name System, Case study of the Global Name Service, Case study of the X.500 Directory Service.

UNIT III

Peer to Peer Systems–Introduction, Napster and its legacy, Peer to Peer middleware, Routing overlays, Overlay case studies-Pastry, Tapestry, Application case studies-Squirrel, OceanStore, Time and Global States-Introduction, Clocks, events and Process states, Synchronizing physical clocks, logical time and logical clocks, global states, distributed debugging.

Coordination and Agreement-Introduction, Distributed mutual exclusion, Elections, Multicast communication, consensus and related problems.

UNIT IV

Transactions and Concurrency control-Introduction, Transactions, Nested Transactions, Locks, Optimistic concurrency control, Timestamp ordering, Comparison of methods for concurrency control. Distributed Transactions-Introduction, Flat and Nested Distributed Transactions, Atomic commit protocols, Concurrency control in distributed transactions, Distributed deadlocks,
Transaction recovery. Replication-Introduction, System model and group communication, Fault tolerant services, Transactions with replicated data.

UNIT V

Security-Introduction, Overview of Security techniques, Cryptographic algorithms, Digital signatures, Case studies-Kerberos, TLS, 802.11 Wi-Fi.
Distributed shared memory, Design and Implementation issues, Sequential consistency and Ivy case study, Release consistency and Munin case study, Other consistency models, CORBA case study-Introduction, CORBA RMI, CORBA Services.

TEXT BOOKS:

1. Distributed Systems Concepts and Design, G Coulouris, J Dollimore and T Kindberg, Fourth Edition, Pearson Education.
2. Distributed Systems, S.Ghosh, Chapman& Hall/CRC, Taylor & Francis Group, 2010.

REFERENCE BOOKS:

1. Distributed Computing, S.Mahajan and S.Shah, Oxford University Press.
2. Distributed Operating Systems Concepts and Design, Pradeep K.Sinha, PHI.
3. Advanced Concepts in Operating Systems, M Singhal, N G Shivarathri, TMH.

J.B. INSTITUTE F ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - II Semester	**4**	**0**	**4**

SOFTWARE PROCESS AND PROJECT MANAGEMENT
(Core Elective-II)

Course Objectives:

- At the end of the course, the student shall be able to: To describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project.
- To compare and differentiate organization structures and project structures.
- To implement a project to manage project schedule, expenses and resources with the application of suitable project management tools.

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 discriminants, 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.
2. Software Project Management, Bob Hughes & Mike Cotterell, fourth edition, Tata Mc Graw Hill.

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 Management, Richard H. Thayer & Edward Yourdon, second edition, Wiley India, 2004.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - II Semester	**4**	**0**	**4**

SOFT COMPUTING
(Core Elective-II)

Course Objectives

- This course explains AI Problems and Search techniques.
- To understand Supervised and Unsupervised Learning Networks Introduces Classical Sets and Fuzzy Sets.
- To understand Genetic Algorithms.

Course Outcomes

- Comprehend the differences between Classical Sets and Fuzzy Sets Get the skill for application of search techniques to solve AI Problems Able to apply Genetic Algorithms for solving real time problems

UNIT I

AI Problems and Search: AI problems, Techniques, Problem Spaces and Search, Heuristic Search Techniques- Generate and Test, Hill Climbing, Best First Search Problem reduction, Constraint Satisfaction and Means End Analysis. Approaches to Knowledge Representation- Using Predicate Logic and Rules.

UNIT II

Artificial Neural Networks: Introduction, Basic models of ANN, important terminologies, Supervised Learning Networks, Perceptron Networks, Adaptive Linear Neuron, Back propagation Network. Associative Memory Networks. Training Algorithms for pattern association, BAM and Hopfield Networks.

UNIT III

Unsupervised Learning Network- Introduction, Fixed Weight Competitive Nets, Maxnet, Hamming Network, Kohonen Self-Organizing Feature Maps, Learning Vector Quantization, Counter Propagation Networks, Adaptive Resonance Theory Networks. Special Networks-Introduction to various networks.

UNIT IV

Introduction to Classical Sets (crisp Sets)and Fuzzy Sets- operations and Fuzzy sets. Classical Relations - and Fuzzy Relations- Cardinality, Operations, Properties and composition. Tolerance and equivalence relations. Membership functions- Features, Fuzzification, membership value assignments, Defuzzification.

UNIT V

Fuzzy Arithmetic and Fuzzy Measures, Fuzzy Rule Base and Approximate Reasoning Fuzzy Decision making Fuzzy Logic Control Systems. Genetic Algorithm- Introduction and basic operators and terminology, Applications: Optimization of TSP, Internet Search Technique

TEXT BOOKS:

1 Principles of Soft Computing- S N Sivanandam, S N Deepa, Wiley India, 2007.
2 Soft Computing and Intelligent System Design -Fakhreddine O Karray, Clarence D Silva, Pearson Edition, 2004.

REFERENCE BOOKS :

1. Artificial Intelligence and SoftComputing- Behavioural and Cognitive Modelling of the Human Brain- Amit Konar, CRC press, Taylor and Francis Group.
2. Artificial Intelligence – Elaine Rich and Kevin Knight, TMH, 1991, rp2008.
3. Artificial Intelligence – Patric Henry Winston – Third Edition, Pearson Education.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - II Semester	**0**	**2**	**2**

WEB TECHNOLOGIES LAB

Objectives:

- The Primary objective of the course is to learn web programming by designing and developing some web based applications.

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 colors-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 WT Lab.:

Write an HTML page including any required Java script 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

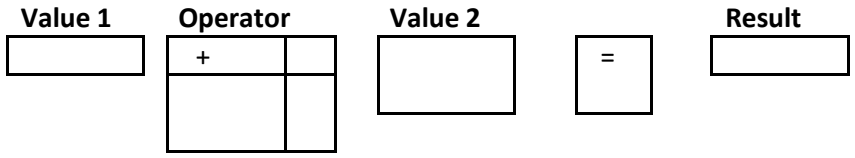
new line character.

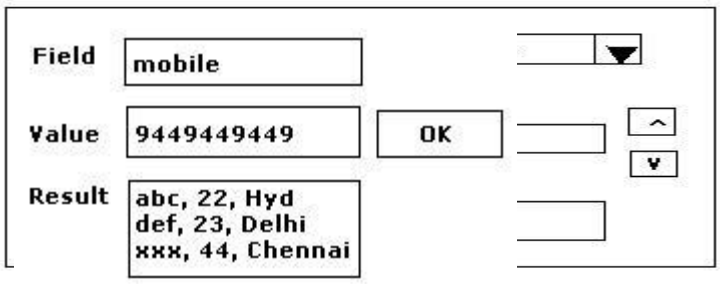
Write a simple calculator servlet that takes two numbers and an operator (+, -, /, * and %) from an HTML page and returns the result page with the operation performed on the operands. It should check in a database if the same expression is already computed and if so, just return the value from database. 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:





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).

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
II Year - II Semester	**0**	**2**	**2**

LINUX PROGRAMMMING LAB

Note: Use Bash for Shell scripts.

1. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
2. Write a shell script that deletes all lines containing a specified word in one or more files Supplied as arguments to it.
3. Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
4. Write a shell script that receives any number of file names as arguments checks if every Argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.
5. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files. Write a shell script to list all of the directory files in a directory.
6. Write a shell script to find factorial of a given integer.
7. Write an awk script to count the number of lines in a file that do not contain vowels.
8. Write an awk script to find the number of characters, words and lines in a file.
9. Write a C program that makes a copy of a file using standard I/O and system calls.
10. Implement in C the following Unix commands using System calls
    a). cat b) mv
12. Write a C program to list files in a directory.
13. Write a C program to emulate the Unix ls –l command.
14. Write a C program to list for every file in a directory, its inode number and file name.
15. Write a C program that redirects standard output to a file.Ex: ls > f1.
16. Write a C program to create a child process and allow the parent to display “parent” and the child to display “child” on the screen.
17. Write a C program to create a Zombie process.
18. Write a C program that illustrates how an orphan is created.
19. Write a C program that illustrates how to execute two commands concurrently with a command pipe. Ex:- ls –l | sort
20. Write C programs that illustrate communication between two unrelated processes using named pipe. Write a C program in which a parent writes a message to a pipe and the child reads the message.
21. Write a C program (sender.c) to create a message queue with read and write permissions to Write 3 messages to it with different priority numbers.
22. Write a C program (receiver.c) that receives the messages (from the above message queue as specified in (22)) and displays them
23. Write a C programs to transfer a large amount of data between Processes, using
    a) a pipe b)a FIFO c)aeue
24. Write a C program to allow cooperating processes to lock a resource for exclusive use, Using a) semaphores b) flock or lockf system calls.
25. Write a C program that illustrates suspending and resuming processes using signals.
26. Write a C program that implements a producer-consumer system with two processes. (using Semaphores).
27. Write client and server programs(using c) for interaction between server and client processes using Unix Domain sockets.
28. Write client and server programs(using c) for interaction between server and client processes using Unix Domain sockets.

