

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi – 590 018, Karnataka

Phone: 0831 -2498100/2405468 Fax;0831-2405467

E-Mail:registrar@vtu.ac.in, Web: www.vtu.ac.in



MCA

(Master of Computer Applications)

**Regulations, Scheme of Teaching and Examinations,
and Syllabus**

Governing the Degree of

MASTERS OF COMPUTER APPLICATIONS (MCA)

Under Outcome Based Education (OBE)

And

Choice Based Credit System (CBCS) Scheme

Effective from the Academic Year 2018 - 19

September - 2018

Price : Rs.100/-

For copies Contact :

REGISTRAR

Visvesvaraya Technological University

“Jnana Sangama”, BELAGAVI-590 018.

Published by :

REGISTRAR

Visvesvaraya Technological University

“Jnana Sangama”, BELAGAVI-590 018.

© Copyright 2018-19

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi – 590 018, Karnataka

Phone: 0831 -2498100/2405468 Fax:0831-2405467

E-Mail: registrar@vtu.ac.in, Web: www.vtu.ac.in

CONTENTS

Regulation Clause	Title	Page Number
--	Definitions of Keywords	04
18OMC1.0	Title, Duration and Credits of the Programme of Study	08
18OMC2.0	Eligibility for Admission (As per the Government orders issued from time to time)	09
18OMC3.0	Courses	11
18OMC4.0	Internship	13
18OMC5.0	Seminar	14
18OMC6.0	Project	14
18OMC7.0	Computation of SGPA and CGPA	19
18OMC8.0	Conversions of Grades into Percentage and Class Equivalence	21
18OMC9.0	Continuous Internal Evaluation and Semester End Evaluation	22
18OMC10.0	Eligibility for Passing and Award of Degree	24
18OMC11.0	Attendance Requirement	26
18OMC12.0	Promotion and Eligibility	28
18OMC13.0	Temporary Discontinuation/Break in the Program	29
18OMC14.0	Award of Prizes, Medals and Ranks	30
18OMC15.0	Applicability and Power to Modify	31
	Scheme of Teaching and Examinations	32
	Syllabus	33

Visvesvaraya Technological University, Belagavi
Regulations Governing The Degree of Master of Computer Applications (MCA)
Under Outcome Based Education (OBE) And Choice Based Credit System (CBCS) Scheme
Effective From Academic Year 2018 -19

Definitions of Keywords

The following are the definitions/descriptions that have been followed for the different terms used in the Regulations of Master of Computer Applications (MCA) Programme:

- 1. Programme:** Is an educational programme in Computer Applications leading to award of Degree. It involves events/activities, comprising of lectures/ tutorials/ laboratory work/ field work, outreach activities/ project work/ vocational training/ viva/ seminars/ Internship/ assignments/ presentations/ quiz/self-study etc., or a combination of some of these.
- 2. Branch: Means MCA Degree Programme.**
- 3. Semester:** Refers to one of the two sessions of an academic year (vide: serial number 4), each session being of sixteen weeks duration (with working days greater than or equal to ninety). The odd semester may be scheduled from August and even semester from February of the year.
- 4. Academic Year:** Refers to the sessions of two consecutive semesters (odd followed by an even) including periods of vacation.
- 5. Course:** Refers to usually referred to as 'papers' and is a component of a programme. All Courses need not carry the same weight. The Courses should define learning objectives and learning outcomes. A Course may be designed to comprise lectures/ tutorials/ laboratory work/ field work/ outreach activities/project work/ vocational training/ viva/ seminars/ term papers/assignments/ presentations/ quiz/self-study etc., or a combination of some of these.
- 6. Credit:** Refers to 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 lecture or two hours of laboratory/practical Courses/ tutorials/ fieldwork per week etc.

- 7. Audit Courses:** Means Knowledge/ Skill enhancing Courses without the benefit of a grade or credit for a Course.
- 8. Choice Based Credit System (CBCS):** Refers to customizing the Course work, through Core, Elective and soft skill Courses, to provide necessary support for the students to achieve their goals.
- 9. Course Registration:** Refers to formal registration for the Courses of a semester (Credits) by every student under the supervision of a Faculty Advisor (also called Mentor, Counselor etc.,) in each Semester for the Institution to maintain proper record.
- 10. Course Evaluation:** Means Continuous Internal Evaluation (CIE) and Semester End Examinations (SEE) to constitute the major evaluations prescribed for each Course. CIE and SEE to carry 40 % and 60 % respectively, to enable each Course to be evaluated for 100 marks, irrespective of its Credits.
- 11. Continuous Internal Evaluation (CIE):** Refers to evaluation of students' achievement in the learning process. CIE shall be by the Course Instructor and includes tests, homework, problem solving, group discussion, quiz, mini-project and seminar throughout the Semester, with weightage for the different components being fixed at the University level.
- 12. Semester end examinations (SEE):** Refers to examination conducted at the University level covering the entire Course Syllabus. For this purpose, Syllabi to be modularized and SEE questions to be set from each MODULE, with a choice confined to the concerned MODULE only. SEE is also termed as University examination.
- 13. First Attempt:** Refers to a student who has completed all formalities and has become eligible to attend the SEE and has attended at least one head of passing, such attempt shall be considered

as first attempt.

14. Credit Based System (CBS): Refers to quantification of Course work, after a student completes teaching – learning process, followed by passing in both CIE and SEE. Under CBS, the requirement for awarding degree is prescribed in terms of total number of credits to be earned by the students.

15. Credit Representation: Refers to Credit Values for different academic activities considered, as per the Table.1. Credits for seminar, project phases, project viva–voce and internship shall be as specified in the Scheme of Teaching and Examination.

Theory/Lectures (L) (hours/week/Semester)	Tutorials (T) (hours/week/Semester)	Laboratory/Practical (P) (hours/week/Semester)	Credits (L:T:P)	Total Credits
4	0	0	4:0:0	4
3	0	0	3:0:0	3
2	2	0	2:1:0	3
2	0	2	2:0:1	3
2	2	2	2:1:1	4
0	0	6	0:0:3	3

16. Letter Grade: It is an index of the performance of students in a said Course. Grades are denoted by letters S, A, B, C, D, E and F.

17. Grading: Grade refers to qualitative measure of achievement of a student in each Course, based on the percentage of marks secured in (CIE plus SEE). Grading is done by Absolute Grading [Refer: 18OMC7.0]. The rubric attached to letter grades are as follows:

S – Outstanding, A – Excellent, B – Very Good, C – Good, D – Above Average, E – Average and F – Fail.

18. Grade Point (GP): Refers to a numerical weightage allotted to each letter grade on a 10-point scale as under.

Letter Grade	S	A	B	C	D	E	F
Grade Point	10	09	08	07	06	04	00

19. Passing Standards: Refers to passing a Course only when getting GP greater than or equal to 04 (as per serial number 18).

20. Credit Point: Is the product of grade point (GP) and number of credits for a Course i.e.,

Credit points (CrP) = GP × Credits for the Course.

21. Semester Grade Point Average (SGPA): Refers to a measure of academic performance of student/s in a semester. It is the ratio of total credit points secured by a student in various Courses of a semester and the total Course credits taken during that semester. [Refer: 18OMC7.0]

22. Cumulative Grade Point Average (CGPA): Is a measure of overall cumulative performance of a student over all semesters. The CGPA is the ratio of total credit points earned by a student in various Courses in all semesters and the sum of the total credits of all Courses in all the semesters. It is expressed up to two decimal places. [Refer: 18OMC7.0]

23. Grade Card: Refers to a certificate showing the grades earned by a student. A grade certificate shall be issued to all the registered students after every semester end examination. The grade certificate will display the programme details (Course code, title, number of credits, grades secured) along with SGPA of that semester and CGPA earned till that semester.

24. University: Visvesvaraya Technological University (VTU), Belagavi.



18OMC1.0	Title, Duration and Credits of the Programme of Study
18OMC1.1	The Programme shall be called Master of Computer Applications, abbreviated as MCA.
18OMC1.2	<p>(i) The program to which students are admitted to I semester of the programme shall be of three academic year duration divided into six semesters and each semester shall be of 16 weeks duration.</p> <p>(ii) The programme to which students are admitted to III semester of the programme, under lateral entry scheme, shall be of two academic year duration divided into four semesters and each semester is of 16 weeks duration.</p> <p>The four semesters of II and III year of the programme are common to students admitted under (i) and (ii).</p>
18OMC1.3	<p>Maximum Duration for Programme Completion:</p> <p>(a) Candidates admitted to I year of the programme shall be allowed a maximum duration of six years to become eligible for the award of MCA degree, failing which he/she may discontinue the programme or register once again as a fresh candidate to I semester.</p> <p>(b) Candidates admitted to II year of the programme under lateral entry scheme shall complete the course within four academic years from the year of first admission, failing which he /she has to discontinue the programme or register once again as a fresh candidate to III semester.</p>
18OMC1.4	<p>Prescribed Number of Credits for the Programme:</p> <p>The number of credits to be completed for the award of degree shall be 132.</p>
18OMC1.5	The Calendar of events in respect of the Programme shall be notified by the University in advance.

18OMC2.0	Eligibility for Admission (As per the Government orders issued from time to time)
18OMC2.1	<p>Admission to I year /I semester MCA: A candidate who has passed any recognized under graduate examination or equivalent examination with Mathematics or Statistics or Computer Science or Computer Applications or Computer Programming or Business Mathematics or Business Statistics as one of the optional subjects and obtained an aggregate minimum of 50% marks taken together in all the subjects in all the years of the Degree Examination is eligible for admission to MCA courses and 45% of marks in qualifying examination (QE) in case of SC, ST and Category-I of Karnataka candidates. Provided that in respect of candidate who has studied and passed one of the subjects specified above in Pre-University Course with fifty percent of marks in that subject shall also be considered for admission. 45% of marks in case of SC, ST and Category-I of Karnataka candidates.</p> <p>Admission to II year /III semester MCA(Lateral entry): A candidate who has passed recognized Bachelor's Degree of minimum of 3 years duration in BCA, B.Sc (I.T. / Computer Science) with Mathematics at 10 + 2 level or at Graduate Level and obtained an aggregate minimum of 50% marks taken together in all the subjects in all the years of the Degree Examination is eligible for admission to MCA courses (45% of marks in QE in case of SC, ST and Category-I of Karnataka candidates) Reservation is applicable only for Karnataka Candidates.</p>
18OMC2.2	For admissions under PG CET qualification and Roaster system of Government of Karnataka:

	<p>There shall be an Entrance Examination (PGCET) for admission to the MCA programme. A candidate seeking admission to MCA Programme offered in any of the Engineering Colleges affiliated to VTU shall appear for this Examination. For admission under Government quota, ranks obtained in PGCET entrance exam, conducted by Karnataka Examination Authority (KEA), shall be considered.</p> <p>For admissions under Management Quota:</p> <p>The candidates should have appeared for the Entrance Examination conducted by KEA (PGCET)/Karnataka Management Aptitude Test (KMAT) or appeared and qualified under any approved entrance examination conducted by the authority recognized by Government of Karnataka/VTU /any other University of Karnataka state.</p> <p>Further, there shall be an Admissions Committee for the MCA Program consisting of the Principal of the College as the Chairman, Head of the concerned Department and one senior staff member of the concerned Department. The Admissions Committee conducts the interview and selects the candidates for admission.</p>
<p>18OMC2.3</p>	<p>(i) The candidates from Universities other than the Universities of Karnataka shall have to obtain Eligibility Certificate from the VTU to seek admission to MCA program in any of the college affiliated to VTU.</p> <p>(ii) The candidates from foreign countries shall have to obtain Eligibility Certificate from the VTU to seek admission to MCA program in any of the college affiliated to VTU. Further, they have to produce equivalence certificate from the association of Indian Universities.</p>

<p>18OMC2.4</p>	<p>The intake under various categories (regular, Lateral entry, SC/ST and category I) shall be as sanctioned by the AICTE, State Government and VTU, from time to time.</p>
<p>18OMC2.5</p>	<p>Admission to vacant seats:</p> <p>Seats remaining vacant (unfilled), after the completion of PG admission process by Karnataka Examination Authority, shall be filled by the Institution by inviting applications through Press notification. The seats shall be filled by Candidates preferably who have PGCET score. In the absence of such Candidates, admission shall be based on merit in the entrance test conducted at the Institution level. An Admissions Committee, consisting of the Principal of the College, Head of the concerned Department and the subject experts, shall be in charge of admissions.</p>
<p>18OMC3.0</p>	<p>Courses</p>
<p>18OMC3.1</p>	<p>The curriculum of the Programme shall be any combination of following type of courses:</p> <p>i) Professional Core Courses (PC) - relevant to the chosen specialization/ branch [May be split into Hard (no choice) and Soft (with choice), if required]. The core course is to be compulsorily studied by a student and is mandatory to complete the requirements of a programme in a said discipline of study.</p> <p>ii) Professional Electives Courses (PE) - relevant to the chosen specialization/ branch: these are the courses, which can be chosen from the pool of papers. It shall be supportive to the discipline/ providing extended scope/enabling an exposure to some other discipline / domain / nurturing student skills.</p> <p>iii) Open Electives Courses (OE), from other technical and/ or emerging specialization areas.</p>

	<p>iv) Project Work, Seminar.</p> <p>v) Audit Courses (AC):</p> <p>(a) The Audit course can be any credit course offered by the program to which the Candidate is admitted (other than the courses considered for completing the prescribed program credits).</p> <p>(b) The students admitted to I semester and interested in audit courses can register for one audit course at a time during II to VI semester.</p> <p>The students admitted to III semester under lateral entry scheme and interested in audit courses can register for one audit course at a time during IV to VI semester.</p> <p>Students who have registered to audit courses, considered on par with students registered to the same course for credit, have to satisfy attendance and CIE requirements. However, they need not have to appear for SEE.</p> <p>(c) Registration for any audit course, in writing, shall be completed at the beginning of each semester. The Institution should intimate the Registrar (Evaluation) about the registration at the beginning of the semester and obtain a formal approval for inclusion of the audit course/s in the Grade cards/ Transcripts issued to the students.</p> <p>vi) Professional training/Internship (referred to as Internship): Preferably at an industry/R and D organization/IT company/ Government organization/Business organization of significant repute for a specified period mentioned in Scheme of Teaching and Examination.</p>
18OMC3.2	A candidate shall exercise his/her option in respect of the electives and register for the same before the

	beginning of the concerned semester. The candidate may be permitted to opt for change of elective subject within 10 days from the date of commencement of the semester as per the calendar of the University.
18OMC3.3	<p>The minimum number of students to be registered for an Elective to be offered shall not be less than ten.</p> <p>However, the above condition shall not be applicable when the class strength is less than ten.</p>
18OMC4.0	Internship
18OMC4.1	<p>Internship: The student shall undergo Internship for the specified period as per the Scheme of Teaching and Examination.</p> <ol style="list-style-type: none"> 1. The internship shall be carried out in any industry/R&D Organization/Business Organization /Research Institute/Institute of national and international repute Business organization/ recognized national and international Professional Bodies, Societies or Organizations. 2. The Department/college shall nominate a faculty to facilitate, guide and supervise students under internship. 3. The students shall report the progress of the internship to the internal guide in regular intervals and seek his/her advise. 4. The internship shall be completed during the period specified in Scheme of Teaching and Examination. 5. After completion of internship, students shall submit a report to the Head of the Department with the approval of both internal and external guides. 6. There shall be 100 marks for CIE (Presentation: 50 marks, Report: 50 marks) and shall be evaluated by the internal panel.