29. Write client and server programs(using c) for interaction between server and client processes using Internet Domain sockets.
30. Write C programs that illustrate two processes communicating using shared memory.

TEXT BOOKS:

1. Advanced Unix Programming, N.B.Venkateswarulu, BS Publications.
2. Unix and Shell programming, B.A.Forouzan and R.F.Gilberg, Cengage Learning.
3. Unix and Shell Programming, M.G. Venkatesh Murthy, Pearson Education, 2005.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
III Year - I Semester	**4**	**0**	**4**

ANDROID APPLICATION DEVELOPMENT

Prerequisites:

- A Course on JAVA A Course on DBMS

Course 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

Course Outcomes:

- Student understands the working of Android OS Practically.
- Student will be able to develop, deploy and maintain the Android Applications.

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 NITs 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.

REFERENCE BOOKS:

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

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
III Year - I Semester	**4**	**0**	**4**

SOFTWARE TESTING METHODOLOGIES

Prerequisites:

- A course on "Software Engineering"

Course Objectives:

- To provide knowledge of the concepts in software testing such as testing process, criteria, strategies, and methodologies.
- To develop skills in software test automation and management using latest tools.

Course Outcomes:

- Design and develop the best test strategies in accordance to the development model.

UNIT I

Introduction: - Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs.

Flow graphs and Path testing:- Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT II

Transaction Flow Testing:-transaction flows, transaction flow testing techniques. Dataflow testing: - Basics of dataflow testing, strategies in dataflow testing, application of dataflow testing.

UNIT III

Domain Testing:-domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domains and testability.

Paths, Path products and Regular expressions: - path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

UNIT IV

Logic Based Testing:- overview, decision tables, path expressions, kv charts, specifications. State, State Graphs and Transition testing:- state graphs, good & bad state graphs, state testing, Testability tips.

UNIT V

Graph Matrices and Application:-Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. (Student should be given an exposure to a tool like JMeter or Win-runner).

TEXT BOOKS:

1. Software Testing techniques - Baris Beizer, Dreamtech, second edition.
2. Software Testing Tools – Dr.K.V.K.K.Prasad, Dreamtech.

REFERENCE BOOKS:

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing,3rd edition, P.C.Jorgensen, Aurbach Publications(Dist. by SPD).
3. Software Testing in the Real World – Edward Kit, Pearson.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
III Year - I Semester	**4**	**0**	**4**

WEB SERVICES & SERVICE ORIENTED ARCHITECTURE (SOA)
(CORE Elective - III)

Pre-requisites:

- The course assumes a reasonable comfort and background about Information Technology and Management Information Systems.

Course Objectives:

- To gain understanding of the basic principles of service orientation
  to learn service oriented analysis techniques
- To learn technology underlying the service design
- To learn the concepts such as SOAP, Registering and Discovering Services.

Course Outcomes: At the end of this course, students are expected to gain the following learning:

- Get the foundations and concepts of service based computing
- Advocate the importance and means of technology alignment with business
  Understanding the basic operational model of web services,

- Gain the knowledge of key technologies in the service oriented computing arena
  Apply and practice the learning through a real or illustrative project/case study.

UNIT – I

Evolution and Emergence of Web Services – Evolution of distributed computing. Core distributed computing technologies – client/server, CORBA, JAVA RMI, Micro Soft DCOM, MOM, Challenges in Distributed Computing, role of J2EE and XML in distributed computing, emergence of Web Services and Service Oriented Architecture (SOA). Introduction to Web Services – The definition of web services, basic operational model of web services, tools and technologies enabling web services, benefits and challenges of using web services.

UNIT - II

Web Service Architecture – Web services Architecture and its characteristics, core building blocks of web services, standards and technologies available for implementing web services, web services communication, basic steps of implementing web services. Describing Web Services – WSDL introduction, non functional service description, WSDL1.1 Vs WSDL 2.0, WSDL document, WSDL elements, WSDL binding, WSDL tools, WSDL port type, limitations of WSDL.

UNIT - III

Brief Over View of XML – XML Document structure, XML namespaces, Defining structure in XML documents, Reuse of XML schemes, Document navigation and transformation. SOAP : Simple Object Access Protocol, Inter-application communication and wire protocols, SOAP as a messaging protocol, Structure of a SOAP message, SOAP envelope, Encoding, Service Oriented Architectures, SOA revisited, Service roles in a SOA, Reliable messaging, The enterprise Service Bus, SOA Development Lifecycle, SOAP HTTP binding, SOAP communication model, Error handling in SOAP.

UNIT – IV

Registering and Discovering Services : The role of service registries, Service discovery, Universal Description, Discovery, and Integration, UDDI Architecture, UDDI Data Model, Interfaces, UDDI Implementation, UDDI with WSDL, UDDI specification, Service Addressing and Notification, Referencing and addressing Web Services, Web Services Notification.

UNIT - V

SOA and web services security considerations, Network-level security mechanisms, Application-level security topologies, XML security standards, Semantics and Web Services, The semantic interoperability problem, The role of metadata, Service metadata, Overview of .NET and J2EE, SOA and Web Service Management, Managing Distributed System, Enterprise management Framework, Standard distributed management frameworks, Web service management, Richer schema languages, WS-Metadata Exchange.

TEXT BOOKS:

1. Web Services & SOA Principles and Technology, Second Edition, Michael P. Papazoglou.
2. Developing Java Web Services, R. Nagappan, R. Skoczylas, R.P. Sriganesh, Wiley India.
3. Developing Enterprise Web Services, S. Chatterjee, J. Webber, Pearson Education.

REFERENCE BOOKS:

1. XML, Web Services, and the Data Revolution, F. P. Coyle, Pearson Education.
2. Building web Services with Java, 2nd Edition, S. Graham and others, Pearson Education.
3. Java Web Services, D.A. Chappell & T. Jewell, O'Reilly, SPD.
4. McGovern, et al., "Java web Services Architecture", Morgan Kaufmann Publishers, 2005.
5. J2EE Wer Services, Richard Monson-Haefel, Pearson Education

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
III Year-I Semester	**4**	**0**	**4**

DISTRIBUTED DATABASES
(CORE ELECTIVE - III)

Prerequisites

- A course on "Database Management Systems"

Course Objectives

- To acquire knowledge on parallel and distributed databases and its applications.
  To study the usage and applications of Object Oriented databases.
- To learn the modelling and design of databases
- To acquire knowledge on parallel and distributed databases and its applications.
- Equip students with principles and knowledge of parallel and object oriented databases.
- Topics include distributed DBMS architecture and design; query processing and optimization; distributed transaction management and reliability; parallel and object database management systems.

Course Outcomes

- Understand theoretical and practical aspects of distributed database systems.
- Study and identify various issues related to the development of distributed database system.
  Understand the design aspects of object oriented database system and related development.

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 transformation 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 - 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.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	L	P	C
III Year - I Semester	4	0	4

INFORMATION RETRIEVAL SYSTEMS
(Core Elective-III)

Prerequisites:

- Data Structures

Course Objectives:

- To learn the important concepts and algorithms
- To understand the data/file structures those are necessary to design, and implement Information Retrieval (IR) systems.

Course Outcomes:

- Can be able to use different information retrieval techniques in various application areas
  Can apply IR principles to locate relevant information large collections of data
- Able to analyse performance of retrieval systems when dealing with unmanaged data sources
  Can be able to implement retrieval systems for web search tasks.

UNIT I
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.

REFERENCE BOOKS:
1. Information Storage and Retrieval Systems: Theory and Implementation, Kowalski, Gerald, Mark T Maybury, Springer.
2. Modern Information Retrival , Ricardo Baeza-Yates, Pearson Education, 2007.
3. Information Retrieval: Algorithms and Heuristics, David A Grossman and Ophir Frieder, 2nd Edition, Springer, 2004.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
III Year - I Semester	**4**	**0**	**4**

ETHICAL HACKING
(Core Elective-IV)

Prerequisites

- A course on "Operating Systems" A course on "Computer Networks"
- A course on "Network Security and Cryptography"

Course Objectives

- The aim of the course is to introduce the methodologies and framework of ethical hacking for enhancing the security.
- The course includes-Impacts of Hacking; Types of Hackers; Information Security Models; Information Security Program; Business Perspective; Planning a Controlled Attack; Framework of Steps (Reconnaissance, Enumeration, Vulnerability Analysis, Exploitation, Deliverable and Integration)

Course Outcomes

- Gain the knowledge of the use and availability of tools to support an ethical hack Gain the knowledge of interpreting the results of a controlled attack

- Understand the role of politics, inherent and imposed limitations and metrics for planning of a test Comprehend the dangers associated with penetration testing

UNIT I

Introduction to Ethical Hacking, Ethics, and Legality:
Ethical Hacking Terminology, Different Types of Hacking Technologies, Different Phases Involved in Ethical Hacking and Stages of Ethical Hacking: Passive and Active Reconnaissance, Scanning, Gaining Access, Maintaining Access, Covering Tracks, Hacktivism, Types of Hacker Classes, Skills Required to Become an Ethical Hacker, Vulnerability Research, Ways to Conduct Ethical Hacking, Creating a Security Evaluation Plan, Types of Ethical Hacks, Testing Types, Ethical Hacking Report.
Footprinting and Social Engineering:
Footprinting , Information Gathering Methodology, Competitive Intelligence, DNS Enumeration Who is and ARIN Lookups, Types of DNS Records, Traceroute, E-Mail Tracking, Web Spiders, Social Engineering, Common Types Of Attacks, Insider Attacks, Identity Theft, Phishing Attacks, Online Scams, URL Obfuscation, Social-Engineering Countermeasures.