	<p>7. The students are permitted to carry out the internship anywhere in India or abroad.</p> <p>The University will not provide any kind of Financial Assistance to any student for internship</p>
18OMC4.2	<p>Failing to undergo Internship: Internship is one of the head of passing. Completion of internship is mandatory. If any student fails to undergo/complete the internship, he/she shall be considered as failed in that Course and the prescribed credits shall not be awarded in that Course. The student, however, can submit the project dissertation and appear for viva voce.</p> <p>The student shall be eligible for the internship credits only after satisfying the conditions prescribed for the same during the subsequent academic year. The reappearance shall be considered as an attempt.</p>
18OMC5.0	Seminar
18OMC5.1	<p>Seminar: Seminar is one of the head of passing.</p> <p>i) Each candidate shall deliver seminar as per the Scheme of Teaching and Examination on the topics chosen from the relevant fields for about 30 minutes.</p> <p>ii) The Head of the Department shall make arrangements for conducting seminars through concerned faculty members of the Department. The committee constituted for the purpose by the Head of the Department shall award the CIE marks for the seminar. The committee shall consist of three faculty from the Department and the senior most acting as the Chairman/Chairperson. [To be read along with 18OMC10.3]</p>
18OMC6.0	Project
18OMC6.1	Project work and Dissertation:

	<p>Each candidate shall carry out the project work independently as per Scheme of Teaching and Examinations under the guidance of one of the faculty members of the Department in the Institution of study. If the project is of inter-disciplinary nature, a co-guide shall be taken from the other concerned department.</p> <p>The topic and title of the dissertation shall be chosen by the candidate in consultation with the guide and co-guide, if any, before the commencement of VI semester.</p> <p>The subject and topic of the dissertation shall be from the major field of studies of the candidate. Modification of only the title but not the field of work may be permitted at the time of final submission of dissertation report during sixth semester. If dissertation has to be carried out in any industry/R&D labs, outside the campus, permission shall be taken from the Principal to that effect.</p> <p>The Principal, shall submit to the University a list showing the name of the student, University Seat Number, title of the project, name/s of the guide/co-guide, at the time of submission of project report to the University.</p>
18OMC6.2	<p>Project is one of the head of passing.</p> <p>The candidate shall submit a soft copy (CD) of the dissertation work to the University. The CD should contain the entire dissertation in monolithic form as a PDF file (not separate chapters).</p> <p>The Guide, after checking the report for completeness shall upload the dissertation along with name, University Seat Number, address, mobile number of the candidate, etc., as prescribed in the form available on online dissertation evaluation portal.</p>

18OMC6.3	<p>Plagiarism Check</p> <p>Once the Guide uploads the dissertation, the same shall be linked for plagiarism check. The allowable plagiarism index is less than or equal to 25%.</p> <p>If the check indicates a plagiarism index greater than 25% :</p> <p>* For the first time, the candidate has to resubmit the dissertation, to the Registrar (Evaluation), Regional Center/Head Office, VTU along with the penal fees of Rs. 2000/- (Rupees Two thousand only).</p> <p>* For the second time, the candidate has to resubmit the dissertation along with the penal fees of Rs. 4000/- (Rupees four thousand only).</p> <p>* If the dissertation is rejected again during second resubmission with reference to plagiarism index, the candidate shall redo the project and submit after a semester's time subject to provisions of 18OMC1.3.</p>
18OMC6.4	<p>The dissertation shall be sent through email for evaluation to two examiners - one internal examiner (guide/co-guide) and one external examiner (first) appointed by the University. The evaluation of the dissertation shall be made independently by each examiner.</p>
18OMC6.5	<p>Examiners shall evaluate the dissertation normally within a period of not more than two weeks from the date of receipt of dissertation through email.</p>
18OMC6.6	<p>The examiners shall independently submit the marks through the specified link.</p>
18OMC6.7	<p>Average of the marks awarded by the two Examiners shall be the final evaluation marks for the dissertation.</p>
18OMC6.8	<p>(a) Viva-voce examination of the candidate shall be</p>

	<p>conducted as per 18OMC6.10, if the dissertation work and the reports are accepted by the external examiner (first).</p> <p>(b) If the external examiner (first) finds that the dissertation work and the report are not up to the expected standard and the minimum passing marks cannot be awarded, the dissertation shall not be accepted for SEE.</p> <p>The external examiner (first) can recommend for modifications/suggestions of dissertation or totally reject the dissertation. The examiner shall offer suggestions for improvement of the dissertation for re-submission or list the reasons for rejection of the dissertation.</p> <p>(c) The resubmitted dissertation incorporating the modifications/suggestions [as per 18OMC6.8 (b)] of the external examiner (first) and satisfying the provision 18OMC6.3 shall be sent again to the external examiner (first) for evaluation. If the dissertation and the report are accepted by the external examiner (first), Viva-voce examination of the candidate shall be conducted as per 18OMC6.10.</p> <p>(d) In case of rejection of dissertation by the external examiner (first), the same will be sent to a Second Examiner (external) approved by the University. The decision of the Second Examiner (external) is final. If the dissertation and the report are accepted by the Second Examiner (external), Viva-voce examination of the candidate shall be conducted as per 18OMC6.10. If the Second Examiner (external) rejects the dissertation and the report, the candidate shall have to carry out the dissertation work once again and submit the dissertation subject to provisions of 18OMC1.3. In such cases of rejection, the candidate shall redo the entire procedure starting from</p>
--	---

	<p>the submission of dissertation in soft copy.</p> <p>(e) In case of rejection of dissertation, with reasons, by the external examiner (first) [as per 18OMC6.8 (b)], the same will be sent to a Second Examiner (external)[not necessarily the same examiner considered under 18OMC6.8 (d)] approved by the University. The decision of the Second Examiner (external) is final. If the dissertation and the report are accepted by the Second Examiner (external), Viva-voce examination of the candidate shall be conducted as per 18OMC6.10. If the Second Examiner (external) rejects the dissertation and the report, the candidate shall have to carry out the dissertation work once again and submit the dissertation subject to provisions of 18OMC1.3. In such cases of rejection, the candidate shall redo the entire procedure starting from the submission of dissertation in soft copy.</p>
18OMC6.9	The candidate, whose dissertation is rejected, can rework on the same topic or choose another topic of dissertation under the same Guide or new Guide if necessary. In such an event, the report shall be submitted subject to provisions of 18OMC1.3.
18OMC6.10	<p>Viva-voce examination of the candidate shall be conducted by the external examiner and internal examiner/ guide.</p> <p>Internal examiner as per the direction of the University shall have to arrive at a mutually convenient date for the conduct of viva-voce examination of the concerned candidate with an intimation to the Registrar (Evaluation). In case one of the examiners expresses his/her inability to attend the viva-voce, the Registrar (Evaluation) shall appoint a substitute examiner in his/her place.</p>
18OMC6.11	The relative weights for the evaluation of dissertation

	and the performance at the viva voce shall be as per the scheme of teaching & examination.																																							
18OMC6.12	The marks awarded by both the Examiners at the viva voce Examination shall be sent jointly to the University immediately after the examination.																																							
18OMC6.13	Examination fee as fixed from time to time by the University for evaluation of dissertation report and conduct of viva voce shall be remitted through the Head of the Institution as per the instructions of Registrar (Evaluation) from time to time.																																							
18OMC6.14	The candidates who fail to submit the dissertation work within the stipulated time have to submit the same at the time of next ensuing examination.																																							
18OMC7.0	Computation of SGPA and CGPA																																							
18OMC7.1	<p>(i) The University adopts absolute grading system wherein the marks are converted to grades, and every semester results will be declared with semester grade point average (SGPA) and Cumulative Grade Point Average (CGPA). The CGPA will be calculated for every semester, except for the first semester.</p> <p>(ii) The grading system with the letter grades and the assigned range of marks under absolute grading system are as given below:</p>																																							
	<table border="1"> <thead> <tr> <th>Level</th> <th>Outstanding</th> <th>Excellent</th> <th>Very Good</th> <th>Good</th> <th>Above Average</th> <th>Average</th> <th>Fail</th> </tr> </thead> <tbody> <tr> <td>Letter Grade</td> <td>S</td> <td>A</td> <td>B</td> <td>C</td> <td>D</td> <td>E</td> <td>F</td> </tr> <tr> <td>Grade Points</td> <td>10</td> <td>9</td> <td>8</td> <td>7</td> <td>6</td> <td>4</td> <td>00</td> </tr> <tr> <td rowspan="2">Percentage of Marks Scored in a Course</td> <td>≥ 90</td> <td><90 ≥80</td> <td><80 ≥70</td> <td><70 ≥60</td> <td><60 ≥ 55</td> <td><55 ≥50</td> <td><50</td> </tr> <tr> <td>(90 -100)</td> <td>(80 - 89)</td> <td>(70 - 79)</td> <td>(60 - 69)</td> <td>(55 - 59)</td> <td>(50- 54)</td> <td>(0 - 49)</td> </tr> </tbody> </table>	Level	Outstanding	Excellent	Very Good	Good	Above Average	Average	Fail	Letter Grade	S	A	B	C	D	E	F	Grade Points	10	9	8	7	6	4	00	Percentage of Marks Scored in a Course	≥ 90	<90 ≥80	<80 ≥70	<70 ≥60	<60 ≥ 55	<55 ≥50	<50	(90 -100)	(80 - 89)	(70 - 79)	(60 - 69)	(55 - 59)	(50- 54)	(0 - 49)
Level	Outstanding	Excellent	Very Good	Good	Above Average	Average	Fail																																	
Letter Grade	S	A	B	C	D	E	F																																	
Grade Points	10	9	8	7	6	4	00																																	
Percentage of Marks Scored in a Course	≥ 90	<90 ≥80	<80 ≥70	<70 ≥60	<60 ≥ 55	<55 ≥50	<50																																	
	(90 -100)	(80 - 89)	(70 - 79)	(60 - 69)	(55 - 59)	(50- 54)	(0 - 49)																																	

(iii) A student obtaining Grade F in a Course shall be considered failed and is required to reappear in subsequent SEE. Whatever the letter grade secured by the student during his /her reappearance shall be retained. However the number of attempts taken to clear a Course shall be indicated in the grade cards/transcripts.

180MC7.2

Computation of SGPA and CGPA

The following expressions shall be used to compute the Semester Grade Point Average (SGPA) and Cumulative Grade Point Average (CGPA) respectively:

$$SGPA = \frac{(\sum [\text{CourseCredits} \times \text{Grade Points}] \text{ for all the Courses in that Semester})}{(\sum [\text{CourseCredits}] \text{ for all the Courses in that Semester})}$$

$$CGPA = \frac{(\sum [\text{CourseCredits} \times \text{GradePoints}] \text{ for all Courses excluding @ those with F grades until that Semester})}{(\sum [\text{Course Credits}] \text{ for all Courses excluding @ those with F grades until that semester})}$$

(a) SGPA and CGPA Calculations: An Illustrative Example for one academic year

Semester (Odd -I, Even: II)	Course Number	Credits	Grade	Grade Points	Credit Points	SGPA, CGPA
I	XX101	5:0:0 = 5	B	8	5 × 8 = 40	SGPA = $\frac{117}{25}$ = 4.68
I	XX102	3:2:0 = 5	Absent(F)	0	5 × 0 = 00	
I	XX103	3:0:0 = 3	A	9	3 × 9 = 27	
I	XX104	0:1:1 = 2	F	0	2 × 0 = 00	
I	XX105	4:1:0 = 5	D	6	5 × 6 = 30	
I	XX106	5:0:0 = 5	E	4	5 × 4 = 20	
Total		25 (18*)	Total	117		
(18*): Total credits of the semester excluding the credits of the courses under F grade. Considered for the calculation of CGPA of the two consecutive semesters under consideration.						
II	XX107	3:1:1 = 5	C	7	5 × 7 = 35	SGPA = $\frac{157}{25}$ = 6.28
II	XX108	4:0:0 = 4	B	8	4 × 8 = 32	
II	XX109	3:0:0 = 3	D	6	3 × 6 = 18	CGPA = $\frac{(117 + 157)}{18 + 23}$ = $\frac{274}{41}$ = 6.28
II	XX110	4:1:0 = 5	E	4	5 × 4 = 20	
II	XX111	2:1:1 = 4	A	9	4 × 9 = 36	
II	XX112	2:0:0 = 2	F	0	2 × 0 = 00	
II	XX113	0:2:0 = 2	B	8	2 × 8 = 16	
Total		25 (23*)	Total	157		

(23*): Total credits of the semester excluding the credits of the courses under F grade. Considered for the calculation of CGPA of the two consecutive semesters under consideration.

If the Student secures letter grades as detailed below after reappearance to SEE, then the SGPA and CGPA shall be calculated as indicated below.						
I	XX102	3:2:0 = 5	D	6	5 × 6 = 30	SGPA (I Semester) = $\frac{117 + 30 + 14}{25}$ = $\frac{161}{25}$ = 6.44
I	XX104	0:1:1 = 2	C	7	2 × 7 = 14	
II	XX112	2:0:0 = 2	D	6	2 × 6 = 12	SGPA (II Semester) = $\frac{(157 + 12)}{25}$ = $\frac{169}{25}$ = 6.76

CGPA at the end of the academic year after passing all the Courses of the two

consecutive semesters under consideration =

$$\frac{(6.44 \times 25 + 6.76 \times 25)}{50} = \mathbf{6.60}$$

Semester	I	II	III	IV	V	VI
Credits of the semester	21	23	21	23	22	22
SGPA	7.08	6.08	7.92	9.24	8.75	9.60

(b) CGPA Calculation of the Programme: An Illustrative Example

$$\text{Thus CGPA} = \frac{(21 \times 7.08 + 23 \times 6.08 + 21 \times 7.92 + 23 \times 9.24 + 22 \times 8.75 + 22 \times 9.60)}{132} = \mathbf{8.11}$$

180MC7.3

Grade Card: Based on the secured letter grades, grade points, SGPA and CGPA, a grade card for each semester and a consolidated grade card indicating the performance in all semesters shall be issued.

180MC8.0

Conversions of Grades into Percentage and Class Equivalence

180MC8.1

Conversions of Grades into Percentage

Conversion formula for the conversion of CGPA into percentage is given below.

Percentage of marks secured,

	$P = [\text{CGPA Earned} - 0.75] \times 10$ <p>Illustration for a CGPA of 8.20:</p> $P = [\text{CGPA Earned } 8.2 - 0.75] \times 10 = 74.5\%$
18OMC8.2	<p>Class Equivalence:</p> <p>After the conversion of final CGPA into percentage of marks (P), a graduating student is reckoned to have passed in</p> <p>(i) First Class with Distinction (FCD) if $P \geq 70\%$</p> <p>(ii) First Class (FC) if $P \geq 60\%$ but $< 70\%$ and</p> <p>(iii) Second Class (SC) if $P < 60\%$.</p>
18OMC9.0	Continuous Internal Evaluation and Semester End Evaluation
18OMC9.1	<p>There shall be a maximum of 40 CIE Marks in each theory and practical paper.</p> <p>A candidate shall obtain not less than 50% of the maximum marks prescribed for the CIE of each Theory course/ Laboratory/Internship/Project/Dissertation. [To be read along with 18OMC9.8]</p>
18OMC9.2	<p>CIE Marks shall be based on</p> <p>a) Tests (for 20 Marks) and</p> <p>b) Assignments, Quiz, Simulation, Experimentation, Mini project, oral examination, field work etc., (for 20 Marks) conducted in respective courses.</p>
18OMC9.3	<p>The CIE marks in a theory course, for 20 marks, shall be based on two tests covering the entire syllabus. An additional test may be conducted for the needy students to provide an opportunity to improve their CIE Marks before the end of the semester. The CIE marks shall be the average of the marks scored</p>

	in at least two of the above tests.
18OMC9.4	<p>The candidates shall write the Tests in Blue Book/s. The Blue book/s and other documents relating to award of CIE marks under 18OMC9.2(b) shall be preserved by the Principal / Head of the Department for at least six months from the date of announcement of University results and made available for verification at the directions of the Registrar (Evaluation).</p>
18OMC9.5	<p>Every page of the CIE marks list shall bear the signatures of the concerned Teacher, Head of the Department and the Principal.</p>
18OMC9.6	<p>The CIE marks list shall be displayed on the Notice Board and corrections, if any, shall be incorporated before submitting to the University.</p>
18OMC9.7	<p>The CIE marks shall be sent to the university by the Principals well in advance before the commencement of Semester End Examinations. No corrections of the CIE marks shall be entertained after the submission of marks list to the University.</p>
18OMC9.8	<p>Candidates obtaining less than 50% of the CIE marks in any course (Theory/ Laboratory/ Internship/ Project) shall not be eligible to appear for the University examination in that course/s. In such cases, the Head of the Department shall arrange for the improvement of CIE marks in the course/ Laboratory when offered in the subsequent academic year subject to the provision of 18OMC1.3.</p>
18OMC9.9	<p>Semester End Evaluation: There shall be a University examination at the end of each semester.</p> <p>Setting Theory Question Papers and Evaluation: Question papers in theory courses shall be set by the</p>

	Examiners appointed by the University. SEE will be conducted for 100 marks & shall be scaled down to 60 marks for calculation of end exam CGPA result.
18OMC9.10	There shall be double valuation of theory papers. The theory Answer booklets shall be valued independently by two examiners appointed by the University.
18OMC9.11	If the difference between the marks awarded by the two Examiners is not more than 15 per cent of the maximum marks, the marks awarded to the candidate shall be the average of two evaluations.
18OMC9.12	If the difference between the marks awarded by the two Examiners is more than 15 per cent of the maximum marks, the answer booklet shall be evaluated by a third Examiner appointed by the university. The average of the marks of nearest two valuations shall be considered as the marks secured by the candidate. In case, if one of the three marks falls exactly midway between the other two, then the highest two marks shall be taken for averaging.
18OMC10.0	Eligibility for Passing and Award of Degree
18OMC10.1	<p>(1) A student who obtains any grade S to E shall be considered as passed and if a student secures F grade in any of the head of passing he/she has to reappear in that head for the SEE.</p> <p>(2) A student shall be declared successful at the end of the Programme for the award of Degree only on obtaining CGPA 5.00, with none of the Courses remaining with F Grade.</p> <p>(3) In case, the SGPA/ CGPA fall below 5.00, the student shall be permitted to appear again for SEE for required number of courses (other than Practical) and times, subject to the provision of 18OMC1.3, to</p>

	make up SGPA/CGPA greater than or equal to 5.0. The student should reject the SEE results of the previous attempt and obtain written permission from The Registrar (Evaluation) to reappear in the subsequent SEE.
18OMC10.2	For a pass in a theory and laboratory(practical) course, the student shall secure minimum of 40 % of the maximum marks prescribed in the Semester End Examination and 50 % of marks in CIE and 50 % in the aggregate of CIE and SEE marks. The Minimum Passing Grade in a course is E.
18OMC10.3	For a pass in Internship/ Seminar/ Project/ Dissertation/ Viva-voce examination, a student shall secure a minimum of 50% of the maximum marks prescribed for the SEE in Internship/ Seminar/ Project/ Dissertation/Viva-voce. The Minimum Passing Grade in a course is E.
18OMC10.4	VI semester candidates having backlog courses are permitted to upload the dissertation report and to appear for SEE. The VI semester grade card shall be released only when the candidate completes all the backlog courses and become eligible for the award of degree.
18OMC10.5	<p>A candidate may at his/her desire reject his/her latest semester, except the VI semester, results of University examination in respect to all courses of that semester. Rejection shall be permitted only once during the entire Programme. The CIE marks of the rejected semester shall remain the same.</p> <p>Rejection of results of the University examination including CIE marks is not permitted.</p>
18OMC10.6	If the rejection of the University examination results of the semester happens to be of an odd semester, the candidate can take admission to the immediate next


	even semester. However, if the rejection of the University result is of even semester, the candidate cannot take admission to the next odd semester.
18OMC10.7	Application for rejection shall be submitted to the Registrar (Evaluation) through the Principal of the college, within thirty days from the date of announcement of results.
18OMC10.8	A candidate, who opts for rejection of results of a semester shall be eligible for the award of class and distinction, but shall not be eligible for the award of rank.
18OMC10.9	Eligibility for Award of Degree: A student shall be declared to have completed the degree of Master of Computer Applications, provided the student has undergone the stipulated course work as per the regulations and has earned the prescribed Credits, as per the Scheme of Teaching and Examination, of the programme.
18OMC11.0	Attendance Requirement
18OMC11.1	Registration and Enrollment: i) Except for the first semester, registration for a semester will be done during a specified week before the semester end examination of the previous semester. ii) The registration sheet should have the Candidate details, course name and code, number of credits and category (core/elective/audit) for each course of that semester. iii) The Faculty Adviser, assigned by the Head of the Department, will counsel the students in planning their courses of study and provide guidance, motivation, emotional support, and enable the

	mentees to reach the desired professional and careergoals.
18OMC11.2	Courses of each semester shall be treated as a separate unit for calculation of the attendance.
18OMC11.3	The candidate has to put in a minimum attendance of 85% in each course with a provision to condone 10% of the attendance by the Vice-Chancellor on the specific recommendation of the Principal of the college where the candidate is studying, based on medical grounds, participation in University/ State/ National/ International level sports and cultural activities, seminars, workshops, paper presentation etc., of significant value. The necessary documents in support are to be submitted along with recommendations to condone the shortage.
18OMC11.4	In case of late admission, approved by competent authority (Karnataka Examination Authority/VTU), to I/III semester of the programme the attendance shall be reckoned from the date of admission to the programme.
18OMC11.5	A candidate, who does not satisfy the attendance requirement (in one or more Courses) as mentioned in 18OMC11.3 shall not be eligible to appear for the SEE of that semester and shall not be permitted to take admission to next higher semester. The candidate shall be required to repeat that semester during the subsequent academic year.
18OMC11.6	Principals of the concerned colleges shall notify regularly, the list of candidates who fall short of attendance.
18OMC11.7	The list of the candidates falling short of attendance shall be sent to the University at least one week prior to the commencement of the examination.