UNIT II

Scanning and Enumeration
Scanning, types of Scanning , CEH Scanning Methodology ,Ping Sweep Techniques, Nmap Command Switches, SYN, Stealth, XMAS, NULL, IDLE, and FIN Scans, TCP Communication Flag Types, War-Dialing Techniques, Banner Grabbing and OS Fingerprinting Techniques, Proxy Servers, Anonymizers , HTTP Tunneling Techniques, IP Spoofing Techniques, Enumeration, Null Sessions, SNMP Enumeration, Windows 2000 DNS Zone Transfer, Steps Involved in Performing Enumeration.

System Hacking

Understanding Password-Cracking Techniques, Understanding the LAN Manager Hash Cracking Windows 2000 Passwords, Redirecting the SMB Logon to the Attacker SMB Redirection, SMB Relay MITM Attacks and Countermeasures NetBIOS DoS Attacks, Password-Cracking Countermeasures, Understanding Different Types of Passwords Passive Online Attacks, Active Online Attacks, Offline Attacks Nonelectronic Attacks, Understanding Keyloggers and Other Spyware Technologies Understand Escalating Privileges, Executing Applications, Buffer Overflows, Understanding Rootkits Planting Rootkits on Windows 2000 and XP Machines, Rootkit Embedded TCP/IP Stack Rootkit Countermeasures, Understanding How to Hide Files, NTFS File Streaming NTFS Stream Countermeasures, Understanding Steganography Technologies, Understanding How to Cover Your Tracks and Erase Evidence, Disabling Auditing, Clearing the Event Log.

UNIT III

Trojans, Backdoors, Viruses, and Worms

Trojans and Backdoors, Overt and Covert Channels, Types of Trojans, Reverse-Connecting Trojans, Netcat Trojan, Indications of a Trojan Attack, Wrapping, Trojan Construction Kit and Trojan Makers, Countermeasure Techniques in preventing Trojans, Trojan-Evading Techniques, System File Verification Sub-objective to Trojan Countermeasures Viruses and Worms, Difference between a Virus and a Worm ,Types of Viruses, Understand Antivirus Evasion Techniques, Understand Virus Detection Methods.

Sniffers

Protocols Susceptible to Sniffing, Active and Passive Sniffing, ARP Poisoning, Ethereal Capture and Display Filters, MAC Flooding, DNS Spoofing Techniques, Sniffing Countermeasures.

Denial of Service and Session Hijacking Denial of Service, Types of DoS Attacks, DDoS Attacks, BOTs/BOTNETs, “Smurf” Attack, “SYN” Flooding, DoS/DDoS Countermeasures, Session Hijacking, Spoofing vs. Hijacking, Types of Session Hijacking, Sequence Prediction, Steps in Performing Session Hijacking, Prevention of Session Hijacking.

UNIT IV

Hacking Web Servers, Web Application Vulnerabilities, and Web-Based Password Cracking Techniques

Hacking Web Servers, Types of Web Server Vulnerabilities, Attacks against Web Servers, IIS Unicode Exploits, Patch Management Techniques, Web Server Hardening Methods Web Application Vulnerabilities, Objectives of Web Application Hacking, Anatomy of an Attack, Web Application Threats, Google Hacking, Web Application Countermeasures Web-Based Password Cracking Techniques, Authentication Types, Password Cracker, Password Attacks: Classification, Password-Cracking Countermeasures.

SQL Injection and Buffer Overflows

SQL Injection, Steps to Conduct SQL Injection, SQL Server Vulnerabilities, SQL Injection Countermeasures Buffer Overflows, Types of Buffer Overflows and Methods of Detection, Stack-Based Buffer Overflows, Buffer Overflow Mutation Techniques.

UNIT V

Linux Hacking

Linux Basics, Compile a Linux Kernel, GCC Compilation Commands, Install Linux Kernel Modules, Linux Hardening Methods.

Penetration Testing Methodologies

Security Assessments, Penetration Testing Methodologies, Penetration Testing Steps, Pen-Test Legal Framework, Automated Penetration Testing Tools, Pen-Test Deliverables.

TEXT BOOK:

1. CEH official Certified Ethical Hacking Review Guide, Wiley India Edition.

REFERENCE BOOKS:

1. Hacking Exponsed Web 2.0, by Rich Annings, Himanshu Dwivedi, Zane Lackey, Tata Mc Graw hill Edition.
2. Ethical Hacking & Network Defense, Michael T. Simpson edition.
3. Hacking Exposed Windows, Joel Scambray, cissp, Stuart Mcclure, Cissp, Third Edition, Tata Mc Graw hill edition.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	L	P	C
III Year - I Semester	4	0	4

MULTIMEDIA AND RICH INTERNET APPLICATIONS
(Core Elective-IV)

Course Objectives:

This course aims to further develop students' competency in producing dynamic and creative graphic solutions for multimedia productions. It provides students with the basic concepts and techniques of interactive authoring. It also introduces students with the advanced scripting skills necessary for implementing highly interactive, rich internet applications using multimedia technologies and authoring tools. Students will develop aesthetic value and competencies in multimedia authoring. Artistic visual style and layout design are stressed, as well as the editing and integration of graphic images, animation, video and audio files. The course allows students to master industry-wide software and technologies to create highly interactive, rich internet applications.

UNIT I

Fundamental concepts in Text and Image: Multimedia and hypermedia, World Wide Web, overview of multimedia software tools. Graphics and image data representation graphics/image data types, file formats, Color in image and video: color science, color models in images, color models in video.

UNIT II

Fundamental concepts in video and digital audio: Types of video signals, analog video, digital video, digitization of sound, MIDI, quantization and transmission of audio.
Multimedia Data Compression: Lossless compression algorithms, Lossy compression algorithms, Image compression standards.

UNIT III

Basic Video compression techniques, Case study: MPEG Video Coding I, Basic Audio compression techniques, Case study: MPEG Audio compression.

Web 2.0

What is web 2.0, Search, Content Networks, User Generated Content, Blogging, Social Networking, Social Media, Tagging, Social Marking, Rich Internet Applications, Web Services, Mashups, Location Based Services, XML, RSS, Atom, JSON, and VoIP, Web 2.0 Monetization and Business Models, Future of the Web.

UNIT IV

Rich Internet Applications(RIAs) with Adobe Flash : Adobe Flash- Introduction, Flash Movie Development, Learning Flash with Hands-on Examples, Publish your flash movie, Creating special effects with Flash, Creating a website splash screen, action script, web sources.

Rich Internet Applications (RIAs) with Flex 3 - Introduction, Developing with Flex 3, Working with Components, Advanced Component Development, Visual Effects and Multimedia,

UNIT V

Ajax- Enabled Rich Internet Application : Introduction, Traditional Web Applications vs Ajax Applications, Rich Internet Application with Ajax, History of Ajax, Raw Ajax example using xmlhttprequest object, Using XML, Creating a full scale Ajax Enabled application, Dojo ToolKit.

TEXT BOOKS:

1. Fundamentals of Multimedia by Ze-Nian Li and Mark S. Drew PHI Learning, 2004
2. Professional Adobe Flex 3, Joseph Balderson, Peter Ent, et al, Wrox Publications, Wiley India, 2009.

REFERENCE BOOKS:

1. AJAX, Rich Internet Applications, and Web Development for Programmers, Paul J Deitel and Harvey M Deitel, Deitel Developer Series, Pearson Education.
2. Multimedia Communications: Applications, Networks, Protocols and Standards, Fred Halsall, Pearson Education, 2001, rp 2005.
3. Multimedia Making it work, Tay Vaughan, 7th edition, TMH, 2008.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
III Year - I Semester	**4**	**0**	**4**

SEMANTIC WEB AND SOCIAL NETWORKS
(Core Elective-IV)

Course Objectives:

- To learn Web Intelligence
- To learn Knowledge Representation for the Semantic Web To learn Ontology Engineering
- To learn Semantic Web Applications, Services and Technology To learn Social Network Analysis and semantic web

UNIT I

Thinking and Intelligent Web Applications, The Information Age, The World Wide Web, Limitations of Todays Web, The Next Generation Web Machine Intelligence, Artificial Intelligence.
Ontology, Inference engines, Software Agents, Berners-Lee www, Semantic Road Map, Logic on the semantic Web.