18OMC12.0	Promotion and Eligibility
18OMC12.1	<p>Promotion:</p> <p>There shall be no restriction for promotion from an odd semester to the next even semester, provided the student has fulfilled the attendance requirement.</p>
18OMC12.2	<p>A candidate shall be eligible for promotion from an even semester to the odd semester (i.e. to the next academic year) if the candidate has not failed in more than four heads of passing of the immediately preceding two semesters put together and has passed in all the courses of all the lower semester examinations. A theory or practical shall be treated as a head of passing.</p> <p>Illustrations:</p> <p>Case: 1 Candidates admitted to I year/I semester of the programme.</p> <p>a) A candidate seeking eligibility to III semester should not have failed in more than 4 heads of passing of first and second semesters taken together.</p> <p>b) A candidate seeking eligibility to V semester should have passed in all the courses of I and II semesters and should not have failed in more than 4 heads of passing of III and IV semesters taken together</p> <p>Case: 2 Candidates admitted to II year/III semester of the programme under lateral entry scheme.</p> <p>A student seeking eligibility to V semester should not have failed in more than 4 heads of passing of III and IV semesters considered together.</p>
18OMC12.3	The Mandatory non – credit courses, if any, shall not be considered for the Eligibility criterion prescribed

	for promotion, award of Class, calculation of SGPA and CGPA. However, a pass in the above courses is mandatory before the completion of Degree.
18OMC13.0	Temporary Discontinuation/Break in the Program
18OMC13.1	<p>(a) If a candidate, for any reason, temporarily discontinues the Programme or take a break from programme during any semester, he/she may be permitted to continue in the programme by registering to the same semester of the prevailing scheme. The candidate shall complete all the remaining course work subject to the provision</p> <p>18OMC13.2 Also the Candidates may have to complete additional course/s, if any, as per the decision of concerned Board of Studies and approval of Dean, Faculty of Engineering, on establishing equivalence between two schemes. A Grade card shall be issued to that effect. Additional course/s shall not be considered for the eligibility criterion prescribed for promotion. However, based on the individual cases, it is considered to decide the SGPA and CGPA to admit the student for the award of degree. Such candidate shall not be eligible for the award of rank.</p> <p>(b) Candidates who takes admission to any semester of the existing scheme from another scheme, as a repeater/fresher because of various reasons have to complete additional course/s, if any, as per the decision of concerned Board of Studies and approval of Dean, Faculty of Engineering, on establishing equivalence between two schemes. A Grade card shall be issued to that effect. Additional course/s shall not be considered for the eligibility criterion prescribed for promotion. However, based</p>

	on the individual cases, it is considered to decide the SGPA and CGPA to admit the student for the award of degree. Such candidate shall not be eligible for the award of rank.
18OMC14.0	Award of Prizes, Medals and Ranks
18OMC14.1	For the award of Prizes and Medals, the conditions stipulated by the Donor shall be considered subject to the provisions of the statutes framed by the University for such awards.
18OMC14.2	<p>(1) For award of rank in Master of Computer Applications, the CGPA secured by the student on completion of the programme is considered.</p> <p>(2) A student shall be eligible for a rank at the time of award of degree of Master of Computer Applications provided the student</p> <p>a) Is not a repeater in any semester</p> <p>b) Has not rejected the results of any semester.</p> <p>c) Has passed I to VI semester in all the courses in first attempt only</p> <p>(3) The total number of ranks awarded shall be 10% of total number of students appeared in VI semester subject to a maximum of 10 ranks.</p> <p>Illustration:</p> <p>a) If 150 students appeared for the VI semester, the number of ranks to be declared will be 10.</p> <p>b) If 84 students appeared for the VI semester, the number of ranks to be declared will be 08.</p> <p>(c) In case of fractional number of ranks, it is rounded to higher integer only when the first decimal place value is greater than or equal to 5.</p> <p>Ranks are awarded based on the merit of the students</p>

	as determined by CGPA. If two or more students get the same CGPA, the tie shall be resolved by considering the number of times a student has obtained higher SGPA. If it is not resolved even at this stage, the number of times a student has obtained higher grades like S, A, B etc., shall be taken into account to decide the order of the rank.
18OMC15.0	Applicability and Power to Modify
18OMC15.1	The regulations governing the Degree of Master of Computer Applications of Visvesvaraya Technological University shall be binding on all concerned.
18OMC15.2	<p>i) Notwithstanding anything contained in the foregoing, the University shall have the power to issue directions/ orders to address any difficulty.</p> <p>ii) Nothing in the foregoing may be construed as limiting the power of the University to amend, modify or repeal any or all of the above.</p> <p style="text-align: center;"></p>

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI
Under Outcome Based Education (OBE) AndChoice Based Credit System (CBCS) Scheme
Scheme of Teaching and Examination – 2018-19
Master of Computer Applications

I SEMESTER

Sl. No	Course Code	Teaching Department	Title	Teaching Hours /Week		Practical (P)	Duration in hours	Examination		Credits	
				Theory (L)	Tutorial (T)			SEE Marks	CIE Marks		
1	18MCA11	MCA	Object Oriented Programming with C++	03	-	-	03	60	40	03	
2	18MCA12	MCA	Unix and Shell Programming	03	-	-	03	60	40	03	
3	18MCA13	MCA	Web Technologies	03	-	-	03	60	40	03	
4	18MCA14	MCA	Software Engineering	03	-	-	03	60	40	03	
5	18MCA15	MCA	Computer Organization	03	-	-	03	60	40	03	
6	18MCA16	MCA	C++ Programming Lab	-	01	03	03	60	40	02	
7	18MCA17	MCA	Unix and Shell Programming Lab	-	01	03	03	60	40	02	
8	18MCA18	MCA	Web Technologies Lab	-	01	03	03	60	40	02	
TOTAL				Theory: 15hours Practical: 09 hours Tutorial/Instruction: 03hours				480	320	800	21

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI
Under Outcome Based Education (OBE) AndChoice Based Credit System (CBCS) Scheme
Scheme of Teaching and Examination – 2018-19
Master of Computer Applications

II SEMESTER

Sl. No	Course Code	Teaching Department	Title	Teaching Hours /Week		Practical (P)	Duration in hours	Examination		Credits	
				Theory (L)	Tutorial (T)			SEE Marks	CIE Marks		
1	18MCA21	MCA	Programming using Java	03	-	-	03	60	40	03	
2	18MCA22	MCA	Data Structures using C++	03	-	-	03	60	40	03	
3	18MCA23	MCA/ Mathematics	Discrete Mathematical Structures and Statistics	03	-	-	03	60	40	03	
4	18MCA24	MCA	Computer Networks	03	-	-	03	60	40	03	
5	18MCA25	MCA	Operating Systems	03	-	-	03	60	40	03	
6	18MCA26	MCA	Java Programming Lab	-	01	03	03	60	40	02	
7	18MCA27	MCA	Data Structures Lab	-	01	03	03	60	40	02	
8	18MCA28	MCA	Computer Networks Lab	-	01	03	03	60	40	02	
9	18MCA29	MCA	Mini Project	-	02	02	03	60	40	02	
TOTAL				Theory: 15hours Practical: 11 hours Tutorial/Instruction: 05 hours				540	360	900	23

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI
Under Outcome Based Education (OBE) And Choice Based Credit System (CBCS) Scheme
Scheme of Teaching and Examination – 2018-19
Master of Computer Applications

III SEMESTER

Sl. No	Course Code	Title	Teaching Hours /Week			Duration in hours	Examination			Credits		
			Teaching Department	Theory (L)	Tutorial (T)		Practical (P)	SEE Marks	CIE Marks		Total Marks	
1	18MCA31	Database Management System	MCA	03	-	-	03	60	40	100	03	
2	18MCA32	Programming using Python	MCA	03	-	-	03	60	40	100	03	
3	18MCA33	Design and Analysis of Algorithms	MCA	03	-	-	03	60	40	100	03	
4	18MCA34	System Software	MCA	03	-	-	03	60	40	100	03	
5	18MCA35X	Elective I	MCA	03	-	-	03	60	40	100	03	
6	18MCA36	DBMS Lab	MCA	-	01	03	03	60	40	100	02	
7	18MCA37	Python Programming Lab	MCA	-	01	03	03	60	40	100	02	
8	18MCA38	Algorithms Lab	MCA	-	01	03	03	60	40	100	02	
TOTAL								Theory: 15hours Practical: 09 hours Tutorial/Instruction: 03hours	480	320	800	21

Elective I	
18MCA351	Software Testing
18MCA352	Optimization Techniques
18MCA353	Advance Computer Networks
18MCA354	Management Information Systems

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI
Under Outcome Based Education (OBE) And Choice Based Credit System (CBCS) Scheme
Scheme of Teaching and Examination – 2018-19
Master of Computer Applications

IV SEMESTER

Sl. No	Course Code	Title	Teaching Hours /Week			Duration in hours	Examination			Credits		
			Teaching Department	Theory (L)	Tutorial (T)		Practical (P)	SEE Marks	CIE Marks		Total Marks	
1	18MCA41	Advanced Java Programming	MCA	03	-	-	03	60	40	100	03	
2	18MCA42	Advanced Web Programming	MCA	03	-	-	03	60	40	100	03	
3	18MCA43	Object Oriented Modeling and Design	MCA	03	-	-	03	60	40	100	03	
4	18MCA44X	Elective II	MCA	03	-	-	03	60	40	100	03	
5	18MCA45X	Elective III	MCA	03	-	-	03	60	40	100	03	
6	18MCA46	Professional Communication & Report Writing	MCA	02	-	-	03	60	40	100	02	
7	18MCA47	Advanced Java Programming Lab	MCA	-	01	03	03	60	40	100	02	
8	18MCA48	Advanced Web Programming Lab	MCA	-	01	03	03	60	40	100	02	
9	18MCA49	Object Oriented Modeling and Design Lab	MCA	-	01	03	03	60	40	100	02	
TOTAL								Theory: 17hours Practical: 09 hours Tutorial/Instruction: 03hours	540	360	900	23

Elective II		Elective III	
18MCA441	Wireless Communication and Mobile Technologies	18MCA451	Enterprise Resource Planning (ERP)
18MCA442	Cyber Security	18MCA452	Data Warehousing and Data Mining
18MCA443	Computer Graphics	18MCA453	Advanced Database Management Systems
18MCA444	Cloud Computing	18MCA454	Big Data Analytics

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI
Under Outcome Based Education (OBE) AndChoice Based Credit System (CBCS) Scheme
Scheme of Teaching and Examination – 2018-19
Master of Computer Applications

V SEMESTER

Sl. No	Course Code	Title	Teaching Department	Teaching Hours/Week			Duration in hours	Examination			Credits
				Theory (L)	Tutorial (T)	Practical (P)		SEE Marks	CIE Marks	Total Marks	
1	18MCA51	Programming using C#.net	MCA	03	-	-	03	60	40	100	03
2	18MCA52	Mobile Applications	MCA	03	-	-	03	60	40	100	03
3	18MCA53	Machine Learning	MCA	03	-	-	03	60	40	100	03
4	18MCA54X	Elective IV	MCA	03	-	-	03	60	40	100	03
5	18MCA55X	Elective V	MCA	03	-	-	03	60	40	100	03
6	18MCA56	C#.net Lab	MCA	-	01	03	03	60	40	100	02
7	18MCA57	Mobile Applications Lab	MCA	-	01	03	03	60	40	100	02
8	18MCA58	Mini Project	MCA	-	02	04	03	60	40	100	03
TOTAL				Theory: 15hours Practical: 10 hours Tutorial/Instruction: 04 hours				480	320	800	22

Elective IV		Elective V	
18MCA541	Multimedia Systems	18MCA551	System Simulation and Modeling
18MCA542	Internet of Things (IoT)	18MCA552	Principles of User Interface Design
18MCA543	Image Processing	18MCA553	Software Architecture
18MCA544	Parallel Computing	18MCA554	Storage Area Networks

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI
Under Outcome Based Education (OBE) AndChoice Based Credit System (CBCS) Scheme
Scheme of Teaching and Examination – 2018-19
Master of Computer Applications

VI SEMESTER

Sl. No	Course Code	Title	Teaching Department	Teaching Hours/Week			Duration in hours	Examination			Credits
				Theory	Practical	Tutorial		SEE Marks	CIE Marks	Total Marks	
1	18MCA61	Internship	MCA	-	-	-	-	-	100	100	4
2	18MCA62	Seminar	MCA	-	-	-	-	-	100	100	2
3	18MCA63	Major Project	MCA	-	-	-	03	60	40	100	16
TOTAL								60	240	300	22

OBJECT ORIENTED PROGRAMMING WITH C++ SEMESTER – I

Semester	I	CIE Marks : 40
Course Code	18MCA11	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA11) will enable student to

- Develop an in-depth understanding of functional, logic, and object-oriented programming paradigms
- Compare Modular programming with Object Oriented Programming and analyze the strengths and weaknesses of Object Oriented Programming.
- Develop skills to write program in C++ using Classes and Objects and implement Encapsulation, Polymorphism and Inheritance.
- Handle run-time errors in a program and deal with files and input/output streams in C++.

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Acquire knowledge on C++ programming concepts.

CO2: Analyze the different concepts of C++.

CO3: Design and Develop the solution to a problem using Object Oriented Programming Concepts.

CO4: Apply the learning into real world problems independently.

MODULE -1:

Introduction to OOPs: What Is Object-Oriented Programming? , Encapsulation, Polymorphism, Inheritance.

C++ Overview: The Origins of C++,The General Form of a C++

Program, different data types, operators, expressions, arrays and strings, reference variables. Function Components, argument passing, inline functions, function overloading, function templates.

MODULE -2:

Classes & Objects: Introduction, Class Specification, Class Objects, access members, defining member functions, data hiding, constructors, destructors, parameterized constructors, static data members, static member functions, scope resolution operator, Passing Objects to Functions, Returning Objects, Object Assignment.

Pointers and dynamic memory allocation: Pointers, Pointer as function arguments, Dynamic Allocation Operators new and delete, Initializing Allocated Memory, Allocating Arrays, Allocating Objects

MODULE -3:

Operator overloading: Operator overloading as member functions and using friend functions. Overloading of binary operators like +, -, *.Creating Prefix and Postfix forms of ++, -- Operators, Operator Overloading Restrictions, Operator Overloading Using a Friend Function to Overload ++ or --, Overloading ().

Inheritance: Base Class, Inheritance & protected members, protected base class inheritance, inheriting multiple base classes, Constructors, Destructors & Inheritance. Passing parameters to base Class Constructors, Granting access, Virtual base classes.

MODULE -4:

Virtual Functions and Runtime Polymorphism: Virtual function - Calling a Virtual function through a base class reference, Virtual attribute is inherited, Virtual functions are hierarchical, pure virtual functions, abstract classes, using Virtual functions, Early & late binding.

Standard C++ I/O Classes: Old vs. Modern C++ I/O, C++ Streams, The C++ Stream Classes, C++'s Predefined Streams, Formatted I/O, Formatting Using the ios Members, Setting the Format Flags, Clearing Format Flags, Overloading << and >>, manipulators

MODULE-5:

Exception Handling: Exception Handling, Fundamentals, Catching Class Types, Using Multiple catch Statements, Handling Derived-Class Exceptions, Exception Handling Options, Catching All Exceptions, Restricting Exceptions, Rethrowing an Exception, Understanding terminate() and unexpected(), uncaught_exception() Function, The exception and bad_exception Classes, Applying Exception Handling.

STL: Class template, An overview of STL, containers, vectors

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Textbooks:

1. Object oriented programming with C++, E. Balaguruswamy, Tata McGraw Hill.
2. Herbert Schildt: C++ The Complete Reference, 4th Edition, Tata McGraw Hill, 2014.
3. K R Venugopal, Rajkumar Buyya, T Ravishanker: Mastering C++, Tata McGraw Hill.

Reference Books:

1. Stanley B. Lippmann, Josee Lajore: C++Primer, 4th Edition, Addison Wesley.
2. Object oriented programming with C++, E. Balaguruswamy, Tata McGraw Hill.
3. Programming in ANSI C, Balaguruswamy, McGraw Hill Education
4. The C Programming Language, Brian W Kernighan, Dennis M Ritchie, PHI, 2nd Edition

UNIX AND SHELL PROGRAMMING

Semester	I	CIE Marks : 40
Course Code	18MCA12	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA12) will enable student to

- Provide introduction to UNIX Operating System and its File System
- Gain an understanding of important aspects related to the SHELL and the process
- Develop the ability to formulate regular expressions and use them for pattern matching.
- Provide a comprehensive introduction to SHELL programming, services and utilities

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

CO1: Understand and experience the UNIX environment, File system and hierarchy.