UNIT II

Ontologies and their role in the semantic web, Ontologies Languages for the Semantic Web –Resource Description Framework(RDF) / RDF Schema, Ontology Web Language(OWL),UML,XML/XML Schema. **Ontology Engineering**, Constructing Ontology, Ontology Development Tools, Ontology Methods, Ontology Sharing and Merging, Ontology Libraries and Ontology Mapping.

UNIT III

Logic, Rule and Inference Engines. Semantic Web applications and services, Semantic Search, e-learning, Semantic Bioinformatics, Knowledge Base.

UNIT IV

XML Based Web Services, Creating an OWL-S Ontology for Web Services, Semantic Search Technology, Web Search Agents and Semantic Methods.
What is social Networks analysis, development of the social networks analysis, Electronic Sources for Network Analysis – Electronic Discussion networks.

UNIT V

Blogs and Online CommUNITies, Web Based Networks. Building Semantic Web Applications with social network features.

TEXT BOOKS:

1. Thinking on the Web - Berners Lee, Godel and Turing, Wiley inter science, 2008.
2. Social Networks and the Semantic Web , Peter Mika, Springer, 2007.

REFERENCE BOOKS:

1. Semantic Web Technologies, Trends and Research in Ontology Based Systems, J.Davies, Rudi Studer, Paul Warren, John Wiley & Sons.
2. Semantic Web and Semantic Web Services -Liyang Lu Chapman and Hall/CRC Publishers, (Taylor& Francis Group)
3. Information sharing on the semantic Web - Heiner Stuckenschmidt; Frank Van Harmelen, Springer Publications.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
III Year - I Semester	**0**	**2**	**2**

ANDROID APPLICATION DEVELOPMENT LAB

Course Objectives:

- To learn how to develop Applications in android environment. To learn how to develop user interface applications.
- To learn how to develop URL related applications.

Course Outcomes:

- Student understands the working of Android OS Practically.
- Student will be able to develop, deploy and maintain the Android Applications.

The student is expected to be able to do the following problems, though not limited. Create an

1. Android application that shows Hello + name of the user and run it on an emulator.

(b) Create an application that takes the name from a text box and shows hello message along with the name entered in text box, when the user clicks the OK button.

2. Create a screen that has input boxes for User Name, Password, Address, Gender (radio buttons for male and female), Age (numeric), Date of Birth (Date Picket), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button. Use (a) Linear Layout (b) Relative Layout and (c) Grid Layout or Table Layout.

3 Develop an application that shows names as a list and on selecting a name it should show the details of the candidate on the next screen with a “Back” button. If the screen is rotated to landscape mode (width greater than height), then the screen should show list on left fragment and details on right fragment instead of second screen with back button. Use Fragment transactions and Rotation event listener.

4 Develop an application that uses a menu with 3 options for dialing a number, opening a website and to send an SMS. On selecting an option, the appropriate action should be invoked using intents.

5 Develop an application that inserts some notifications into Notification area and whenever a notification is inserted, it should show a toast with details of the notification.

6 Create an application that uses a text file to store user names and passwords (tab separated fields and one record per line). When the user submits a login name and password through a screen, the details should be verified with the text file data and if they match, show a dialog saying that login is successful. Otherwise, show the dialog with Login Failed message.

7 Create a user registration application that stores the user details in a database table.

8 Create a database and a user table where the details of login names and passwords are stored. Insert some names and passwords initially. Now the login details entered by the user should be verified with the database and an appropriate dialog should be shown to the user.

9 Create an admin application for the user table, which shows all records as a list and the admin can select any record for edit or modify. The results should be reflected in the table.

10 Develop an application that shows all contacts of the phone along with details like name, phone number, mobile number etc.

11 Create an application that saves user information like name, age, gender etc. in shared preference and retrieves them when the program restarts.

12 Create an alarm that rings every Sunday at 8:00 AM. Modify it to use a time picker to set alarm time.

Create an application that shows the given URL (from a text field) in a browser.

13 Develop an application that shows the current location's latitude and longitude continuously as the device is moving (tracking).

14 Create an application that shows the current location on Google maps.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA	**L**	**P**	**C**
III Year - I Semester	**0**	**2**	**2**

SOFTWARE ENGINEERING METHODOLOGIES AND UML LAB

PART-A

USING ARGO UML - CASE TOOL

LIST OF EXPERIMENTS

1. Prepare the following documents for two or three of the experiments listed below and develop the software engineering methodology.

2. Program Analysis and Project Planning. Thorough study of the problem - Identify project scope, Objectives, Infrastructure.
3. Software requirement Analysis. Describe the individual Phases / Modules of the project, Identify deliverables.

4. Data Modeling. Use work products - Data dictionary, Use diagrams and activity diagrams, build and test lass diagrams, Sequence diagrams and add interface to class diagrams.
5. Software Development and Debugging

6. Software Testing. Prepare test plan, perform validation testing, Coverage analysis, memory leaks, develop test case hierarchy, Site check and Site monitor.
7. Using WinRunner Design a test plan for Library Management System.

SUGGESTED LIST OF APPLICATIONS

1. Student Marks Analyzing System

2. Quiz System

3. Online Ticket Reservation System

4. Payroll System

5. Course Registration System

6. Expert Systems

7. ATM Systems

8. Stock Maintenance

9. Real-Time Scheduler

10. Remote Procedure Call Implementation

11. Library Management System

PART-B

UML LAB

Course Objectives:

The student should take up the case study of Unified Library application which is mentioned in the theory, and Model it in different views i.e Use case view, logical view, component view,

Deployment view, Database design, forward and Reverse Engineering, and Generation of documentation of the project.
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 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.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

ALGORITHMICS
(Open Elective)

UNIT –I

Relevant Mathematics: Existential and Universal Quantifiers, Logarithms and Exponentials, The Time(and Space) Complexity of an Algorithm, Asymptotic Notations and Their Properties, Adding Mode easy Approximations, Recurrence Relations,

Abstractions: Different representations of Algorithms, Abstract Data Types (ADTs),

UNIT – II

Iterative Algorithms and Loop Invariants: Iterative algorithms: Measures of Progress and Loop Invariants, Examples Using More –Of- the – Input Loop Invariants,

UNIT – III

Narrowing the Search Space: Binary Search, Iterative Searching Algorithm Euclid's GCD Algorithm, The Loop Invariant for Lower Bound,

UNIT – IV

Recursion: Abstractions, Techniques and theory, Some Sample Algorithms of Recursive Algorithms, Recursion on trees, Recursive Images, Parsing with Context-free Grammars.

UNIT – V

Optimization Problems: Definition, Graph Search Algorithms, Network Flow and Linear programming, Greedy Algorithms, Recursive backtracking, Dynamic Programming Algorithms, Examples of Dynamic Programs, Reduction and NP-Completeness, Randomized Algorithms.

TEXT BOOKS:

1. How to think about Algorithms by Jeff Edmonds Cambridge 2003 and 2008.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

BIG DATA ANALYTICS
(Open Elective)

UNIT I

Big Data Analytics: What is big data, History of Data Management; Structuring Big Data; Elements of Big Data ; Big Data Analytics; Distributed and Parallel Computing for Big Data;

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; Data Scientist; Terminologies used in Big Data Environments; Basically Available Soft State Eventual Consistency (BASE); Open source Analytics Tools;

UNIT- II

Understanding Analytics and Big Data: Comparing Reporting and Analysis, Types of Analytics; Points to Consider during Analysis; Developing an Analytic Team; Understanding Text Analytics;

Analytical Approach and Tools to Analyze Data: Analytical Approaches; History of Analytical Tools; Introducing Popular Analytical Tools; Comparing Various Analytical Tools.

UNIT III

Understanding MapReduce Fundamentals and HBase : The MapReduce Framework; Techniques to Optimize MapReduce Jobs; Uses of MapReduce; Role of HBase in Big Data Processing; Storing Data in Hadoop : Introduction of HDFS, Architecture, HDFC Files, File system types, commands, org.apache.hadoop.io package, HDF, HDFS High Availability; Introducing HBase, Architecture, Storing Big Data with HBase , Interacting with the Hadoop Ecosystem; HBase in Operations-Programming with HBase; Installation, Combining HBase and HDFS;

UNIT IV

Big Data Technology Landscape and Hadoop : NoSQL, Hadoop; RDBMS versus Hadoop; Distributed Computing Challenges; History of Hadoop; Hadoop Overview; Use Case of Hadoop; Hadoop Distributors;

HDFC (Hadoop Distributed File System), HDFC Daemons, read,write, Replica Processing of Data with Hadoop; Managing Resources and Applications with Hadoop YARN.

UNIT V

Social Media Analytics and Text Mining: Introducing Social Media; Key elements of Social Media; Text mining; Understanding Text Mining Process; Sentiment Analysis, Performing Social Media Analytics and Opinion Mining on Tweets;

Mobile Analytics: Introducing Mobile Analytics; Define Mobile Analytics; Mobile Analytics and Web Analytics; Types of Results from Mobile Analytics; Types of Applications for Mobile Analytics; Introducing Mobile Analytics Tools;

TEXT BOOKS:

1. BIG DATA and ANALYTICS, Seema Acharya, Subhasinin Chellappan, Wiley publications.
2. BIG DATA, Black BookTM , DreamTech Press, 2015 Edition.