CO2: Understand the mechanism of process creation and process management using commands.

CO3: Explore special privileged commands of system administration.

CO4: Analyze the usage of different shell commands, variables and AWK filtering.

CO5: Use Unix commands and language constructs in building shell scripts.

MODULE-1:

Introduction of UNIX and Shell:

Introduction, History, Architecture, Experience the Unix environment, Basic commands, cat, cal, date, calendar, who, printf, tty, stty, uname, passwd, echo, tput, bc, script, spell and ispell, Introduction to Shell Scripting, Shell Scripts, read, Command Line Arguments, Exit Status of a Command, The Logical Operators && and ||, exit, if, and case conditions, expr, sleep and wait, while, until, for, \$, @, redirection, set and shift, trap.

MODULE-2:

UNIX File System:

The file, what's in a filename? The parent-child relationship, pwd, the Home directory, absolute path names, using absolute pathnames for a command, cd, mkdir, rmdir, Relative path names, The UNIX file system. Basic File Attributes: ls -l, the -d option, File Permissions, chmod, Security and File Permission, users and groups, security level, changing permission, changing ownership and group, File Attributes, More file attributes: hard link, symbolic link, umask, find.

MODULE-3:

Simple Filters:

Pr, head, tail, cut, paste, sort, uniq, tr commands, Filters using Regular Expression: grep, Regular Expression, egrep, fgrep, sed instruction, Line Addressing, Inserting and Changing Text, Context addressing, writing selected lines to a file, the -f option, Substitution, Properties of Regular Expressions.

MODULE-4:

Awk and Advanced Shell Programming:

Awk-Advanced Filters: Simple awk Filtering, Splitting a Line into Fields, printf, the Logical and Relational Operators, Number Processing, Variables, The -f option, BEGIN and END positional Parameters, getline, Built-in variables, Arrays, Functions, Interface with the Shell, Control Flow. The sh command, export, the Command, Conditional Parameter Substitution, Merging Streams, Shell Functions, eval, Exec Statement and Examples

MODULE-5:

Process and System Administration:

Process basics, PS, internal and external commands, running jobs in background, nice, at and batch, cron, time commands, Essential System Administration root, administrator's privileges, startup & shutdown, managing disk space, cpio, tar, Customizing the Environment : System Variables, profile, stty, Aliases, Command History, On-line Command Editing.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbook:

1. Your UNIX- The Ultimate Guide, Sumitabha Das, Tata Mc Graw Hill.

Reference Books:

1. “Unix Shell Programming”, Yashwant Kanetkar.
2. “Beginning Shell Scripting”, Eric Foster-Johnson, John C Welch, Micah Anderson, Wrox publication.
3. UNIX: Concepts and Applications, Sumitabha Das, Tata Mc Graw Hill,
4. “Introduction to UNIX” by M. G. Venkatesh Murthy.

WEB TECHNOLOGIES

Semester	I	CIE Marks : 40
Course Code	18MCA13	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA13) will enable student to

- Provide a clear understanding of fundamentals of Web Technologies.
- Deliver the importance of XHTML, HTML5, CSS, JavaScript, JQuery and XML in developing web applications.

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Understand the fundamentals of web and thereby develop web applications using various web development languages and tools.

CO2: Build the ability to select the essential technology needed to develop and implement web Applications.

CO3: Use JavaScript and jQuery to develop dynamic and interactive web page.

CO4: Write a well formed / valid XML document.

CO5: Design XML document with presentation using CSS.

MODULE -1:

Web Fundamentals

Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security, the Web Programmers Toolbox.

Introduction to XHTML

Basic syntax, Standard structure, Basic text markup, Images, Hypertext Links. Lists, Tables, Forms, Frames, Syntactic differences between HTML and XHTML.

MODULE -2:

Introduction to HTML5

New features of HTML5, HTML5 DocType, HTML5 Structure, Tags- nav, section, article, aside, header, footer, HTML5 Form Elements- Search, tel, url, email, number and range, HTML5 Media tags- Audio and video.

Cascading Style Sheets

Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The box model, Background images, The and <div> tags, Conflict resolution.

MODULE -3:

The basics of JavaScript

Overview of JavaScript, Object orientation and JavaScript, general Syntactic characteristics, Primitives, operations, and expressions, Screen output and keyboard input, Control statements, Object creation and modification, Arrays, Functions, Constructors, Pattern matching using regular expressions, Errors in scripts.

JavaScript and XHTML Documents

The JavaScript Execution Environment, The Document Object Model, Elements Access in Java Script, Events and Event Handling, Handling Events from Body Elements, Handling Events from Text Box and password Elements, The DOM2 Model, The navigator Object, Dom Tree Traversal and Modification.

MODULE -4:

Dynamic Documents with JavaScript

Introduction, Positioning Elements, Moving Elements, Element Visibility, Changing Colors and Fonts, Dynamic Content, Stacking Elements, Locating the Mouse Cursor, Reacting to a Mouse Click, Slow Movement of Elements, Dragging and Dropping Elements

Introduction to XML

Introduction, Syntax of XML, XML Document Structure, Document type definitions, Namespaces, XML schemas, displaying raw XML documents, Displaying XML documents with CSS, Web services.

MODULE -5:

Introduction to jQuery

Introducing jQuery, jQuery fundamentals, Creating the wrapped element set, Bringing pages to life with jQuery, Understanding the browser event models, The jQuery Event Model, Sprucing up with animations and effects.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Robert W. Sebesta, "Programming the World Wide Web", 4th Edition, Pearson education, 2012.
2. Kogent Learning solutions Inc., "HTML 5 Black Book: Covers CSS3, JavaScript, XML, XHTML AJAX, PHP & JQuery: Black Book", Dreamtech Press, 2011.
3. Bear Bibeault, Yehuda Katz: jQuery in Action. 3rd Edition, DreamTech India, 2008.

Reference Books:

1. Randy Connolly, Ricardo Hoar, "Fundamentals of Web Development", 2nd Edition, Pearson, 2018.
2. Jeffrey C. Jackson: Web Technologies-A Computer Science Perspective, Pearson Education, 7th Impression, 2012.
3. Chris Bates: Web Programming Building Internet Applications, 3rd Edition, WileyIndia, 2009.
4. Zak Ruvalcaba Anne Boehm, "Murach's HTML5 and CSS3", 3rd Edition, Murachs/Shroff Publishers & Distributors Pvt Ltd, 2016.

SOFTWARE ENGINEERING

Semester	I	CIE Marks : 40
Course Code	18MCA14	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA14) will enable student to

- Provide a clear understanding of Software Engineering, Software Product Development Phases and its applications in real world.
- Expose to recent trends in Software Engineering like extreme programming and evolutionary methods, Use of Component Based Software Engineering.
- Apply testing techniques, viz. black box and white box testing, testing tools and methodology and analyze modeling techniques.

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Categorize problems based on their characteristics and practical importance.

CO2: Apply the correct process models for software development.

CO3: Apply the techniques, skills, and modern engineering tools necessary for engineering practice.

CO4: Define, formulate and analyze a problem as per the testing techniques.

CO5: Apply new Generation of Software Engineering Technology to Meet Current and Future Industrial Challenges of Emerging Software Trends.

MODULE -1:

Overview

Introduction: Professional Software Development Attributes of good software, software engineering diversity, IEEE/ ACM code of software engineering ethics, case studies

Software Process & Agile Software Development

Software Process models: waterfall, incremental development,

reuses oriented, Process activities; Coping with change, The rational Unified process. Agile methods, Plan-driven and agile Development, Extreme Programming, Agile project management, Scaling agile methods.

MODULE -2:

Requirements Engineering

Functional and non-functional requirements, The software requirements document, Requirements specification, Requirements engineering processes, Requirement elicitation and analysis, Requirements validation, Requirements management

Component-based software engineering

Components and component model, CBSE process, Component composition.

MODULE -3:

System Modeling, Architectural Design

Context models, Interaction models, Structural models, Behavioral models, Model-driven engineering, Software architecture: the role of software architecture, architectural views, component and connector view, Architectural styles for C&C view, Documenting architectural design

MODULE -4:

Design and implementation

Design: Design concepts, Function oriented design, detailed design, verification, matrix (Complexity matrix for function oriented design)

Distributed Software engineering

Distributed system issues, Client-server computing, Architectural patterns for distributed systems, Software as a service.

MODULE -5

Planning a software Project

Process planning, Effort estimation, Project scheduling and staffing, Software configuration management plan, Quality plan, Risk Management, Project monitoring plan.

Software Testing

Testing fundamentals, Black-box testing, White-box testing, Testing process

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Ian Sommerville : Software Engineering, 9th edition, Person Education Ltd, 2011. (Chapters:- 1, 2, 3, 4, 5, 17, 18)
2. Pankaj Jalote: Software Engineering, Wiley India Pvt Ltd (2010) (Chapters:-4, 6.1, 6.2, 6.5, 6.6)

Reference Books:

1. Roger S Pressman: Software Engineering-A Practitioners approach, 6th Edition, McGraw-Hill, 2010.
2. Hans Van Vliet: Software Engineering Principles and Practices, 3rd Edition, Wiley-India, 2010

COMPUTER ORGANIZATION

Semester	I	CIE Marks : 40
Course Code	18MCA15	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA15) will enable student to

- Provide a clear understanding of the basic structure and operation of a digital computer.
- Study different number system representation and conversion from one number system to another and Boolean Algebra.
- Learn the working of flip-flops, logical gates, multiplexers, and adders.
- Study the memory system cache memories and virtual memory.
- Explain the different ways of communicating with I/O devices and standard I/O interfaces.

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Understand the basic of Digital Systems

CO2: Realize the concept of Computer System Organization

CO3: Apply the concepts of Input/output Organization and Memory System

CO4: Analyze the Performance of Memory System and Memory Management

CO5: Analyze and Implementation of Stack Organization

MODULE 1:

Binary Systems and Combinational Logic

Digital Computers and Digital Systems, Binary Numbers, Number Base Conversion, Octal and Hexadecimal Numbers, subtraction using r's and r-1 complements, Binary Code, Binary Storage and Registers, Binary Logic, Integrated Circuits. Axiomatic Definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra, Boolean Functions, Canonical and Standard Forms, Other Logic Operations, Digital Logic Gates, The map Method, Two – and Three – Variable Maps, Four – Variables Map .

MODULE 2:

Arithmetic Circuits and Sequential Logic

NAND and NOR Implementation, Other Two- Level Implementations, Don't Care Conditions. Introduction, Adders, Subtractors, Binary Parallel Adder, Decimal Adder, Magnitude Comparator, Decoders, Multiplexers, BOOTH algorithm for signed numbers with example. Sequential Logic: Introduction, different types of Flip–Flops, Triggering of Flip- Flops.

MODULE 3:

Computer Types, Functional Units, Basic Operational Concepts, Bus structure, Software, Performance, Multiprocessing and Multi computers, Machine Instruction: Memory Locations and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes.

MODULE 4:

Assembly Language and Input/Output Organization

Basics of Assembly Language Programme, Examples from Assembly Language Programming. Accessing I/O Devices, Interrupts, Enabling & Disabling Interrupts, Handling Multiple devices, Controlling I/P O/P device behavior, Exceptions, DMA, Buses.

MODULE 5:

The Memory System

Basic Concepts, Semiconductor RAM Memories, Internal organization of memory chips, static memories, dynamic RAM, Synchronous D-RAM, Structure of larger Memories. Read – Only Memories, Speed, Size, and Cost, Cache Memories, Virtual Memories, Memory Management Requirements.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.

- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. M.Morris Mano, “Digital Logic and Computer Design”, Pearson, 2012.
2. Carl Hamacher, Zvonko Vranesic Safwat Zaky, ”Computer Organization”, 5th edition, TataMcGraw-Hill, 2011

Reference Books:

1. John P. Hayes, “Computer Architecture and Organization”, Tata McGraw-Hill, Edition, 2012.
2. Soumitra Kumar Mandal, “Digital Electronics Principles and Applications”, Tata McGraw-Hill, 2010
3. Hamacher , “ Computer Organization” , McGraw-Hill Education

C++ PROGRAMMING LAB

Semester	I	CIE Marks : 40
Course Code	18MCA16	SEE Marks : 60
Contact Periods (L:T:P)	0-1-3	Exam Hours : 03
Credits : 02		

Course learning objectives:

This course (18MCA16) will enable student to

- Provide knowledge of C++ Editors and writing programs using Classes and Objects.
- Introduce concepts like operator overloading, function overloading, function templates and class templates and implement them using C++.
- Implement the three traits of Object Oriented Programming, namely, Encapsulation, Inheritance and Polymorphism using C++.
- Inculcate the knowledge of handling Input/output Streams, Exceptions and Files in C++.

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Apply and implement major programming and object oriented concepts like function overloading, operator overloading, Encapsulations, and inheritance, message passing to solve real-world problems.

CO2: Use major C++ features such as Virtual functions, Templates for data type independent designs and File I/O to deal with large data sets.

CO3: Analyze, design and develop solutions to real-world problems applying OOP Concepts of C++.

Laboratory Experiments:

PART-A

1. Write a C++ program to find the sum for the given variables using function with default arguments.
2. Write a C++ program to swap the values of two variables and demonstrates a function using call by value.
3. Write a C++ program to swap the values of two variables and demonstrates a function using Call by reference using

reference type (&).

4. Write a C++ program to swap the values of two variables and demonstrates a function using Call by reference using pointer (*).
5. Write a C++ program to swap the values of two dynamically allocated variables and release the memory after swapping. (use new & delete operators)
6. Write a program to find the largest, smallest & second largest of three numbers. (use inline function MAX and MIN to find largest & smallest of 2 numbers)
7. Write a program to calculate the volume of different geometric shapes like cube, cylinder and sphere and hence implement the concept of Function Overloading.
8. Write a C++ program to create a template function for Bubble Sort and demonstrate sorting of integers and doubles.

PART-B

1. Define a STUDENT class with USN, Name, and Marks in 3 tests of a subject. Declare an array of 10 STUDENT objects. Using appropriate functions, find the average of the two better marks for each student. Print the USN, Name and the average marks of all the students.
2. Write a C++ program to create a class called COMPLEX and implement the following overloading functions ADD that return a complex number:
(i) ADD (a, s2) – where ‘a’ is an integer (real part) and s2 is a complex number
(ii) ADD (s1, s2) – where s1 and s2 are complex numbers.
3. Friend functions and friend classes:
 - a) Write a program to define class name HUSBAND and WIFE that holds the income respectively. Calculate and display the total income of a family using Friend function.
 - b) Write a program to accept the student detail such as name and 3 different marks by get_data() method and display the name and average of marks using display() method. Define a friend class for calculating the average of marks using the method mark_avg().
4. Create a class called MATRIX using two-dimensional array of

integers. Implement the following operations by overloading the operator == which checks the compatibility of two matrices to be added and subtracted. Perform the addition and subtraction by overloading the + and – operators respectively. Display the results by overloading the operator <<. If (m1==m2) then m3 = m1+m2 and m4 = m1- m2 else display error.

5. Write a program to create an HUMAN class with features as number of Head, Legs, Hands.(NOTE: Number of Head, Legs and Hands are of integer types)
 - a. Create an object HUMAN1 using default constructor. (Default features to have 1 Head, 2 Legs and 2 Hands)
 - b. Create an object HUMAN2 with customized inputs using Parameterized Constructor
 - c. Create an object HUMAN3 using existing object HUMAN1 (Copy Constructor).
 - d. All Humans die after their lifetime. (Destructor)
6. Demonstrate Simple Inheritance concept by creating a base class FATHER with data members SurName and BankBalance and creating a derived class SON, which inherits SurName and BankBalance feature from base class but provides its own feature FirstName and DOB. Create and initialize F1 and S1 objects with appropriate constructors and display the Father & Son details. (Hint : While creating S1 object, call Father base class parameterized constructor through derived class by sending values)
7. Create an abstract base class EMPLOYEE with data members: Name, EmpID and BasicSal and a pure virtual function Cal_Sal(). Create two derived classes MANAGER (with data members: DA and HRA and SALESMAN (with data members: DA, HRA and TA). Write appropriate constructors and member functions to initialize the data, read and write the data and to calculate the net salary. The main() function should create array of base class pointers/references to invoke overridden functions and hence it implements run-time polymorphism.
8. Write a program to implement FILE I/O operations on characters. I/O operations includes inputting a string,

Calculating length of the string, Storing the string in a file, fetching the stored characters from it, etc.

9. Write a program to implement Exception Handling with minimum 5 exceptions Classes including two built-in exceptions.
10.
 - a. Write a program to concatenate 2 strings using STL String class functions.
 - b. Write a simple C++ program to store and display integer elements using STL Vector class.

Note 1: In the practical Examination each student has to pick one question from PART-A and PART-B each.

Note 2: Change of program is not permitted in the Practical Examination.

UNIX AND SHELL PROGRAMMING LAB

Semester	I	CIE Marks : 40
Course Code	18MCA17	SEE Marks : 60
Contact Periods (L:T:P)	0-1-3	Exam Hours : 03
Credits : 02		

Course learning objectives:

This course (18MCA17) will enable student to

- Provide a clear understanding of core aspects of UNIX operating system, focusing on editors, command usage, filters, regular expressions, and other utility tools.
- Explore the fundamentals of UNIX command set and their usage to provide sufficient knowledge on writing Shell scripts.

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Understand the Unix programming environment.

CO2: Acquire the knowledge of using Vi editor.

CO3: Design and implement shell scripts to manage users with different types of Permission and file based applications.

CO4: Build Awk scripts.

Laboratory Experiments:

- a) Explore the Unix environment.
- b) Explore vi editor with vim tutor. Perform the following operations using vi editor, but not limited to:
 1. Insert character, delete character, replace character
 2. Save the file and continue working
 3. Save the file and exit the editor
 4. Quit the editor
 5. Quit without saving the file
 6. Rename a file
 7. Insert lines, delete lines,
 8. Setline numbers
 9. Search for a pattern
 10. Move forward and backward
- 1a. Write a shell script that takes a valid directory name as an argument and recursively descend all the sub-directories, finds the maximum length of any file in that hierarchy and write this

maximum value to the standard output.

- 1b. Write a shell script that accepts a path name and creates all the components in that path name as directories. For example, if the script is named mpc, then the command mpc a/b/c/d should create directories a, a/b, a/b/c, a/b/c/d.
- 2a. Write a shell script that accepts two filenames as arguments, checks if the permissions for these files are identical and if the permissions are identical, output common permissions and otherwise output each filename followed by its permissions.
- 2b. Write a shell script which accepts valid log-in names as arguments and prints their corresponding home directories, if no arguments are specified, print a suitable error message.
- 3a. Create a script file called file-properties that reads a filename entered and outputs its Properties.
- 3b. Write shell script to implement terminal locking (similar to the lock command). It should prompt the user for a password. After accepting the password entered by the user, it must prompt again for the matching password as confirmation and If match occurs, it must lock the keyword until a matching Password is entered again by the user, Note that the script must be written to disregard BREAK, control-D. No time limit need be implemented for the lock duration.
- 4a. Write a shell script that accept one or more file names as argument and convert all of them to uppercase, provided they exist in current directory.
- 4b. Write a shell script that displays all the links to a file specified as the first argument to the script. The second argument, which is optional, can be used to specify in which the search is to begin. If this second argument is not present, the search is to begin in current working directory. In either case, the starting directory as well as all its subdirectories at all levels must be searched. The script need not include any error checking.
- 5a. Write a shell script that accepts a filename as argument and display its creation time if file exist and if it does not send output error message.
- 5b. Write a shell script to display the calendar for current month with current date replaced by *or** depending on whether the

date has one digit or two digits.

- 6a. Write a shell script to find a file/s that matches a pattern given as command line argument in the home directory, display the contents of the file and copy the file into the directory ~/mydir.
- 6b. Write a shell script to list all the files in a directory whose filename is atleast 10 characters.(use expr command to check the length)
- 7a. Write a shell script that gets executed displays the message either “Good Morning” or “Good Afternoon” or “Good Evening” depending upon time at which the user logs in.
- 7b. Write a shell script that accept a list of filenames as its argument, count and report occurrence of each word that is present in the first argument file on other argument files.
- 8a. Write a shell script that determine the period for which as specified use ris working on system and display appropriate message.
- 8b. Write a shell script that reports the logging in of as specified user within one minute after he/she login. The script automatically terminate if specified user does not login during as specified period of time.
- 9a. Write a shell script that accept the filename, starting and ending line number as an argument and display all the lines between the given line number.
- 9b. Write a shell script that folds long lines into 40 columns. Thus any line that exceeds 40 characters must be broken after 40th, a “\” is to be appended as the indication of folding and the processing is to be continued with the residue. The input is to be supplied through a text file created by the user.
- 10a. Write an awkscript that accepts date argument in the form of dd-mm-yy and display it in the form month, day and year. The script should check the validity of the argument and in the case of error, display a suitable message.
- 10b. Write an awkscript to delete duplicated line from a text file. The order of the original lines must remain unchanged.
- 11a. Write an awkscript of find out total number of books sold in each discipline as well as total book sold using associate array down table as given below.

Electrical 34

Mechanical 67
Electrical 80
Computer Science 43
Mechanical 65
Civil 98
Computer Science 64

- 11b. Write an awkscript to compute gross salary of an employee accordingly to rule given below.
If basic salary is <10000 then HRA=15% of basic &DA=45% of basic
If basic salary is >=10000 then HRA=20% of basic &DA=50% of basic.

Note 1: In the practical Examination each student has to pick one question from a lot of all 11 questions.

Note 2: Change of program is not permitted in the Practical Examination.

WEB TECHNOLOGIES LAB

Semester	I	CIE Marks : 40
Course Code	18MCA18	SEE Marks : 60
Contact Periods (L:T:P)	0-1-3	Exam Hours : 03
Credits : 02		

Course learning objectives:

This course (18MCA18) will enable student to

- Make use of different tools to develop Web Applications.
- Build platform independent dynamic web applications using JavaScript, JQuery
- Show the usage of XML for data storage and to display the XML document in browser.

Course Outcome (CO):

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

CO1: Develop web pages using XHTML and HTML5.

CO2: Demonstrate the usage of CSS in designing web pages.

CO3: Execute simple programming questions using JavaScript.

CO4: Create dynamic web pages by manipulating the DOM elements.

CO5: Design and implement user interactive dynamic web based applications using jQuery.

Laboratory Experiments:

1. Create an XHTML page that provides information about your department. Your XHTML page must use the following tags:
 - a) Text Formatting tags
 - b) Horizontal rule
 - c) Meta element
 - d) Links
 - e) Images
 - f) Tables(Use of additional tags encouraged).
2. Develop and demonstrate the usage of inline, external and internal style sheet using CSS. Use XHTML page that contains at least three paragraphs of text, listed elements and a table with four rows and four columns.
3. Develop and demonstrate a XHTML file that includes Javascript script for the following problems:

- a) Input : A number n obtained using prompt Output : The first n Fibonacci numbers
 - b) Input : A number n obtained using prompt Output : A table of numbers from 1 to n and their squares using alert
4. Write a JavaScript program to generate n number of random numbers and store them in an array. Sort the generated numbers in ascending order using array sort method. Develop separate functions to find mean and median of numbers that are in the array. Display the results with appropriate messages.
 5. Create a XHTML document that describes the form for taking orders for popcorn. Text boxes are used at the top of the form to collect the buyer's name and address. These are placed in a borderless table to force the text box align vertically. A second table to collect actual order. Each row of this table names a product, displays the price, and uses text box with size 2 to collect the quantity ordered using <td> tag. The payment method is input by the user through one of four radio buttons. Provide provision for submission of order and clear the order form.

Sample output form

Product Name	Price	Quantity
Unpopped Popcorn (1 lb.)	\$3.00	<input type="text"/>
Caramel Popcorn (2 lb. canister)	\$3.50	<input type="text"/>
Caramel Nut Popcorn (2 lb. canister)	\$4.50	<input type="text"/>
Toffey Nut Popcorn (2 lb. canister)	\$5.00	<input type="text"/>

6. Develop, test and validate an XHTML document that has checkboxes for apple (59 cents each), orange (49 cents each), and banana (39 cents each) along with submit button. Each checkboxes should have its own onclick event handler. These

handlers must add the cost of their fruit to a total cost. An event handler for the submit button must produce an alert window with the message 'your total cost is \$xxx', where xxx is the total cost of the chose fruit, including 5 percent sales tax. This handler must return 'false' (to avoid actual submission of the form data).

Modify the document to accept quantity for each item using textboxes.

7. a) Develop and demonstrate, a HTML document that collects the USN (the valid format is : A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by three upper-case characters followed by two digits; (no embedded spaces are allowed) from the user. Use JavaScript that validate the content of the document. Suitable messages should be display in the alert if errors are detected in the input data. Use CSS and event handlers to make your document appealing.
b) Modify the above program to get the current semester also(restricted to be a number from 1 to 6)
8. Develop and demonstrate a HTML file which includes JavaScript that uses functions for the following problems:
 - a. Parameter: A string Output: The position in the string of the left-most vowel.
 - b. Parameter: A number Output: The number with its digits in the reverse order.
9. Develop and demonstrate a HTML5 page which contains
 - a) Dynamic Progressive bar.
 - b) Display Video file using HTML5 video tag.
10. Develop and demonstrate, using JavaScript script, a XHTML document that contains three short paragraphs of text, stacked on top of each other, with only enough of each showing so that the mouse cursor can be placed over some part of them. When the cursor is placed over the exposed part of any paragraph, it should rise to the top to become completely visible. Modify the above document so that when a text is moved from the top stacking position, it returns to its original position rather than to the bottom.
11. Develop a simple calculator to perform arithmetic (addition,

subtraction, multiplication and division) operations on given two numbers. Use an HTML tag that allows the user to input two numbers and to display the result of arithmetic operation. Write suitable HTML and JavaScript and CSS to your simple calculator. The following figure show sample document display.

A SIMPLE CALCULATOR

Number 1 =

Number 2 =

Result =

12. Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, and Name of the College, Branch, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.
13. Develop and demonstrate using jQuery to solve the following:
 - a) Limit character input in the text area including count.
 - b) Based on check box, disable/enable the form submit button.
14. Develop and demonstrate using jQuery to solve the following:
 - a) Fade in and fade out all division elements.
 - b) Animate an element, by changing its height and width.

Note 1: In the practical Examination each student has to pick one question from a lot of all 14 Questions.

Note 2: Change of program is not permitted in the Practical Examination.

SEMESTER – II PROGRAMMING USING JAVA

Semester	II	CIE Marks : 40
Course Code	18MCA21	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course (18MCA21) will enable students to:

- Familiarize the basic concepts of java Programming constructs and OOPs concepts
- Grasp the significance of Generalization and runtime polymorphism applications, usage of Packages, Interfaces, Exceptions and Multithreading
- Learn to apply Wrappers, Autoboxing, Generics, collection framework and I/O operations and concepts of Networking using Java network classes

Course Outcome(CO):

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

CO1: Understand the basic programming constructs of Java. Apply suitable OOP concepts to develop Java programs for a given scenario.

CO2: Illustrate the concepts of Generalization and runtime polymorphism applications

CO3: Exemplify the usage of Packages, Interfaces, Exceptions and Multithreading

CO4: Demonstrate Enumerations, Wrappers, Auto boxing, Generics, collection framework and I/O operations

CO5: Implement the concepts of Networking using Java network classes

MODULE-1:

Java Programming Fundamentals

The Java Language, The Key Attributes of Object-Oriented Programming, The Java Development Kit, A First Simple Program, The Java Keywords, Identifiers in Java.

Introducing Data Types and Operators

Java's Primitive Types, Literals, A Closer Look at Variables, The Scope and Lifetime of Variables, operators, Shorthand Assignments,

Type conversion in Assignments, Operator Precedence.

Program Control Statements

Input characters from the Keyboard, Use break, Use continue ,

Introduction to Classes, Objects and Methods

Class Fundamentals, How Objects are Created, Reference Variables and Assignment, Methods, Constructors, Parameterized Constructors, The new operator Revisited, Garbage Collection and Finalizers, The this Keyword.

More Data Types and Operators

Arrays, Multidimensional Arrays, Alternative Array Declaration Syntax, Assigning Array References, Using the Length Member, The For-Each Style for Loop, Strings.

MODULE-2:

String Handling

String Fundamentals, The String Constructors, Three String-Related Language Features, The Length() Method, Obtaining the characters within a string, String comparison, using indexOf() and lastIndexOf(), Changing the case of characters within a string, String Buffer and String Builder.

A Closer Look at Methods and Classes

Controlling Access to Class Members, Method Overloading, Overloading Constructors, Recursion, Understanding Static, Introducing Nested and Inner Classes, Varargs: Variable-Length Arguments.

MODULE-3:

Inheritance

Inheritance Basics, Member Access and Inheritance, Constructors and Inheritance, Using super to Call Super class constructors, Using super to Access Super class Members, Creating a Multilevel Hierarchy, When are Constructors Executed, Superclass References and Subclass Objects, Method Overriding, Overridden Methods support polymorphism, Why Overridden Methods, Using Abstract Classes, Using final, The Object Class.

Interfaces

Interface Fundamentals, Creating an Interface, Implementing an Interface, Using Interface References, Implementing Multiple

Interfaces, Constants in Interfaces, Interfaces can be extended, Nested Interfaces

Packages

Package Fundamentals, Packages and Member Access, Importing Packages, Static Import

MODULE-4:

Exception Handling

The Exception Hierarchy, Exception Handling Fundamentals, The Consequences of an Uncaught Exception, using Multiple catch clauses, Catching subclass Exceptions, try blocks can be nested, Throwing an Exception, A Closer look at Throwable, using finally, using throws, Java's Built-in Exceptions,

Multithreaded Programming

Multithreading fundamentals The Thread Class and Runnable Interface, Creating Thread, Creating Multiple Threads, Determining When a Thread Ends, Thread Priorities, Synchronization, using Synchronization Methods, The Synchronized Statement , Thread Communication using notify(), wait() and notifyAll(), suspending, Resuming and stopping Threads.

MODULE-5:

Enumerations, Auto boxing

Enumerations, Java Enumeration are class types, TheValues() and Valueof() Methods, Constructors, methods, instance variables and enumerations, Autoboxing, Annotations (metadata)

Networking with Java.net

Networking fundamentals, The Networking classes and Interfaces, The Inet Address class, The Socket Class, The URL class, The URL Connection Class, The Http URL Connection Class.

Exploring Collection Framework

Collections Overview, The Collection Interfaces, The collection Classes. The Arrays Class.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.

- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each MODULE

Textbooks:

1. Java Fundamentals, A comprehensive Introduction by Herbert Schildt, Dale Skrien. Tata McGraw Hill Edition 2013. (Chapters:1,2,3,4,5,6,7,8,9,10,11,12,13,15,22,23,24,25,26)

Reference Books:

1. Java Programming by Hari Mohan Pandey, Pearson Education, 2012.
2. Java 6 Programming, Black Book, KoGenT, Dreamtech Press, 2012.
3. Java 2 Essentials, Cay Hortsman, second edition, Wiley

DATA STRUCTURES USING C++

Semester	II	CIE Marks : 40
Course Code	18MCA22	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course (18MCA22) will enable students to:

- Familiarize the knowledge of various types of data structures, operations and algorithms sorting and searching operations.
- Use the concepts of Stack, Queue, Lists, Trees, Hashing, Searching and Sorting techniques
- Build solutions for real world problems using concepts of data structures

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Acquire knowledge of

- Various types of data structures, operations and algorithms
- Sorting and searching operations

CO2: Analyze the performance of

- Stack, Queue, Lists, Trees, Hashing, Searching and Sorting techniques

CO3: Implement all the applications of Data structures in a high-level language

CO4: Design and apply appropriate data structures for solving computing problems

MODULE-1:

Classification of Data Structures: Primitive and Non- Primitive, Linear and Nonlinear; Data structure Operations, ADT, Array as ADT, Operations - Insert, Delete, Search, Sort, String Definition, Representation, String as ADT, Operations – Insert, Delete, Concatenate, Comparing, Substring.

Stack: Definition, Representation, Stack as ADT, Operations and Applications: Polish and reverse polish expressions, Infix to postfix conversion, evaluation of postfix expression, infix to prefix, postfix to infix conversion;

MODULE-2:

Recursion - Recursive definition and processes, Properties of recursive definition or Algorithm, Recursive algorithms: Factorial, GCD, Fibonacci Sequence, Tower of Hanoi.

Queue: Definition, Representation, Queue as ADT, Operations, Queue Variants: Circular Queue, Priority Queue, Double Ended Queue; Applications of Queues. Programming Examples.

MODULE-3:

Linked List: Limitations of array implementation, Memory Management: Static (Stack) and Dynamic (Heap) Memory Allocation, Memory management functions. Definition, Representation, Operations: getnode() and Freenode() operations, Types: Singly Linked List. Linked list as a data Structure, Inserting and removing nodes from a list, Linked implementations of stacks, Example of list operations such as insert and delete an element before a key element, Header nodes, Array implementation of lists.: Circular Linked List: Inserting, deleting and searching elements in a lists, Double Linked List: Inserting and Deleting Nodes, Queue as doubly linked lists, such as insert into position, Delete an specified element. Application of Linked Lists: Stacks, Queues, Double-ended Queues, Priority Queues, Sparse Matrix and Polynomials using Lists, Trees, BST.

MODULE-4:

Trees: Definitions, Terminologies, Array and linked Representation of Binary Trees, Types- Complete/full, Almost Complete, Strictly, Skewed; Traversal methods - Inorder, postorder, preorder; Binary Search Trees - Creation, Insertion, Deletion, Traversal, Searching.

MODULE-5:

Sorting & Searching: Bubble sort, Insertion Sort, Selection sort, Quick sort, Linear Search, Binary Search and BST.

Hashing: The Hash Table organizations, Hashing Functions, Static and Dynamic Hashing, Collision-Resolution Techniques, Programming Examples.

Question paper pattern:

- The question paper will have ten questions.

- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each MODULE

Textbooks:

1. Programming in ANSI C, Balaguruswamy, McGraw Hill Education.
2. Data Structures Using C and C++ by YedidyahLangsam and Moshe J. Augenstein and Aaron M Tenanbanum, 2nd Edition, Pearson Education Asia, 2002.
3. Introduction to Data Structure and Algorithms with C++ by Glenn W. Rowe. Prentice Hall.

Reference Books:

1. Principles of Data Structures using C & C++ by Vinu V. Das, New Age International, 2006
2. Data Structures Using C , Balaguruswamy, McGraw Hill Education

DISCRETE MATHEMATICAL STRUCTURES AND STATISTICS

Semester	II	CIE Marks : 40
Course Code	18MCA23	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course (18MCA23) will enable students to:

- Familiarize the logical notation to define and reason about fundamental mathematical concepts such as sets, relations, functions and integers.
- Use elementary combinatorial processes such as permutations and combinations.
- Understand probabilities and distributions for simple combinatorial processes, and statistical methods for correlation and regression.

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Use the logical notation to define and reason about fundamental mathematical concepts such as sets, relations, functions and integers.

CO2: Calculate numbers of possible outcomes of elementary combinatorial processes such as permutations and combinations.

CO3: Calculate probabilities and distributions for simple combinatorial processes; calculate expectations.

CO4: Apply statistical methods for correlation and regression. Fitting a curve to a discrete data.

MODULE-1:

Fundamentals of Logic

Basic connectives and truth tables, logical equivalence, laws of logic, logical implication rules of inference. Quantifiers Propositional logic, equivalences, predicates and quantifiers, rules of inference, introduction to proofs, proof methods.