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

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

BIOINFORMATICS
(Open Elective)

UNIT-I

Introduction to Bioinformatics and Biological Databases, Sequence allignment, Pairwise Sequence allignment, multiple sequence allignment, database Similarities.

UNIT-II

Molecular phylogenetics: Basics, gene phylogene Vs Systems Phylogene,Tree construction methods and programs, advanced Statistical approaches, profiles and Hidden markow models.

UNIT-III

Gene and promoter prediction: Gene Prediction, promoter and regulatory element pridiction, RNA structure prediction, protine motives and domain prediction
.

UNIT-IV

Structural Bioinformatics: Basics,Protine structure Visualization, comparision,classofication, protein secondary structure prediction, protein tertiary structure prediction.

UNIT-V

Genomics and Proteomics: Genome Mapping, Assembly, comparison, functional genomics, proteomics.

TEXT BOOKS:

1. Essential Bioinformatics: Jin Xiong 2006, Cambridge University Press.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

BIOMETRICS
(Open Elective)

Objectives:

- To learn the biometric technologies
- To learn the computational methods involved in the biometric systems.
- To learn methods for evaluation of the reliability and quality of the biometric systems.

UNIT – I

INTRODUCTION & HANDWRITTEN CHARACTER RECOGNITION

Introduction – history – type of Biometrics – General Architecture of Biometric Systems – Basic Working of biometric Matching – Biometric System Error and performance Measures – Design of Biometric Systems – Applications of Biometrics – Benefits of Biometrics Versus Traditional Authentication Methods

– character Recognition – System Overview – Geature Extraction for character Recognition – Neura; Network for handwritten Charater Recognition – Multilayer Neural Network for Handwritten Character Recognition – Devanagari Numeral Recognition – Isolated Handwritten Devanagari Charater Recognition suing Fourier Descriptor and Hidden markov Model.

UNIT –II.

FACE BIOMETRICS & RETINA AND IRIS BIOMETRICS

Introduction –Background of Face Recognition – Design of Face Recognition System – Neural Network for Face Recognition – Face Detection in Video Sequences – Challenges in Face Biometrices – Face Recognition Methods – Advantages and Disadvantages – Performance of Biometrics – Design of Retina Biometrics – Iris Segmentation Method – Determination of Iris Region – Experimental Results of Iris Localization – Applications of Iris Biometrics – Advantages and Disadvantages. VEIN AND

FINGERPRINT BIOMETRICS & BIOMETRIC HAND GESTURE RECOGNITION FOR INDIAN SIGN LANGUAGE. Biometrics Using Vein Pattern of Palm – Fingerprint Biometrics – Fingerprint Recognition System – Minutiae Extraction – Fingerprint Indexing – Experimental Results – Advantages and Disadvantages – Basics of Hand Geometry – Sign Language – Indian Sign Language – SIFT Algorithms-Practical Approach Advantages and Disadvantages.

UNIT –III.

PRIVACY ENHANCEMENT USING BIOMETRICS & BIOMETRIC CRYPTOGRAPHYAND MULTIMODAL BIOMETRICS

Introduction – Privacy Concerns Associated with Biometric Developments – Identity and Privacy – Privacy Concerns – Biometrics with Privacy Enhancement – Comparison of Various Biometrics in Terms of Privacy – Soft Biometrics - Introduction to Biometric Cryptography – General Purpose Cryptosystem – Modern Cryptography and Attacks – Symmetric Key Ciphers – Cryptographic Algorithms – Introduction to Multimodal Biometrics – Basic Architecture of Multimodal Biometrics – Multimodal Biometrics Using Face and Ear – Characteristics and Advantages of Multimodal Biometrics Characters – AADHAAR : An Application of Multimodal Biometrics.

UNIT –IV

WATERMARKING TECHNIQUES & BIOMETRICS : SCOPE AND FUTURE

Introduction – Data Hiding Methods – Basic Framework of Watermarking – Classification of Watermarking – Applications of Watermarking – Attacks on Watermarks – Performance Evaluation – Characteristics of Watermarks – General Watermarking Process – Image Watermarking Techniques – Watermarking Algorithm – Experimental Results – Effect of Attacks on Watermarking Techniques – Scope and Future Market of Biometrics – Biometric Technologies – Applications of Biometrics -Biometrics – and Information Technology Infrastructure – Role of Biometrics in Enterprise Security – Role of Biometrics in Border Security – Smart Card Technology and Biometric – Radio Frequency Identification Biometrics – DNA Biometrics – Comparative Study of Various Biometrics Techniques.

UNIT –V

IMAGE ENHANCEMENT TECHNIQUES & BIOMETRICS STANDS

Introduction – current Research in image Enhancement Techniques – Image Enhancement – Frequency Domain Filters – Databases and Implementation – Standard Development Organizations – Application Programming Interface – Information Security and Biometric Standards – Biometric Template Interoperability.

TEXT BOOKS:

1. BIOMETRICS: CONCEPTS AND APPLICATIONS by G R SINHA and SANDEEP B. PATIL, Wiely, 2013.
2. Biometrics for Network Security – Paul Reid, Pearson Education.

REFERENCE BOOKS:

1. Biometrics – Identity verification in a networked world – Samir Nanavathi, Micheal Thieme, Raj Nanavathi, Wiley – dream Tech.
2. Biometrics – The Ultimate Reference – John D. Woodward, Jr.Wiley Dreamtech.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

COMPUTER FORENSICS
(Open Elective)

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.

REFERENCE BOOKS:

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

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

CYBER SECURITY
(Open Elective)

Objectives:

- To learn about cyber crimes and how they are planned
- To learn the vulnerabilities of mobile and wireless devices
- To learn about the crimes in mobile and wireless devices

UNIT-I

Introduction to Cybercrime:
Introduction, Cybercrime and Information security, who are cybercriminals, Classifications of Cybercrimes, Cybercrime: The legal Perspectives and Indian Perspective, Cybercrime and the Indian ITA 2000, A Global Perspective on Cybercrimes. **Cyber offenses:** How criminals Plan Them
Introduction, How Criminals plan the Attacks, Social Engineering, Cyber stalking, Cyber cafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector, Cloud Computing.

UNIT-II

Cybercrime: Mobile and Wireless Devices
Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for Organizations, Organizational Measures for Handling Mobile, Organizational Security Policies an Measures in Mobile Computing Era, Laptops.

UNIT III

Cybercrimes and Cyber security: the Legal Perspectives Introduction
Cyber Crime and Legal Landscape around the world, Why Do We Need Cyber laws: The Indian Context, The Indian IT Act, Challenges to Indian Law and Cybercrime Scenario In India, Digital signatures and the Indian IT Act, Amendments to the Indian IT Act, Cybercrime and Punishment ,Cyber law, Technology and Students: Indian Scenario.

UNIT IV

Understanding Computer Forensics : Introduction, Historical background of Cyber forensics, Digital Forensics Science, The Need for Computer Forensics, Cyber Forensics and Digital evidence, Forensics Analysis of Email, Digital Forensics Lifecycle, Chain of Custody concept, Network Forensics, Approaching a computer, Forensics Investigation, Challenges in Computer Forensics, Special Tools and Techniques,Forensics Auditing

UNIT V

Cyber Security: Organizational Implications
Introduction, Cost of Cybercrimes and IPR issues, Web threats for Organizations, Security and Privacy Implications, Social media marketing: Security Risks and Perils for Organizations, Social Computing and the associated challenges for Organizations.

TEXT BOOKS:

1. **Cyber Security:** *Understanding Cyber Crimes, Computer Forensics and Legal Perspectives*, Nina Godbole and Sunil Belapure, Wiley INDIA.
2. Introduction to Cyber Security , Chwan-Hwa(john) Wu,J.David Irwin.CRC Press T&F Group
3 Cyber Security Essentials, James Graham, Richard Howard and Ryan Otson, CRC Press.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

DISTRIBUTED SYSTEMS SECURITY
(Open Elective)

Objectives:

- To learn about the distributed systems and security
- To learn about the host-level application level threats and vulnerabilities
- To learn about service-level sikytuibs

UNIT-I

Introduction – Distributed Systems, Distributed Systems Security. Security in Engineering: Secure Development Lifecycle Processes - A Typical Security Engineering Process – Security Engineering Guidelines and Resources. Common Security Issues and Technologies: Security Issues, Common Security Techniques.

UNIT-II

Host-level Threats and Vulnerabilities: Transient code Vulnerabilities - Resident Code Vulnerabilities - Malware: Trojan Horse – Spyware - Worms/Viruses – Eavesdropping – Job Faults. Infrastructure-Level Threats and Vulnerabilities: Network-Level Threats and Vulnerabilities - Grid Computing Threats and Vulnerabilities – Storage Threats and Vulnerabilities – Overview of Infrastructure Threats and Vulnerabilities.

UNIT-III

Application-Level Threats and Vulnerabilities: Application-Layer Vulnerabilities –Injection Vulnerabilities - Cross-Site Scripting (XSS) - Improper Session Management - Improper Error Handling - Improper Use of Cryptography - Insecure Configuration Issues - Denial of Service - Canonical Representation Flaws - Overflow Issues. Service-Level Threats and Vulnerabilities: SOA and Role of Standards - Service-Level Security Requirements - Service-Level Threats and Vulnerabilities - Service-Level Attacks - Services Threat Profile.

UNIT-IV

Host-Level Solutions: Sandboxing – Virtualization - Resource Management - Proof-Carrying Code - Memory Firewall – Antimalware. Infrastructure-Level Solutions: Network-Level Solutions - Grid-Level Solutions - Storage-Level Solutions. Application-Level Solutions: Application-Level Security Solutions.

UNIT-V

Service-Level Solutions: Services Security Policy - SOA Security Standards Stack – Standards in Dept - Deployment Architectures for SOA Security - Managing Service-Level Threats - Compliance in Financial Services - SOX Compliance - SOX Security Solutions – Multilevel Policy-Driven Solution Architecture - Case Study: Grid - The Financial Application – Security Requirements Analysis. Future Directions - Cloud Computing Security – Security Appliances - Usercentric Identity Management - Identity-Based Encryption (IBE) - Virtualization in Host Security.

TEXT BOOKS:

1. Abhijit Belapurakar, Anirban Chakrabarti and et al., “Distributed Systems Security: Issues. Processes and solutions”, Wiley, Ltd., Publication, 2009.
2. Abhijit Belapurkar, Anirban Chakrabarti, Harigopal Ponnapalli, Niranjan Varadarajan, Srinivas Padmanabhuni and Srikanth Sundarrajan,

REFERENCE BOOKS:

1. “Distributed Systems Security: Issues, Processes and Solutions”, Wiley publications, 2009.
2. Rachid Guerraoui and Franck Petit, “Stabilization, Safety, and Security of Distributed Systems”, Springer, 2010.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

E – COMMERCE
(Open Elective)

Objectives:

- Identify the major categories and trends of e-commerce applications.
- Identify the essential processes of and 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 pryment 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.
2.E-Commerce fundamentals and applications Hendry Chan, Raymond Lee, Tharam Dillon, Ellizabeth Chang, John Wiley.

REFERENCES BOOKS:

1. E-Commerce, S.Jaiswal – Galgotia.
2. E-Commerce, Efrain Turbon, Jae Lee, David King, H.Michael Chang.
3. Electronic Commerce – Gary P.Schneider – Thomson.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

EMBEDDED SYSTEMS
(Open Elective)

Objectives:

- To explain various embedded system applications and design requirements.
- To construct embedded system hardware.
- To develop software programs to control embedded system.
- To generate product specification for embedded system.

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 - Testing, Simulation and Debugging Techniques and Tools: Testing on Host Machine, Simulators, Laboratory Tools

TEXT BOOK:

1. Embedded Systems, Raj Kamal, Second Edition TMH.
2. Embedded/Real-Time Systems, Dr.K.V.K.K.Prasad, dreamTech press

REFERENCE BOOKS:

1. The 8051 Microcontroller and Embedded Systems, Muhammad Ali Mazidi, Pearson.
2. The 8051 Microcontroller, Third Edition, Kenneth J.Ayala, Thomson.
3. An Embedded Software Primer, David E. Simon, Pearson Education.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

INTELLECTUAL PROPERTY RIGHTS
(Open Elective)

UNIT-I

Introduction to Intellectual Property Law – The Evolutionary Past - The IPR Tool Kit- Para -Legal Tasks in Intellectual Property Law Ethical obligations in Para Legal Tasks in Intellectual Property Law - Introduction to Cyber Law – Innovations and Inventions Trade related Intellectual Property Right

UNIT-II

Introduction to Trade mark – Trade mark Registration Process – Post registration Procedures – Trade mark maintenance - Transfer of Rights - Inter partes Proceeding – Infringement - Dilution Ownership of Trade mark – Likelihood of confusion - Trademarks claims – Trademarks Litigations – International Trade mark Law

UNIT-III

Introduction to Copyrights – Principles of Copyright Principles -The subjects Matter of Copy right – The Rights Afforded by Copyright Law – Copy right Ownership, Transfer and duration – Right to prepare Derivative works – Rights of Distribution – Rights of Perform the work Publicity Copyright Formalities and Registrations - Limitations - Copyright disputes and International Copyright Law – Semiconductor Chip Protection Act

UNIT -IV

The law of patents-patent searches –Patent ownership and transfer-Patent infringement-
International Patent Law

UNIT-V

Introduction to Trade Secret – Maintaining Trade Secret – Physical Security – Employee Limitation - Employee confidentiality agreement - Trade Secret Law - Unfair Competition – Trade Secret Litigation – Breach of Contract – Applying State Law

TEXT BOOKS:

1. Debirag E.Bouchoux: “Intellectual Property” 4e . Cengage learning, New Delhi
2. M.Ashok Kumar and Mohd.Iqbal Ali: “Intellectual Property Right” Serials Pub.
3. Cyber Law. Texts & Cases, South-Western’s Special Topics Collections

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

INTERNET OF THINGS
(Open Elective)

Objectives:

- To introduce the terminology, technology and its applications
- To introduce the concept of M2M (machine to machine) with necessary protocols
- To introduce the python scripting language which is used in many IoT devices
- To introduce theraspberry PI platform, that is widely used in IoT application
- 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 enabaled 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

IoT and M2M – Software defined networks, network function virtualization, difference between SDN and NFV for IoTBasics of IoT System Management with NETCOZF, YANG- NETCONF, YANG, SNMP NETOPEER

UNIT III

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 IV

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

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 frameworkDesigning a RESTful web API

TEXT BOOKS:

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

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

INTERNET TECHNOLOGIES AND SERVICES
(Open Elective)

OBJECTIVE :

- The Student who has knowledge of programming with java should be able to develop web based
- Solutions using mult-tier architecturs
- He should have good understanding of different technologies on client and server sidecomponents as follow: Client side: HTML 5,CSS3,javascript Ajax, JQUERY and JSON Server Side:Serlets JSP Database:Myaq With Hibenate And Connection pooling
- Framework: struts with validation framework, Internationalization (I18N)
- SOA: Service oriented architecture, web services fundamentals, axis framework for WS

UNIT I

Client Side Technologies:
Overview of HTML - Common tags, XHTML, capabilities of HTML5 Cascading Style sheets, CSS3 enhancements, linking to HTML Pages, Classes in CSS .Introduction to JavaScripts, variables, arrays, methods and string manipulation, BOM/DOM (Browser/Document Object Model), accessing elements by ID, Objects in JavaScript.Dynamic HTML with JavaScript and with CSS, form validation with JavaScript, Handling Timer Events Simplifying scripting with JQuery, JASON for Information exchange.

UNIT II

Introduction to Java Servlets:
Introduction to Servlets: Lifecycle of a Servlet, Reading request and initialization parameters, Writing output to response, MIME types in response, Session Tracking: Using Cookies and Sessions
Steps involved in Deploying an applicationDatabase Access with JDBC and Connection Pooling

Introduction to XML, XML Parsing with DOM and SAX Parsers in Java Ajax - Ajax programming with JSP/Servlets, creating XML Http Object for various browsers, Sending request, Processing response data and displaying it.Introduction to Hibernate

UNIT III Introduction to JSP:

JSP Application Development: Types of JSP Constructs (Directives, Declarations, Expressions, Code Snippets), Generating Dynamic Content, Exception Handling, Implicit JSP Objects, Conditional Processing, Sharing Data Between JSP pages, Sharing Session and Application Data, Using user defined classes with jsp:useBean tag, Accessing a Database from a JSP

UNIT IV

Introduction to Struts Framework:
Introduction to MVC architecture, Anatomy of a simple struts2 application, struts configuration file, Presentation layer with JSP, JSP bean, html and logic tag libraries, Struts Controller class, Using form data in Actions, Page Forwarding, validation frame work, Internationalization

UNIT V

Service Oriented Architecture and Web Services

Overview of Service Oriented Architecture – SOA concepts, Key Service Characteristics, Technical Benefits of a SOA.Introduction to Web Services– The definition of web services, basic operational model of web services, basic steps of implementing web services. Core fundamentals of SOAP – SOAP Message Structure, SOAP encoding, SOAP message exchange models,Describing Web Services –Web Services life cycle, anatomy of WSDL Introduction to Axis– Installing axis web service framework, deploying a java web service on axis. Web Services Interoperability **–** Creating java and .Net client applications for an Axis Web Service (Note: The Reference Platform for the course will be open source products Apache Tomcat Application Server, MySQL database, Hibernate and Axis).