MODULE-2:

Sets Theory and Probability

Sets and subsets, set operations, laws of set theory, counting and venn

diagrams. A first word on probability, axioms of probability, conditional probability, Bayes' theorem.

MODULE-3:

Fundamentals of Counting and Properties of Integers

The rules of Sum and Product, Permutations and Combinations, The Binomial theorem, Mathematical Induction, Recursive definitions: Fibonacci and Lucas numbers

MODULE-4:

Random variables and Probability Distributions

Concept of a random variable Discrete probability distributions, Continuous probability distributions, Mean, Variance and Covariance of random variables. Binomial and Poisson distributions, Exponential and Normal distributions with mean, variables and problems.

MODULE-5:

Statistical methods and Curve Fitting

Correlation, coefficient of correlations, lines of regression-principle of least square. Rank correlation. Curve Fitting- Graphical method, Principle of least square- to fit a straight line and parabola. Fitting of other curves of the form $y = ax^b$ $y = ae^{bx}$ $xy^n = b$

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each MODULE

Textbooks:

1. Discrete and Combinatorial Mathematics by Ralph P. Grimaldi and B V Ramana, 5th edition, Pearson, 2011.
(Chapters: 1.1 to 1.3 4.1, 4.2, 2.1 to 2.5, 3.1 to 3.6)
2. Probability and Statistics for Engineers and Scientists by

Walpole Myers Ye Pearson Education, Eighth edition.

(Chapters: 3.1–3.3, 4.1 to 4.4, 5.3, 5.6, 6.2 to 6.4, 6.6, 6.7)

3. Higher Engineering Mathematics by Dr. B. S. Grewal, Khanna publishers, 40th edition
(Chapters: 25.12 to 25.16, 24.1 to 24.6)

Reference Books:

1. Discrete Mathematics and its Applications by Kenneth H Rosen, 7th edition, (Indian adaptation by Kamala Krithivasan), Tata McGraw Hill, 2011.
2. Discrete Mathematical Structures with Applications to Computer Science by J.P. Tremblay and R. Manohar, McGrawHill.
3. Probability and Statistics for Engineers by Richard A. Johnson and C. B. Gupta, Pearson Education.
4. Discrete Mathematics by J. K. Sharma, Macmillan Publishers India Ltd. 3rd edition 2011.

COMPUTER NETWORKS

Semester	II	CIE Marks : 40
Course Code	18MCA24	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course (18MCA24) will enable students to:

- Impart knowledge on basic concepts of computer networks
- Provide an understanding of various types of Networks & Communication media
- Provide knowledge of different components required to build different types of networks
- Give an insight of Understand layering concepts of TCP/IP and OSI models and working principles of various application protocols

Course Outcome (CO): The students will be able to

CO1: Understand the computer network concepts.

CO2: Know various types of Networks & Communication media

CO3: Identify the components required to build different types of networks

CO4: Understand layering concepts of TCP/IP and OSI models

CO5: Understand the working principles of various application protocols

MODULE -1:

Introduction: Applications, Requirements, Network Architecture, Implementing Network Software, Performance.

MODULE -2:

Getting Connected: Perspectives on Connecting, Encoding (NRZ, NRZI, Manchester, 4B/5B), Framing, Error Detection, Reliable Transmission, Ethernet and Multiple Access Networks (802.3), Wireless.

MODULE -3:

Internetworking and Advanced Internetworking: Switching and Bridging, Basic Internetworking (IP), Routing, The Global Internet, Routing among Mobile Devices

MODULE -4:

End-to-End Protocols and Congestion Control: Simple Demultiplexer (UDP), Reliable Byte Stream (TCP), Queuing Disciplines, TCP Congestion Control, Congestion-Avoidance Mechanisms.

MODULE -5:

Network Security and Applications: Cryptographic Building Blocks, Key Predistribution, Firewalls, Traditional Applications, Infrastructure Services.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each MODULE

Textbook:

1. “Computer Networks A Systems Approach” by Larry L Peterson and Bruce S Davie, 5th Edition, MKP – 2012 – (1, 2, 3.1, 3.2, 3.3, 3.4, 4.1, 5.1, 5.2, 6.2, 6.3, 6.4, 8.1, 8.2, 8.5, 9.1, 9.3)

Reference Books:

1. James F. Kurose, Keith W. Ross, “Computer Networking – A Top-Down Approach Featuring The Internet”, Fifth Edition, Pearson Education, 2009.
2. Nader. F. Mir, “Computer And Communication Networks”, Pearson Prentice Hall Publishers, 2010.
3. Ying-Dar Lin, Ren-Hung Hwang, Fred Baker, “Computer Networks: An Open Source Approach”, Mc Graw Hill Publisher, 2011.
4. Behrouz A. Forouzan, “Data Communication And Networking”, Fourth Edition, Tata McGraw – Hill, 2011.

Operating Systems

Semester	II	CIE Marks : 40
Course Code	18MCA25	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course (18MCA25) will enable students to:

- Impart knowledge on basic of Computer and Operating System Structure
- Provide knowledge of Process Management concepts
- Give an insight of Synchronization and Deadlock, and system software concepts like Assemblers, Loaders and Linkers

Course Outcome (CO): The students will be able to

CO1: Understand the basic of Computer and Operating System Structure

CO2: Realize the concept of Process Management

CO3: Analyze and Evaluation of Synchronization and Deadlock

CO4: Understand the concepts of File System & Secondary Storage

MODULE -1:

Computer and Operating Systems Structure

Basic Elements, Processor Registers, Instruction Execution, The Memory hierarchy, Cache Memory, I/O Communication Techniques, Introduction to Operating System, Mainframe Systems, Desktop Systems, Multiprocessor Systems, Distributed Systems, Clustered Systems, Real - Time Systems, Handheld Systems, Feature Migration, Computing Environments. System Structures: System Components, Operating – System Services, System Calls, System Programs, System Structure, Virtual Machines, System Design and Implementation, System Generation

MODULE -2:

Process Management and Mutual Execution

Process, Process States, Process Description, Process Control, Execution of the Operating System, Security Issues, Processes and Threads, Symmetric Multiprocessing(SMP), Micro kernels, CPU Scheduler and Scheduling. Principles of Concurrency, Mutual

Exclusion: Hardware Support, Semaphores , Monitors , Message Passing, Readers/Writes Problem.

MODULE -3:

Deadlock and Memory Management

Principles of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, An Integrated Deadlock Strategy, Dining Philosophers Problem. Memory Management: Swapping, Contiguous Memory Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Process Creation, Page Replacement, Allocation of Frames, Thrashing

MODULE -4

File System and Secondary Storage

File Concept, Access Methods, Directory Structure, File System Mounting, File Sharing, Protection, File – System Structure, File – System Implementation, Directory Implementation, Allocation Methods, Free–Space Management, Disk Structure, Disk Scheduling, Disk Management.

MODULE -5:

Computer Security and Case study of Linux Operating system

The Security Problem, User Authentication, Program Threats, System Threats. Linux System Linux history , Design Principles, Kernel modules, Process , management, scheduling, Memory management, File systems, Input and output, Inter-process communications

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each MODULE

Textbooks:

1. Silberschatz, Galvin, Gagne, “Operating System Concepts”

John Wiley, Sixth Edition, 2004.

2. William Stallings, “Operating System Internals and Design Principles” Pearson, 6th edition, 2012

Reference Books:

1. Chakraborty, “Operating Systems” Jaico Publishing House, 2011.

2. Dhananjay M. Dhamdhere, “Operating Systems – A Concept – Based Approach”, Tata McGraw – Hill, 3rd Edition, 2012.

JAVA PROGRAMMING LAB

Semester	II	CIE Marks : 40
Course Code	18MCA26	SEE Marks : 60
Contact Periods (L:T:P)	0-1-3	Exam Hours : 03
Credits : 02		

Course Learning Objectives:

This course (18MCA26) will enable students to:

- Familiarize Java programming language fundamentals and run time environment
- Write programs using OOCs using Java
- Write the multithreading and client side programs using Java
- Implement the concepts of data structure using java programming language

Course Outcome (CO): The students will be able to

CO1: Understand Java programming language fundamentals and run time environment.

CO2: Acquire knowledge and skill necessary to write java programs.

CO3: Learn the object oriented concepts and its implementation in Java

CO4: Implement the multithreading and client side programming.

PART A

1. a) Write a JAVA program to demonstrate Constructor Overloading and Method Overloading.
b) Write a JAVA program to implement Inner class and demonstrate its Access protection.
2. Write a program in Java for String handling which performs the following:
 - i) Checks the capacity of String Buffer objects.
 - ii) Reverses the contents of a string given on console and converts the resultant string in upper case.
 - iii) Reads a string from console and appends it to the resultant string of (ii).
3. a). Write a JAVA program to demonstrate Inheritance.
b). Simple Program on Java for the implementation of Multiple inheritance using interfaces to calculate the area of a rectangle and triangle.
4. Write a JAVA program which has

- i) A Class called Account that creates account with 500Rs minimum balance, a deposit() method to deposit amount, a withdraw() method to withdraw amount and also throws LessBalanceException if an account holder tries to withdraw money which makes the balance become less than 500Rs.
 - ii) A Class called Less Balance Exception which returns the statement that says withdraw amount (Rs) is not valid.
 - iii) A Class which creates 2 accounts, both account deposit money and one account tries to withdraw more money which generates a Less Balance Exception take appropriate action for the same.
5. Write a JAVA program using Synchronized Threads, which demonstrates Producer Consumer concept.
 6. Write a JAVA program to implement a Queue using user defined Exception Handling (also make use of throw, throws).
 7. Complete the following:
 - i. Create a package named shape.
 - ii. Create some classes in the package representing some common shapes like Square, Triangle, and Circle.
 - iii. Import and compile these classes in other program.
 8. Write a JAVA program to create an enumeration Day of Week with seven values SUNDAY through SATURDAY. Add a method isWorkday() to the DayofWeek class that returns true if the value on which it is called is MONDAY through FRIDAY. For example, the call DayOfWeek.SUNDAY.isWorkDay () returns false.
 9. Write a JAVA program which has
 - i). A Interface class for Stack Operations
 - ii). A Class that implements the Stack Interface and creates a fixed length Stack.
 - iii). A Class that implements the Stack Interface and creates a Dynamic length Stack.
 - iv). A Class that uses both the above Stacks through Interface reference and does the Stack operations that demonstrates the runtime binding.
 10. Write a JAVA program which uses FileInputStream / FileOutPutStream Classes.
 11. Write JAVA programs which demonstrate utilities of Linked List Class.

12. Write a JAVA program which uses Datagram Socket for Client Server Communication.

Note 1: In the practical Examination student has to execute one program from a lot of all the 12 questions and demonstrate Part B Mini Project.

Note 2: Change of program is not permitted in the Practical Examination.

DATA STRUCTURES LAB

Semester	II	CIE Marks : 40
Course Code	18MCA27	SEE Marks : 60
Contact Periods (L:T:P)	0-1-3	Exam Hours : 03
Credits : 02		

Course Learning Objectives:

This course (18MCA27) will enable students to:

- Familiarize the knowledge of
 - Various types of data structures, operations and algorithms
 - Sorting and searching operations
- Implement and analyse the performance of
 - Stack, Queue, Lists, Trees, Hashing, Searching and Sorting techniques
- Implement all the applications of Data structures in a high-level language
- Suggest and apply appropriate data structures for solving computing problems

Course Outcome (CO): The students will be able to

CO1: Acquire knowledge of

- Various types of data structures, operations and algorithms
- Sorting and searching operations

CO2: Analyze the performance of

- Stack, Queue, Lists, Trees, Hashing, Searching and Sorting techniques

CO3: Implement all the applications of Data structures in a high-level language

CO4: Design and apply appropriate data structures for solving computing problems

Lab Exercises:

1. Convert an infix notation to postfix notation.
2. Evaluate a given postfix expression and its values for the variables.
3. Simulate the working of a circular queue providing the following operations—Insert, Delete and Display.
4. Demonstrate recursion
 - a. Calculate GCD and LCM of 3 integer numbers
 - b. Solve Towers of Hanoi Problem

- c. Calculate the sum for a given number 'n' from 1 to n.
5. Simulate the working of a linked list providing the following operations
 - a. Insert at the beginning
 - b. Insert at the end
 - c. Insert before a given element
 - d. Insert at the position
 - e. Display
 6. Simulate the working of a circular linked list providing the following operations
 - a. Delete from the beginning
 - b. Delete from the end
 - c. Delete a given element
 - d. Delete every alternate element
 - e. Display
 7. Simulate the working of a dequeue.
 8. Simulate the working of a double linked list to implement stack and queue.
 9. Create a binary tree and implement the tree traversal techniques of inorder, preorder and postorder.
 10. Implement quick sort.
 11. Implement a program to merge two doubly linked list.
 12. Implement the search techniques of
 - a. Linear Search
 - b. Binary Search

Note 1: In the practical Examination student has to execute one program from a lot of all the 12 questions.

Note 2: Change of program is not permitted in the Practical Examination.

COMPUTER NETWORKS LAB

Semester	II	CIE Marks : 40
Course Code	18MCA28	SEE Marks : 60
Contact Periods (L:T:P)	0-1-3	Exam Hours : 03
Credits : 02		

Course Learning Objectives:

This course (18MCA28) will enable students to:

- Familiarize the basic terminologies used for computer networking.
- Implement the computer networks concepts like TCP/IP, IPC and Congestion avoidance techniques
- Simulate the working of wired networks and analyze its performance

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Understand the basic terminologies used for computer networking.

CO2: Understand the functions of layers in the Internet Model.

CO3: Demonstrate application layer protocols used for process to process communication.

CO4: Demonstrate subnetting and routing mechanisms for a given network topology. Exemplify link layer functionalities.

PART – A

Note: Implement the following Computer Networks concepts using C/C++

1. Write a program for distance vector algorithm to find suitable path for transmission.
2. Using TCP/IP sockets, write a client-server program to make the client send the file name and to make the server send back the contents of the requested file if present.
3. Write a program for Hamming code generation for error detection and correction.
4. Write a program for congestion control using leaky bucket algorithm.

PART – B

Note: Simulate the following Computer Networks concepts using any network simulators.

1. Simulate a three nodes point — to — point network with duplex links between them. Set the queue size and vary the bandwidth and find the number of packets dropped.
2. Simulate the network with five nodes n0, n1, n2, n3, n4, forming a star topology. The node n4 is at the center. Node n0 is a TCP source, which transmits packets to node n3 (a TCP sink) through the node n4. Node n1 is another traffic source, and sends UDP packets to node n2 through n4. The duration of the simulation time is 10 seconds.
3. Simulate to study transmission of packets over Ethernet LAN and determine the number of packets drop destination.
4. Write a TCL Script to simulate working of multicasting routing protocol and analyze the throughput of the network.
5. Simulate the different types of internet traffic such as FTP and TELNET over a wired network and analyze the packet drop and packet delivery ratio in the network.
6. Simulate the transmission of ping messages over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.

Note 1: In the practical Examination student has to execute one program from PART-A and one from PART-B.

Note 2: Change of program is not permitted in the Practical Examination.

MINI PROJECT

Semester	II	CIE Marks : 40
Course Code	18MCA29	SEE Marks : 60
Contact Periods (L:T:P)	0-2-2	Exam Hours : 03
Credits : 02		

Course Learning Objectives:

This course (18MCA29) will enable students to:

- Acquire practical knowledge within the chosen area of technology for project development
- Identify, analyze, formulate and handle programming projects with a comprehensive and systematic approach
- Contribute as an individual or in a team in development of technical projects
- Develop effective communication skills for presentation of project related activities

Course outcomes:

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

CO1: Identify a suitable problem making use of the technical and engineering knowledge gained from previous courses with the awareness of impact of technology on the society and their ethical responsibilities.

CO2: Ability to segregate work and execute/implement projects using appropriate tools.

CO3: Develop skills to disseminate technical and general information by means of oral as well as written presentation skills.

Develop an application using the languages and concepts learnt in the theory and exercises listed in I and II semester with a good look and feel effects.

Note :

1. In the examination student has to demonstrate the project.
2. A team of maximum two students must develop the project. However during the examination, each student must demonstrate the project individually.
3. The team must submit a brief project report (20-30 pages) that must include the following
 - a. Introduction

- b. Requirement Analysis
- c. Software Requirement Specification
- d. Analysis and Design
- e. Implementation
- f. Testing
- g. Conclusion

SEMESTER – III DATABASE MANAGEMENT SYSTEM

Semester	III	CIE Marks : 40
Course Code	18MCA31	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA31) will enable student to

- Understand the fundamental concepts of Database Management systems.
- Design ER Diagrams, Schema and Relational tables.
- Understand how to develop database Application.

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Demonstrate the fundamentals of data models and conceptualize and depict a

database system and Make use of ER diagram in developing ER Model

CO2: To Summarize the SQL and relational database design.

CO3: Illustrate transaction processing, concurrency control techniques and recovery

CO4: Inference the database design in the real world entities.

MODULE -1:

Introduction

Characteristics of Database approach, Actors on the Scene, Workers behind the scene, Advantages of using DBMS approach, Data models, schemas and instances, Three -schema architecture and data independence, Database languages and interfaces, the database system environment, Centralized and client -server architectures, Classification of Database Management systems, Entity-Relationship Model: Conceptual Database using high level conceptual data models for Database Design, A Sample Database Application, Entity types, Entity sets Attributes and Keys Relationship types, Relationship Sets, Roles and Structural Constraints Weak Entity Types.

MODULE -2:

Relational Model

Relational Model and Relational Algebra: Relational Model Concepts, Relational Model Constraints, Relational Model Constraints and Relational Database Schema Update Operations, Transactions and Dealing with Constraint violations, Unary Relational operations, Relational Algebra Operations from Set Theory, Binary Relational Operations, JOIN and DIVISION, Additional Relational Operations, Examples of Queries in Relational Algebra Relational Database Design Using ER-to-Relational Mapping

MODULE -3:

Introduction to SQL

Overview of the SQL Query Language, SQL Data Definition, Basic structure of SQL Queries, Additional Basic Operations, Null values, Aggregate Functions, nested Sub queries, Modification of the Database, Join Expressions, Views, Transactions, Integrity Constraints, SQL Data Types and Schemas, Authorization. Database programming issues and techniques, Embedded SQL.