TEXT BOOKS:

1. Web Programming, building internet applications, Chris Bates 3rd edition, WILEY Dreamtech .

2. The complete Reference Java 7th Edition , Herbert Schildt., TMH.

REFERENCE BOOKS:

1. Programming the world wide web,4th edition,R.W.Sebesta,Pearson

2. Core SERVLETS ANDJAVASERVER PAGES VOLUME 1: CORE
3. TECHNOLOGIES , Marty Hall and Larry Brown Pearson

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

MOBILE COMPUTING
(Open Elective)

UNIT-I
Introduction, Mobile Computing Architecture, Mobile Computing through Telephony, Emerging Technologies

UNIT-II
Global System for Mobile Communications (GSM), Short Message Service (SMS), General Packet Radio Services (GPRS), Wireless Application Protocol (WAP), CDMA and 3G.

UNIT-III
Wireless LAN, Intelligent Network and Internetworking, Client Programming, Programming for PalmOS, Wireless Devices with Symbian OS.

UNIT-IV
J2ME Introduction, J2ME Architecture, MIDLET, MidLet Suite , J2ME Profiles, Wireless Devices with WindowsCE, Voice Over Internet Protocol and Convergence, Session Internet Protocol(SIP),other protocols.

UNIT-V
Multimedia, IP Multimedia Subsystems, Security Issues in Mobile Computing, Next Generation Networks.

TEXTBOOKS:

1. Mobile Computing Technology, Applications and Service Creation by Ashok Talukder , Hasan Ahmed, Roopa R Yavagal.
2. Mobile Computing Principles by Raza B'Far, Cambridge.
3. Mobile Computing by Raj Kamal 2e.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

MOBILE APPLICATION SECURITY
(Open Elective)

Objectives:

- To understand the mobile issues and development strategies
- To understand the WAP and mobile security issues
- To understand the Bluetooth security issues.

UNIT I:

Top Mobile Issues and Development Strategies: Top Issues Facing Mobile Devices, Physical Security , Secure Data Storage (on Disk), Strong Authentication with Poor Keyboards , Multiple-User Support with Security, Safe Browsing Environment , Secure Operating Systems, Application Isolation, Information Disclosure, Virus, Worms, Trojans, Spyware, and Malware , Difficult Patching/Update Process, Strict Use and Enforcement of SSL, Phishing , Cross-Site Request Forgery (CSRF), Location Privacy/Security, Insecure Device Drivers, Multifactor Authentication, Tips for Secure Mobile Application Development .

UNIT II:

WAP and Mobile HTML Security :WAP and Mobile HTML Basics , Authentication on WAP/Mobile HTML Sites , Encryption , Application Attacks on Mobile HTML Sites ,Cross-Site Scripting , SQL Injection , Cross-Site Request Forgery , HTTP Redirects , Phishing , Session Fixation , Non-SSL Login , WAP and Mobile Browser Weaknesses , Lack of HTTPOnly Flag Support , Lack of SECURE Flag Support , Handling Browser Cache , WAP Limitations.

UNIT III:

Bluetooth Security: Overview of the Technology , History and Standards , Common Uses , Alternatives , Future , Bluetooth Technical Architecture , Radio Operation and Frequency, Bluetooth Network Topology , Device Identification , Modes of Operation , Bluetooth Stack ,Bluetooth Profiles , Bluetooth Security Features , Pairing , Traditional Security Services in Bluetooth, Security “Non-Features” , Threats to Bluetooth Devices and Networks, Bluetooth Vulnerabilities , Bluetooth Versions Prior to v1.2, Bluetooth Versions Prior to v2.1.

UNIT IV:

SMS Security: Overview of Short Message Service, Overview of Multimedia Messaging Service, Wireless Application Protocol (WAP), Protocol Attacks , Abusing Legitimate Functionality, Attacking Protocol Implementations, Application Attacks , iPhone Safari , Windows Mobile MMS, Motorola RAZR JPG Overflow, Walkthroughs ,Sending PDUs ,Converting XML to WBXML .

UNIT V

Enterprise Security on the Mobile OS: Device Security Options , PIN , Remote , 346 Secure Local Storage , Apple iPhone and Keychain , Security Policy Enforcement ,Encryption ,Full Disk Encryption ,E-mail Encryption , File Encryption , Application Sandboxing, Signing, and Permissions , Application Sandboxing , Application Signing , Permissions , Buffer Overflow Protection ,Windows Mobile , iPhone ,Android ,BlackBerry , Security Feature Summary.

TEXT BOOKS:

1. “Mobile Application Security”, Himanshu Dwivedi, Chris Clark, David Thiel, TATA McGRAW-Hill.

REFERENCE BOOKS:

1. “Mobile and Wireless Network Security and Privacy”, Kami S.Makki,et al, Springer.
2. “Android Security Attacks Defenses”, Abhishek Dubey, CRC Press.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

OPEN STACK CLOUD COMPUTING
(Open Elective)

UNIT I

Keystone OpenStack Identity Service. Installing OpenStack Identity service. Starting OpenStack Image Service. Installing OpenStack Image Service, Configuring OpenStack Image Service with MySQL, Configuring OpenStack Image Service with OpenStack Identity Service, Managing images with OpenStack Image Service, Registering a remotely stored image, Sharing images among tenants, Viewing shared images. Starting OpenStack Compute. Installing OpenStack Compute Controller services, Creating a sandbox Compute server with VirtualBox and Vagrant, Installing OpenStack Compute packages, Stopping and starting Nova services.Installation of command-line tools on Ubuntu. OpenStack Compute services. Compute Managing security groups. Launching our first Cloud instance, Terminating your instance.

UNIT II.

Installing OpenStack Object Storage. Configuring OpenStack Object Storage Service, Making rings, Stopping and starting OpenStack Object Storage. Configuring OpenStack Object Storage with OpenStack Identity Service, Setting up SSL access, Testing OpenStack Object Storage.

Using OpenStack Object Storage. Installing the swift client tool. Creating containers, Uploading objects, Listing containers and objects, Downloading objects, Deleting containers and objects. Using OpenStack Object Storage ACLs.Administering OpenStack Object Storage. Preparing drives for OpenStack Object Storage, Managing OpenStack Object Storage cluster with swift-init, Checking cluster health. Benchmarking OpenStack Object Storage. Detecting and replacing failed hard drives , Collecting usage statistics.

UNIT III.

Starting OpenStack Block Storage. Configuring OpenStack Compute for Cinder volume. OpenStack Networking. Configuring Flat networking with DHCP. Configuring VLAN Manager networking. Configuring per tenant IP ranges for VLAN Manager. Automatically assigning fixed networks to tenants, Modifying a tenant's fixed network, Manually associating floating IPs to instances, Manually disassociating floating IPs from instances, Automatically assigning floating IPs. Creating a sandbox Network server for Neutron with VirtualBox and Vagrant. Installing and configuring OVS for Neutron. Creating a Neutron network 203, Deleting a Neutron network, Creating an external Neutron network.

UNIT IV

Using OpenStack Dashboard. Installing OpenStack Dashboard, Using OpenStack Dashboard for key management, Using OpenStack Dashboard to manage Neutron networks, Using OpenStack Dashboard for security group management, Using OpenStack Dashboard to launch instances, Using OpenStack Dashboard to terminate instances, Using OpenStack Dashboard for connecting to instances using VNC, Using OpenStack Dashboard to add new tenants, Using OpenStack Dashboard for user management.

Automating OpenStack Installations. Installing Opscode Chef Server. Installing Chef Client, Downloading cookbooks to support DHCP, Razor, and OpenStack. Installing PuppetLabs Razor and DHCP from cookbooks. Setting up a Chef environment for OpenStack. Booting the first OpenStack node into Razor, Defining a Razor broker, model, and policy. Monitoring the node installation. Using Chef to install OpenStack, Expanding our OpenStack environment.

UNIT V.

Highly Available OpenStack. Using Galera for MySQL clustering. Configuring HA Proxy for MySQL Galera load balancing, Installing and setting up Pacemaker and Corosync, Configuring Keystone and Glance with Pacemaker and Corosync, Bonding network interfaces for redundancy.Troubleshooting. Understanding logging. Checking OpenStack services. Troubleshooting OpenStack Compute services. Troubleshooting OpenStack Object Storage services. Troubleshooting OpenStack Dashboard. Troubleshooting OpenStack Authentication, Troubleshooting OpenStack Networking, Submitting Bug reports, Getting help from the commUNITy.
Monitoring. Monitoring OpenStack services with Nagios. Monitoring Compute services with Munin. Monitoring instances using Munin and Collectd. Monitoring the storage service using StatsD/Graphite. Monitoring MySQL with Hyperic.

TEXT BOOKS:

1. OpenStack Cloud Computing Cookbook - Second Edition, Kevin Jackson , Cody Bunch, October 2013, Packt Publishing-OpenSource.

REFERENCE BOOKS: https://www.packtpub.com/virtualization-and-cloud/openstack-cloud-computing-cookbook-second-edition

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

PRINCIPLES OF INFORMATION SECURITY

(Open Elective)

UNIT – I

Introduction to Information Security, Need For Security,

UNIT –II

Legal Ethical and Professional Issues in Information Security, Planning For Security.