MODULE -4:

Database Design

Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms based on Primary Keys, General Definitions of 2nd and 3rd Normal Forms, Boyce Codd Normal Forms, Stored Procedures and functions, Triggers.

MODULE -5:

Transaction Management

Transaction Concept, A Simple Transaction Model, Transaction Atomicity and Durability, Serializability, Transaction Isolation and Atomicity, Transaction Isolation Levels, Implementation of Isolation Levels. Concurrency Control: Lock Based Protocols, Deadlock Handling. Recovery System: Failure Classification, Storage, Recovery and Atomicity.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.

- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Elmasri and Navathe: Fundamentals of Database Systems, 5th Edition, Addison -Wesley, 2011.
2. Silberschatz, Korth and Sudharshan Data base System Concepts,6th Edition, Tata McGraw Hill, 2011.

Reference Books:

1. C.J. Date, A. Kannan, S. Swamynatham: An Introduction to Database Systems, 8th Edition, Pearson education,2009.
2. Raghu Ramakrishnan and Johannes Gehrke: Database Management Systems, 3rd Edition, McGraw-Hill, 2003.

PROGRAMMING USING PYTHON

Semester	III	CIE Marks : 40
Course Code	18MCA32	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA32) will enable student to

- Understand basic primitives of Python programming.
- Know how to handle files and design GUI
- Develop Applications using Python programming

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Understand and comprehend the basics of Python programming.

CO2: Apply knowledge in real time applications.

CO3: Understands about files and its applications.

CO4: Use standard programming constructs.

MODULE -1:

Installing Python, Simple program using Python, Expressions and Values, Variables and Computer Memory, error detection, Multiple line statements, Designing and using functions, functions provided by Python, Tracing function calls in memory model, omitting return statement. Working with Text: Creating Strings of Characters, Using Special Characters in Strings, Creating a Multiline String, Printing Information, Getting Information from the Keyboard.

MODULE -2:

A Boolean Type , Choosing Statements to Execute, Nested If Statements , Remembering the Results of a Boolean Expression Evaluation , A Modular Approach to Program Organization, Importing Modules , Defining Your Own Modules, Testing Code Semi automatically Grouping Functions Using Methods: Modules, Classes, and Methods , Calling Methods the Object-Oriented Way, Exploring String Methods, Underscores.

MODULE -3

Storing Collections of Data Using Lists: Storing and Accessing Data in Lists, modifying Lists, Operations on Lists, Slicing Lists, Aliasing, List Methods, Working with a List of Lists. Repeating Code Using Loops: Processing Items in a List, Processing Characters in Strings, Looping Over a Range of Numbers, Processing Lists Using Indices, Nesting Loops in Loops, Looping Until a Condition Is Reached, Repetition Based on User Input, Controlling Loops Using Break and Continue Reading and Writing.

MODULE -4

Files: Kinds of files, Opening a File, Techniques for Reading Files, Files over the Internet, Writing Files, and Writing Algorithms That Use the File-Reading Techniques, Multiline Records. Storing Data Using Other Collection Types: Storing Data Using Sets, Storing Data Using Tuples, Storing Data Using Dictionaries, Inverting a Dictionary, Using the In Operator on Tuples, Sets, and Dictionaries, Comparing Collections.

MODULE -5

Collection of New Information Object-Oriented Programming : Understanding a Problem Domain , Function “Isinstance,” Class Object, and Class Book , Writing a Method in Class Book, Plugging into Python Syntax: More Special Methods ,Creating Graphical User interface: Building a Basic GUI, Models, Views, and Controllers, Customizing the Visual Style Introducing few more Widgets, Object-Oriented GUIs, Keeping the Concepts from Being a GUI Mess.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Practical Programming: An introduction to Computer Science Using Python, second edition, Paul Gries, Jennifer Campbell, Jason Montojo, The Pragmatic Bookshelf.
2. Learning with Python: How to Think Like a Computer Scientist Paperback – Allen Downey , Jeffrey Elkner, 2015

Reference Books:

1. Introduction to Python for Computational Science and Engineering (A beginner's guide), Hans Fangohr.
2. Exploring Python, Timothy A. Budd, Mc Graw Hill Education
3. Python for Informatics: Exploring Information, Charles Severance.
4. Learning Python, Fourth Edition, Mark Lutz, O'Reilly publication

DESIGN AND ANALYSIS OF ALGORITHMS

Semester	III	CIE Marks : 40
Course Code	18MCA33	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA33) will enable student to

- Know the importance of designing efficient algorithm
- Know various possible algorithm design techniques/methods
- Analyze the algorithm and understand its performance

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Categorize problems based on their characteristics and practical importance.

CO2: Develop Algorithms using iterative/recursive approach

CO3: Compute the efficiency of algorithms in terms of asymptotic notations

CO4: Design algorithm using an appropriate design paradigm for solving a given problem

CO5: Classify problems as P, NP or NP Complete

CO6: Implement algorithms using various design strategies and determine their order of growth.

MODULE -1:

Introduction, Fundamentals of the Analysis of Algorithm Efficiency

Notion of Algorithm, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Fundamental data Structures. Analysis Framework, Asymptotic Notations and Basic efficiency classes, Mathematical analysis of Recursive and Non-recursive algorithms.

MODULE -2:

Brute Force: Selection Sort and Bubble Sort, Sequential Search, Exhaustive search and String Matching.

Divide-and-Conquer:

Mergesort, Quicksort, Binary Search, Binary tree Traversals and related properties, Multiplication of large integers.

MODULE -3:

Decrease-and-Conquer

Insertion Sort, Depth First and Breadth First Search, Topological sorting, Algorithms for Generating Combinatorial Objects: generating permutations.

Greedy Technique

Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees.

MODULE -4:

Space and Time Tradeoffs

Sorting by Counting, Input Enhancement in String Matching, Hashing.

Dynamic Programming

Computing a binomial coefficient, Warshall's and Floyd's Algorithms, The Knapsack Problem and Memory Functions

MODULE -5:

Limitations of Algorithm Power

Lower-Bound Arguments, Decision Trees, P, NP and NP-Complete Problems.

Coping with Limitations of Algorithm Power

Backtracking: n-Queens problem, Hamiltonian Circuit Problem, Subset – Sum Problem. Branch-and-Bound: Assignment Problem, Knapsack Problem, Traveling Salesperson Problem.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbook:

1. Anany Levitin: Introduction to the Design and Analysis of Algorithms, Pearson Education, 2nd Edition.(Chapters 1.1-

1.4, 2.1-2.4, 3.1, 3.2, 3.4, 4.1-4.5, 5.1-5.4, 7.1-7.3, 8.1, 8.2, 8.4, 9.1-9.4, 11.1-11.3, 12.1-12.2)

Reference Books:

1. Cormen T.H., Leiserson C.E., and Rivest R.L.: Introduction to Algorithms, PHI 1998.
2. Horowitz E., Sahani S., Rajasekharan S.: Computer Algorithms, Galgotia Publication 2001.
3. Michael T Goodrich and Roberto Tamassia : Algorithm Design, Wiley India
4. R C T Lee, S S Tseng, R C Chang, Y T Tsai : Introduction to Design and Analysis of Algorithms: A Strategic Approach, Tata McGraw Hill

SYSTEM SOFTWARE

Semester	III	CIE Marks : 40
Course Code	18MCA34	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA34) will enable student to

- Understand the basic design of various machine architectures
- Build ability to design an assembler
- Know the importance of various components like loaders, linkers, macro processors and compilers.

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Understand the introductory concepts of system software, SIC and SIC/XE machine architecture.

CO2: Understand the design and implementation of Assemblers with implementation examples.

CO3: Design and implement the linkers and loaders, macro processors and respective implementation examples.

CO4: Learn the basic design and working of compilers.

MODULE -1:

Machine Architecture & Introduction to Assemblers

Introduction, System Software and Machine Architecture, Simplified Instructional Computer (SIC) – SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples. VAX Architecture, UltraSPARC Architecture. Basic Assembler Function - A Simple SIC Assembler, Assembler Algorithm and Data Structures.

MODULE -2:

Assemblers

Machine Dependent Assembler Features - Instruction Formats & Addressing Modes, Program Relocation. Machine Independent Assembler Features – Literals, Symbol-Definition Statements, Expression, Program Blocks, Control Sections and Programming

Linking, Assembler Design Operations - One-Pass Assembler, Multi-Pass Assembler, Implementation Examples – MASM Assembler, SPARC Assembler

MODULE -3:

Loaders and Linkers

Basic Loader Functions - Design of an Absolute Loader, A Simple Bootstrap Loader, Machine-Dependent Loader Features – Relocation, Program Linking, Algorithm and Data Structures for a Linking Loader; Machine-Independent Loader Features – Automatic Library Search, Loader Options, Loader Design Options - Linkage Editor, Dynamic Linkage, Bootstrap Loaders, Implementation Examples – MS-DOS linker, SunOS Linker.

MODULE -4:

Macro Processor

Basic Macro Processor Functions - Macro Definitions and Expansion, Macro Processor Algorithm and Data Structures, Machine-Independent Macro Processor Features – Concatenation of Macro Parameters, Generation of Unique Labels, Conditional Macro Expansion, Keyword Macro Parameters, Macro Processor Design Options – Recursive Macro Expansion, General- Purpose Macro Processors, Macro Processing Within Language Translators, Implementation Examples - MASM, Macro Processor.

MODULE -5:

Compilers

Basic Compilers Functions- Grammars, Lexical Analysis, Syntactic Analysis, Code Generation. Machine Dependent Compiler Features- Intermediate Form of the Program, Machine dependent code Optimization. Machine Independent Compiler Features- Structured variables, Machine Independent code Optimization. Compiler Design Options- Division into passes, Interpreters, P-code Compilers, Compiler-Compilers.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.

- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbook:

1. Leland.L.Beck: System Software, 3rd Edition, Addison-Wesley, 1997.

Reference Books:

1. J. Nithyashri, “System Software”, 2nd Edition, Tata McGraw Hill, 2010.

DBMS LAB

Semester	III	CIE Marks : 40
Course Code	18MCA36	SEE Marks : 60
Contact Periods (L:T:P)	0-1-3	Exam Hours : 03
Credits : 02		

Course learning objectives:

This course (18MCA36) will enable student to

- Know various database technologies available
- Know how to connect backend databases with frontend GUI.
- Design database and SQL Queries to access databases.
- Develop real world database applications

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Understand, appreciate the underlying concepts of database technologies

CO2: Able to create database with different types of integrity constraints and use the SQL commands such as DDL, DML, DCL, TCL to access data from database objects.

CO3: Design and implement a database schema for a given problem domain

CO4: Perform embedded and nested queries

CO5: Take up real world problems independently

Lab Experiments:

Instructions for the Exercises:

1. Draw ER diagram based on given scenario with various Constraints.
 2. Create Relational Database Schema based on the scenario using Mapping Rules.
 3. Perform the given queries using any RDBMS Environment.
 4. Suitable tuples have to be entered so that queries are executed correctly
 5. The results of the queries may be displayed directly
- I. Create the following tables with properly specifying Primary keys, Foreign keys and solve the following queries.

BRANCH(Branchid,Branchname,HOD)

STUDENT(USN,Name,Address,Branchid,sem)

BOOK(Bookid,Bookname,Authorid,Publisher,Branchid)

AUTHOR(Authorid,Authorname,Country,age)

BORROW(USN,Bookid,Borrowed_Date)

Queries:

- 1 List the details of Students who are all Studying in 2nd sem MCA.
 - 2 List the students who are not borrowed any books.
 - 3 Display the USN, Student name, Branch_name, Book_name, Author_name , Books_Borrowed_Date of 2nd sem MCA Students who borrowed books.
 - 4 Display the number of books written by each Author.
 - 5 Display the student details who borrowed more than two books.
 - 6 Display the student details who borrowed books of more than one Author.
 - 7 Display the Book names in descending order of their names.
 - 8 List the details of students who borrowed the books which are all published by the same Publisher.
- II. Design an ER-diagram for the following scenario, Convert the same into a relational model and then solve the following queries.

Consider a Cricket Tournament “ABC CUP” organized by an organization. In the tournament there are many teams are contesting each having a Teamid,Team_Name, City, a coach. Each team is uniquely identified by using Teamid. A team can have many Players and a captain. Each player is uniquely identified by Playerid, having a Name, and multiple phone numbers,age. A player represents only one team. There are many Stadiums to conduct matches. Each stadium is identified using Stadiumid, having a stadium_name,Address (involves city,area_name,pincode).A team can play many matches. Each match played between the two teams in the scheduled date and time in the predefined Stadium. Each match is identified uniquely by using Matchid. Each match won by any of the one team that also wants to record in the database. For each match man_of_the match award given to a player.

Queries:

- 1 Display the youngest player (in terms of age) Name, Team

- name , age in which he belongs of the tournament.
- 2 List the details of the stadium where the maximum number of matches were played.
- 3 List the details of the player who is not a captain but got the man_of_match award at least in two matches.
- 4 Display the Team details who won the maximum matches.
- 5 Display the team name where all its won matches played in the same stadium.

III. Consider the following Scenario and design an ER-Diagram, map the designed ER-diagram into a Relational model.

Consider an organization “ABC” having many employees. An employee works for one department. Each employee identified by using Empid, having Name, address (described as House_no, city, district, state, pin code) and more than one phone numbers. Department identified by using Dno, having Dname, Dlocation. Each Department having a manager . Each department having many employees. There are many Projects , each project is controlled by the department. Each Project uniquely identified by Pno, having Project name, Project location. An employee works on many Projects. Number of hours per week worked on each project by an Employee also needs to be recorded in the database . A project is worked by many employees. Each employee supervised by the supervisor. Employee having many dependents. Dependents having the dependent_name, gender, age, address. Dependents are identified by Empid.

T1(Empid, Emp_Name,city, district, state, pin_code, phoneno, Dno,Dname,Dlocation, Dept_mgr_id, Pno, Project_name, Project_location, Number_of_Hours,Supervisor_Empid, Dependent_name, gender, address)

Deduce the above Relation T1 into the 3NF and then solve the following queries.

Queries:

1. Display the details of the employees who are working on both the projects having project_no 5 and 10.
2. Display the details of employees having atleast two dependents.
3. Display the project name on which more number of employees are working.

4. Retrieve the employees who do not have any dependents.
 5. Display the Employee details whose total number of hours per week working on various projects is maximum than all other employees.
 6. Create a view to display the number of employees working in each department.
- IV Design an ER-diagram for the following scenario, Convert the same into a relational model, normalize Relations into a suitable Normal form and then solve the following queries.

A country can have many Tourist places . Each Tourist place is identified by using tourist_place_id, having a name, belongs to a state, Number of kilometers away from the capital city of that state,history. There are many Tourists visits tourist places every year. Each tourist is identified uniquely by using Tourist_id, having a Name, age, Country and multiple emailids. A tourist visits many Tourist places, it is also required to record the visted_date in the database. A tourist can visit a Tourist place many times at different dates. A Tourist place can be visited by many tourists either in the same date or at different dates.

Queries:

- 1 List the state name which is having maximum number of tourist places.
 - 2 List details of Tourist place where maximum number of tourists visited.
 - 3 List the details of tourists visited all tourist places of the state “KARNATAKA”.
 - 4 Display the details of the tourists visited at least one tourist place of the state, but visited all states tourist places.
 - 5 Display the details of the tourist place visited by the tourists of all country.
- V Design an ER-diagram for the following scenario, Convert the same into a relational model, normalize Relations into a suitable Normal form and then solve the following queries.

A country wants to conduct an election for the parliament. A country having many constituencies. Each constituency is identified uniquely by Constituency_id, having the Name, belongs to a state,Number_of_voters. A constituency can have many voters. Each voter is uniquely identified by using Voter_id, having the Name, age,

address (involves Houseno,city,state,pincode). Each voter belongs to only one constituency. There are many candidates contesting in the election. Each candidates are uniquely identified by using candidate_id, having Name, phone_no, age, state. A candidate belongs to only one party. There are many parties. Each party is uniquely identified by using Party_id, having Party_Name, Party_symbol. A candidate can contest from many constituencies under a same party. A party can have many candidates contesting from different constituencies. No constituency having the candidates from the same party. A constituency can have many contesting candidates belongs to different parties. Each voter votes only one candidate of his/her constituency.

Queries:

- 1 List the details of the candidates who are contesting from more than one constituencies which are belongs to different states.
- 2 Display the state name having maximum number of constituencies.
- 3 Create a stored procedure to insert the tuple into the voter table by checking the voter age. If voter's age is at least 18 years old, then insert the tuple into the voter else display the "Not an eligible voter msg".
- 4 Create a stored procedure to display the number_of_voters in the specified constituency. Where the constituency name is passed as an argument to the stored procedure.
- 5 Create a TRIGGER to UPDATE the count of "Number_of_voters" of the respective constituency in "CONSTITUENCY" table, AFTER inserting a tuple into the "VOTERS" table.

Note 1: In the practical Examination each student has to pick one question from a lot of all the 5 questions.

Note 2: Change of program is not permitted in the Practical Examination.

PYTHON PROGRAMMING LAB

Semester	III	CIE Marks : 40
Course Code	18MCA37	SEE Marks : 60
Contact Periods (L:T:P)	0-1-3	Exam Hours : 03
Credits : 02		

Course learning objectives:

This course (18MCA37) will enable student to

- Use basic primitives and constructs in a python program.
- Apply OOPS concepts to develop python programs.
- Learn how to create GUI and solve real-world problems.

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Apply object-oriented programming concepts to develop dynamic interactive Python applications.

CO2: Use the procedural statements: assignments, conditional statements, loops, method calls and arrays

CO3: Design, code, and test small Python programs with a basic understanding of top-down design.