UNIT – III

Risk Management, Security Technology: Firewalls and VPNs, Security Technology: Intrusion Detection and Prevention Systems, and Other Security Tools.

UNIT – IV

Cryptography, Physical Security, Implementing Information Security,

UNIT – V

Security and Personnel, Information Security Maintenance.

TEXT BOOKS:

1. Principles of Information Security by Whitman, Thompson

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

SOCIAL MEDIA INTELLIGENCE
(Open Elective)

UNIT – I

The Beginnings of Social Media Intelligence: What is Social Media monitoring? Anecdotal referencing of Social Media Comments, Text Mining, Some Simple Metrics, Using Social Media as Early Warning System.

Fundamental of Opinion Formation: Affecting Opinion versus Biasing Expression, How Do We Form Opinions?, How Do Expectations Affect Opinion?, How Do Expertise and Knowledge Influence How We Form Opinions?, Opinion Formation in a Social Context, Bandwagon behavior and Information Cascades, Implications for Social Media Intelligence.

UNIT – II

Why Do We Share our Opinions : Poster versus Lurkers, What Motivates Us to Post/, Posting Motivations and Selection effects, Implications for Social Media Intelligence.
The Social effects of Strangers : How Does Social Context Affect Our Behavior?, How Influential is the Social Context/, How Does Social Context Affect Opinion Expression/, Bandwagon Behavior in Opinion expression, Differentiating Our opinions, Multiple Audience Effects, /can We Trust the Wisdom of Crowds.

UNIT – III

Opinion Ecosystems and the Evolution Within : Life Cycle Dynamics, Preference Mismatching and Sequential Dynamics, Social Dynamics, Are Social Media CommUNITies the Cause of Opinion Radicalization ?, Online Echo Chambers, Implications for Social Media Monitoring and Metrics.

Are Social Media Fragmenting the Population ? : Self-Organization, Birds of a Feather Flock Together, Geography No Longer Defines Our CommUNITies, The influential Hypothesis, The New Influential, How Can We Identify Influentials, Influence in e-Commerce, Some Concluding Remarks.

UNIT – IV

Managing Social Media CommUNITies for Better Social Media Intelligence: Creating an Inviting Environment, The Benefits of a Well-Managed Opinion CommUNITy (and the Costs of Not Managing the CommUNITy at All) Quality of Intelligence Depends on the Quality of the Opinion CommUNITy, Creating and Manipulating Buzz, Buzz Campaign or Fraud?, Identifying Fraudulent Opinions

Cutting Through the Online Chatter : A New Paradigm for Marketing Research, Measure What Matters, Cast a Wide Net, Analyze the Text, Understand the biases, Establish Links to Performance metrics.

UNIT – V

Intelligence Integration : Overview of Marketing Research Methods, Using Social Media for Marketing research, Tracking Brand Health, Understanding Market Structure, Social Shopping, Integration with Data from Other Parts of the Organization, Intelligence Dashboards.

Building Social Media Intelligence into Our Strategies : How Can Social Media Intelligence Help Integrate an Organization's Strategy?, Multichannel Strategies, Rapid Response System, Integrated CRM, Leveraging Social Data, Seeding Strategies.

Moving from Social Media monitoring to Social Media Intelligence : Social Media Intelligence today, Social Media Intelligence tomorrow, Building on the Science of Opinion, tapping into Opinion Ecosystems, Developing an Integrated Strategy.

REFERENCE BOOKS:

1. SOCIAL MEDIA INTELLIGENCE : by Wendly W.Moe, David A. Schweidel, Cambride University, edition

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

STORAGE AREA NETWORKS
(Open Elective)

Objectives:

- To understand storage area networks characteristics and components.
- To become familiar with the SAN vendors and their products
- To learn fibre channel protocols and how SAN components use them to communicate with each other
- To become familiar with cisco MDS 9000 multilayer directors and fabric switches thoroughly learn cisco SAN-OS features.
- To understand the use of all SAN-OS commands. Practice variations of SANOS features

UNIT I

Introduction to Storage Technology Review data creation and the amount of data being created and understand the value of data to a business, challenges in data storage and data management, Solutions available for data storage, Core elements of a data center infrastructure, role of each element in supporting business activities

UNIT II

Storage Systems Architecture Hardware and software components of the host environment, Key protocols and concepts used by each component ,Physical and logical components of a connectivity environment ,Major physical components of a disk drive and their function, logical constructs of a physical disk, access characteristics, and performance Implications, Concept of RAID and its components , Different RAID levels and their suitability for different application environments: RAID 0, RAID 1, RAID 3, RAID 4, RAID 5, RAID 0+1, RAID 1+0, RAID 6, Compare and contrast integrated and modular storage systems ,High-level architecture and working of an intelligent storage system

UNIT III

Introduction to Networked Storage Evolution of networked storage, Architecture, components, and topologies of FC-SAN, NAS, and IP-SAN, Benefits of the different networked storage options, Understand the need for long-term archiving solutions and describe how CAS fulfills the need , Understand the appropriateness of the different networked storage options for different application environments

UNIT IV

Information Availability & Monitoring & Managing Data enter List reasons for planned/unplanned outages and the impact of downtime, Impact of downtime, Differentiate between business continuity (BC) and disaster recovery (DR) ,RTO and RPO, Identify single points of failure in a storage infrastructure and list solutions to mitigate these failures, Architecture of backup/recovery and the different backup/recovery topologies , replication technologies and their role in ensuring information availability and business continuity, Remote replication technologies and their role in providing disaster recovery and business continuity capabilities

Identify key areas to monitor in a data centre, Industry standards for data centre monitoring and management, Key metrics to monitor for different components in a storage infrastructure, Key management tasks in a data centre.

UNIT V

Securing Storage and Storage Virtualization Information security, Critical security attributes for information systems, Storage security domains, List and analyzes the common threats in each domain, Virtualization technologies, block-level and file-level virtualization technologies and processes Case Studies The technologies described in the course are reinforced with EMC examples of actual solutions. Realistic case studies enable the participant to design the most appropriate solution for given sets of criteria.

TEXT BOOKS:

1. EMC Corporation, Information Storage and Management, Wiley.

REFERENCE BOOKS:

1. Robert Spalding, “Storage Networks: The Complete Reference“, Tata McGraw Hill, Osborne,
2003.
2. Marc Farley, “Building Storage Networks”, Tata McGraw Hill, Osborne, 2001.
3. Meeta Gupta, Storage Area Network Fundamentals, Pearson Education Limited, 2002.

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

WEB USABILITY

(Open Elective)

UNIT I

Introduction to Usability, Human Factors,

UNIT II

User-Centered Design, Usability Aware Design,

UNIT III

Accessibility, Understanding your Users and Goals,

UNIT IV

Heuristic Evaluation, Usability Testing,

UNIT V

Other Tools and Techniques, Transferring Data into Change

TEXT BOOKS:

Web Usability Hand Book by Mark Pearrow, Thomson Delmar learning

J.B. INSTITUTE OF ENGINEERING & TECHNOLOGY
UGC AUTONOMOUS

MCA

ORGANIZATION STRUCTURE AND PERSONNEL MANAGEMENT
(Open Elective)

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.

UNIT-I

Introduction to Management & Organisation: Concepts of Management and organization- nature, importance and Functions and Theories of Management, Systems Approach to Management, Leadership Styles, Social responsibilities of Management. Designing Organisational Structures: Basic concepts related to Organisation - Departmentation and Decentralisation, Types and Evaluation of mechanistic and organic structures of organisation and suitability.

UNIT-II

Personnel Management-I: Evolution, objectives, Personnel policies, Personnel Management vs HRM, Position of the personnel department in the organization. Role of Personnel Manager as Line manager and Staff Manager. Manpower planning, recruitment and selection, interviewing techniques, transfers, promotion and its policies.

UNIT-III

Personnel Management-II: Training and development: Objectives and policies planning, organizing the training department, training manager and his job, Methods of on- and off- the- job training, career planning, objectives and methods of performance appraisal.

UNIT-IV

Understanding Human Behaviour: Personality – Johari Window- Transactional Analysis. Perception: Perceptual process, Development of Attitudes and Values, Understanding Group Dynamics, Team Effectiveness, Strategies to deal with conflicts and stress.

UNIT-V

Contemporary Strategic Issues: Total Quality Management (TQM), Six sigma and People Capability Maturity Model (PCMM) Levels, Performance Management, Business Process outsourcing (BPO), Business Process Re-engineering and Bench Marking, Balanced Score Card.

TEXT BOOKS

1. A.R.Aryasri, Organisational Structure and Personnel Management, TMH, 2009.
2. Robbins: Organisational Behaviour, Pearson, 2008

REFERENCES

1. Udai pareek: Understanding Organisational Behaviour, PHI, 2009
2. P.Subbarao, Management and Organisational Behaviour, HPH, 2009
3. Mamoria & Gankar, Personnel Management, HPH, 2009
4. VSP Rao, Organisational Behaviour, Excel, 2009