CO4: Learn how to create GUI and solve real-world problem using language idioms, data structures and standard library

Laboratory Experiments:

1. Write a program to sum all the elements from n1 to n2 where n1 and n2 are positive integers
2. Input an array of n numbers and find separately the sum of positive numbers and negative numbers.
3. Write a program to search an element using linear search
4. Write a program to search an element using binary search.
5. Write a program to simulate stack.
6. Using a stack evaluate an arithmetic expression.
7. Write a program to multiply two matrices.
8. Write a program to find the roots of a quadratic equation
9. Write a program to Insert a number in a sorted array.
10. Write a Python Program to check whether the given string is palindrome or not using built in string manipulation methods.
11. Write a Python Program to read a word and prints the number of letters, vowels and percentage of vowels in the word using dictionary

12. Write a Python Program to check a given sentence is a pangram or not using function/Module.
13. Write a Python Event driven Program for file operations Press 1: to open file in read mode 2: open the file in write mode 3: current position of the file pointer #4: Reposition the pointer at the beginning 5: exit.
14. Write an Object oriented Python program to create two Time objects: currentTime, which contains the current time; and breadTime, which contains the amount of time it takes for a bread maker to make bread. Then we'll use addTime to figure out when the bread will be done. Write the printTime function to display the time when the bread will be done by the bread maker.

Note 1: In the practical Examination each student has to pick one question from a lot of all 14 questions.

Note 2: Change of program is not permitted in the Practical Examination.

ALGORITHMS LAB

Semester	III	CIE Marks : 40
Course Code	18MCA38	SEE Marks : 60
Contact Periods (L:T:P)	0-1-3	Exam Hours : 03
Credits : 02		

Course learning objectives:

This course(18MCA38) will enable student to:

- Understand the concepts of time complexity and space complexity practically
- Design algorithms using different techniques/methods for the same problem
- Know the importance of recurrence relation in designing algorithms

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Implement the concepts of time and space complexity, divide-and-conquer strategy, dynamic programming, greedy and approximate algorithms.

CO2: Describe the methodologies of how to analyze an algorithm

CO3: Choose a better algorithm to solve the problems.

- 1 Implement Recursive Binary search and Linear search and determine the time required to search an element. Repeat the experiment for different values of n, the number of elements in the list to be searched and plot a graph of the time taken versus n.
- 2 Sort a given set of elements using the Insertion sort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.
- 3 Sort a given set of elements using Merge sort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.
- 4 Obtain the Topological ordering of vertices in a given graph.
- 5 Implement 0/1 Knapsack problem using dynamic programming.
- 6 From a given vertex in a weighted connected graph, find

- shortest paths to other vertices using Dijkstra's algorithm.
- 7 Sort a given set of elements using Quick sort method and determine the time required sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.
 - 8 Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm.
 - 9 Check whether a given graph is connected or not using DFS method.
 - 10 Find a subset of a given set $S = \{s_1, s_2, \dots, s_n\}$ of n positive integers whose sum is equal to a given positive integer d. For example, if $S = \{1, 2, 5, 6, 8\}$ and $d = 9$ there are two solutions $\{1, 2, 6\}$ and $\{1, 8\}$. A suitable message is to be displayed if the given problem instance doesn't have a solution.
 - 11 a). Implement Horspool algorithm for String Matching.
b). Find the Binomial Co-efficient using Dynamic Programming.
 - 12 Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.
 - 13 a). Implement Floyd's algorithm for the All-Pairs- Shortest-Paths Problem.
b). Compute the transitive closure of a given directed graph using Warshall's algorithm.
 - 14 Implement N Queen's problem using Back Tracking.

Note 1: In the practical Examination student has to execute one program from a lot of all the 14 questions.

Note 2: Change of program is not permitted in the Practical Examination.

SOFTWARE TESTING

Semester	III	CIE Marks : 40
Course Code	18MCA351	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA351) will enable student to

- Know the essence of software testing and Debugging
- Study various types of testing
- Build ability to generate new test cases

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Acquire knowledge of basic principles and knowledge of software testing and Debugging and test cases.

CO2: Understand the perceptions on testing like levels of testing, generalized pseudo code and with related examples

CO3: Study the various types of testing.

CO4: Analyze the difference between functional testing and structural testing.

CO5: Analyze the performance of fault based testing.

MODULE -1:

Basics of Software Testing, Basic Principles, Test case selection and Adequacy

Humans, Errors and Testing, Software Quality; Requirements, Behavior and Correctness, Correctness Vs Reliability; Testing and Debugging; Test Metrics; Software and Hardware Testing; Testing and Verification; Defect Management; Execution History; Test Generation Strategies; Static Testing; Test Generation from Predicates. Sensitivity, Redundancy, Restriction, Partition, Visibility and Feedback, Test Specification and cases, Adequacy Criteria, Comparing Criteria

MODULE -2:

A perspective on Testing

Basic definitions, Test cases, Insights from a Venn diagram, Identifying test cases, Error and fault taxonomies, Level of testing, Examples: Generalized pseudo code, The triangle problem, the Next

Date function, The commission problem, The SATM (Simple Automation Teller Machine) problem, The currency converter, Saturn windshield wiper

MODULE -3:

Boundary value testing, Equivalence class testing, Decision table based testing

Boundary value analysis, Robustness testing, Worst-case testing, special value testing, Examples, Random testing, Equivalence classes, Equivalence test cases for triangle problem, Next Date function and commission problem, Guidelines and observations, Decision tables, Test cases for triangle problem

MODULE -4:

Path Testing, Data flow testing, Levels of Testing, Integration Testing

DD Paths, Test coverage metrics, Basis path testing, guidelines and observations, Definition Use testing, Slice based testing, Guidelines and observations. Traditional view of testing levels, Alternative life cycle models, the SATM systems, separating integration and system testing, Guidelines and observations.

MODULE -5:

Fault Based Testing, Planning and Monitoring the Process, Documenting Analysis and Test

Assumptions in fault-based testing, Mutation Analysis, Fault-based Adequacy Criteria; Variations on mutation Analysis; From Test case specification to Test Cases, Scaffolding, Generic vs. specific Scaffolding, Test Oracles, Self checks as oracles, Capture and Replay. Quality and Process, Test and Analysis strategies and plans, Risk Planning, Monitoring the Process, Improving the process, The quality team, Organizing documents, Test strategy document, Analysis and test plan, Test design specifications documents, Test and analysis reports.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the

topics under a Module.

- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Adithya P.Mathur “ Foundations of Software Testing – Fundamental Algorithms and Techniques”, Pearson Education India, 2011
2. Mauro Pezze, Michael Young, Software testing and Analysis- Process, Principles and Techniques, Wiley India, 2012
Paul C Jorgensen, “Software Testing A Craftsman's Approach”, Auerbach publications, 3rd edition, 2011.

Reference Books:

1. KshirasagaraNaik, PriyadarshiTripathy: Software Testing and Quality Assurance, Wiley India 2012
2. M.G. Limaye: Software Testing-Principles, Techniques and Tools – McGraw Hill, 2009

OPTIMIZATION TECHNIQUES

Semester	III	CIE Marks : 40
Course Code	18MCA352	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA352) will enable student to

- Know the concept of optimization and different optimization techniques
- Design optimized algorithms/solutions for real world problems using various techniques

Course Outcomes (CO): At the end of this course, the students will be able to

CO1: Understand the importance of operations research & acquire skills to develop linear programming mathematical models to real world problems

CO2: Understand the essence and foundations of the simplex algorithm and write the dual of the given primal problems.

CO3: Recognize, formulate and giving optimal solution to a Transportation problem and Assignment problems.

CO4: Analyse & Solve Simple Game Theory Problems.

CO5: Solve problems of job sequencing of production runs

MODULE-1:

Introduction

Operations Research-A quantitative approaches to Decision Making, Features of OR, OR Approaches to Problem Solving, Methodology of Operations Research.

Linear programming: Introduction, Structure of Linear Programming Model, Advantages, General Mathematical model of LPP, Examples of LP Model Formulation, Graphical solution methods of LP problem.

MODULE-2 :

Linear Programming: The Simplex Method, Two-Phase method, Big M method

MODULE-3:

Duality in Linear Programming, Formulation of Dual Linear Programming Problem and Examples.

Assignment Problem: Mathematical model of Assignment Problem, Hungarian method for solving Assignment problem.

MODULE-4:

Transportation Problem:

Transportation Problem, Mathematical model of Transportation Problem, Methods of finding Initial solution (North west corner rule, Least cost method, Vogel's Approximation method), Test for Optimality in TP using MODI method (uv -method).

MODULE-5 :

Theory of Games: Introduction, Two-person zero-sum game, pure strategies (Minmax and Maxmin principles), Mixed strategies, The rules principles of Dominance, Algebraic method to solve games without saddle point, Graphical method to solve the games.

Sequencing Problems: Processing n jobs through two machines (Johnson's Procedure)

Question Paper Pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbook:

1. Operations Research Theory and Applications By JK Sharma, 5th Edition, MACMILLAN publishers India, (Chapter1,2,3,4,5,9,10,11,12,20)

Reference Books:

1. Operation Research, S. D. Sharma - Kedarnath Ramnath and Co, 2002

2. Operations Research – An Introduction, Taha H.A. –Low price Edition, 7th Edn,2006
3. Introduction to Operation Research, Hiller and Liberman, Mc Graw Hill. 5th edition 2001
4. Operation Research, Prem Kumar Gupta, D S Hira,S Chand pub, New Delhi, 2007.

ADVANCE COMPUTER NETWORKS

Semester	III	CIE Marks : 40
Course Code	18MCA353	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA353) will enable student to

- Enhance the understanding of TCP-IP architecture
- Understand the concepts of protocols, network interfaces, and design/performance issues
- Understand and apply TCP-IP over wired and wireless networks

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Understand the terminology and concepts of TCP-IP reference model and IPV6 message format and its services.

CO2: Acquire the concepts of protocols, network interfaces, and design/performance issues in Local Area Networks and wide area networks.

CO3: Analyze the difference between wireless networks and satellite network.

CO4: Evaluate the performance of TCP/IP over asymmetric networks

MODULE 1:

Introduction to Computer Networks

History of TCP/IP, TCP Applications and Services, Performance Study of TCP/IP, earning of TCP Performance? TCP, TCP Services , Header Format, Encapsulation in IP, Acknowledgment Mechanism, Retransmission Mechanism, Connection Establishment and Termination, Control and Sliding Window, Congestion Control ,UDP ,UDP Services, Header Format ,Encapsulation in IP,IP Services, Fragmentation and Reassembly , Header Format and IP Version 6, Reasons for Network Measurement, Measurement Tasks, Classification of Measurement Tools, Popular Measurement Tools and Their Applications, Tcpcdump, Tcpstat, Ttcp & Netperf.

Distributed Benchmark System.

MODULE 2:

TCP/IP Network Simulation and TCP Modeling

The Role of Simulation, Steps of a Systematic Simulation Study, Types of Simulations, Continuous versus Discrete Event, Terminating versus Steady State, Synthetic versus Trace Driven Simulation, Simulation Validation and Verification, Confidence Level of Simulation Results, Confidence Level Formula, Terminating Simulation, Steady-State Simulation, Common Simulation Mistakes, Simulation with Self-Similar Traffic Network Simulators: Model Construction and Parameter Setting Data Collection, Simulation Execution, Presentation of Results and Examples of TCP/IP Simulation. Motivation for Mathematical Modeling of TCP, Essentials of TCP Modeling, Window Dynamics Packet-Loss Process, Gallery of TCP Models, Periodic Model, Detailed Packet Loss Model, Stochastic Model with General Loss Process, Control System Model and Network System Model.

MODULE 3:

TCP/IP Performance over Wireless Networks & TCP/IP Performance over Mobile Networks & Optical Networks Layer

Wireless Networks: Generic Characteristics, Wireless Local Area Networks and Cellular Communications Networks. TCP Performance Issues over Wireless Links, Inappropriate Reduction of Congestion Window, Throughput Loss in WLANs and Throughput Loss in Cellular Communication Systems. Improving TCP Performance over Wireless Links: Splitting TCP Connections, Snooping TCP at Base Stations, Notifying the Causes of Packet Loss, Adding Selective Acknowledgments to TCP and Comparison of Enhancement Schemes.

Wireless System Evolution and TCP/IP: Trends in Cellular Communication Systems, Trends in Wireless LAN Systems, TCP/IP over Heterogeneous Wireless systems.

Cellular and Ad Hoc Networks: TCP Performance in Cellular Networks, Mobile IP, Impact of Mobility on TCP Performance, Approaches to Improve TCP Performance, TCP Performance in Ad

Hoc Networks, Dynamic Source Routing, Impact of Mobility on TCP Performance, Approaches to Improve TCP Performance. Evolution of Optical Networks, IP over DWDM, Multiprotocol Label Switching, Multiprotocol Lambda Switching, Optical Burst Switching, Optical Packet Switching: Optical Packet Format, Congestion Resolution in Optical Packet Switches, Performance of TCP/IP over Optical Networks, Optical Packet Network End-to-End Performance, Mapping of TCP in Optical Packets, Optical Packet Design in the TCP/IP Environment.

MODULE 4:

TCP/IP Performance over Satellite Networks & TCP/IP Performance over Asymmetric Networks

A Brief History of Data Satellites, Motivations for Using Satellites, Types of Satellites Satellite Internet Architectures, Satellite Characteristics Affecting TCP: Long Feedback Loop, Link Impairment, Bandwidth-Delay Product, Bandwidth Asymmetry, Variable Delays, LEO Handoff Spectral Congestion, Security. TCP Enhancements for Satellite Networks: Path MTU Discovery, TCP for Transactions, Window Scaling, Large Initial Window, Byte Counting, Delayed ACKs after Slow Start, Explicit Congestion Notification, Multiple Connections, Pacing TCP Segments, TCP/IP Header Compression, and Security Issues Conclusions for TCP Enhancements. Advanced Enhancements and New Versions of TCP: Quick-Start TCP, High Speed TCP, TCP Peach, Explicit Transport Error Notification TCP Westwood and XCP. New Transport Protocols for Satellite Links: Satellite Transport Protocol, Space Communications Protocol Specifications-Transport Protocol. Types of Network Asymmetry: Bandwidth Asymmetry, Media-Access Asymmetry, Loss Rate. Asymmetry Impact of Asymmetry on TCP performance: Bandwidth Asymmetry, Media Access Asymmetry. Improving TCP Performance over Asymmetric Networks: Uplink Bandwidth Management Handling Infrequent ACK. Experimental Evaluation Of Performance Improvement Techniques Experiments with Bandwidth Asymmetry, Experiments with Media Access Asymmetry

MODULE 5:

TCP/IP Performance over Asymmetric Networks & New TCP Standards and Flavors

Types of Network Asymmetry: Bandwidth Asymmetry, Media-Access Asymmetry, Loss Rate. Asymmetry Impact of Asymmetry on TCP Performance: Bandwidth Asymmetry, Media Access Asymmetry. Improving TCP Performance over Asymmetric Networks: KS: Uplink Bandwidth Management Handling Infrequent ACK. Experimental Evaluation of Performance Improvement Techniques Experiments with Bandwidth Asymmetry, Experiments with Media Access Asymmetry. Duplicate Acknowledgments and Fast Retransmit, Fast Recovery and TCP Reno, TCP New Reno, TCP with Selective Acknowledgments, Forward Acknowledgments, TCP Vegas ,Overview of Other Features and Options and Performance Comparison of TCP Flavors.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbook:

1. High Performance TCP/IP: Networking Concepts, Issues, and Solutions, Mahbub Hassan and Raj Jain, IST Edition, 2009 PHI Learning Chapters 1,2,3,4,5,6,7,8,9,10,11,12,13 (excluding those topics which are not in the syllabus)

Reference Books:

1. TCP/IP Illustrated (Volume I, Volume II and Volume III), W. Richard Stevens, Addison-Wesley

MANAGEMENT INFORMATION SYSTEMS

Semester	III	CIE Marks : 40
Course Code	18MCA354	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA354) will enable student to

- Provide a clear understanding of Management Information Systems in Real world.
- Apply information communication technologies to solve business problems
- Describe Lifecycle Components of Enterprise Resource Planning, Components, and Challenges.

Course Outcome (CO):

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

CO1: Recognize the roles and uses of technology in business systems, operations and describe organizational structure .

CO2: Equip the students with preliminaries of technologies used in business information systems.

CO3: Familiarize students with the Business applications and E-Commerce initiatives.

MODULE -1:

Introduction to MIS

MIS: Concept, Definition, Role of the Management Information System, Impact of the Management Information System, MIS and User, Management as Control System, A support to the Management, Management effectiveness and Organization Effectiveness.

Strategic Management of Business Performance:

The Concept of Corporate Planning, Essentiality of Strategic Planning, Development of the Business Strategies, Types of Strategies, Tools of Planning, Balance Score Card, Score card and Dash Board.

MODULE -2:

Decision-Making

Decision-Making Concepts, Decision-Making Process, Decision Analysis by Analytical Modeling, Behavioural Concepts in Decision-Making, Organizational Decision-Making, MIS and Decision-Modeling. (Case study)

Information, Knowledge, Business Intelligence:

Information Concepts, Information: A Quality Product, Classification of the Information, Methods of Data and Information Collection, Value of the information, Business intelligence.

MODULE -3:

Technology of Information Systems:

Introduction, Data Processing, Transaction Processing, Application Processing, Information System Processing, OLAP(on-line analytical Processing) for Analytical Information, TQM(Total Quality Management) of Information System. Human Factors and User Interface, Evaluation and feasibility of IT Solutions.

MODULE -4:

E-Business Technology

Introduction to E-Business, Models of E-Business: Business to Business(B2B), Business to Customer(B2C), Customer to Business(C2B), Customer to Customer (C2C), Internet and World Wide Web(WWW), Internet/Extranet, Security in E-Business, Electronic Payment Systems, Impact of Web on Strategic Management. Enterprise: Content Management System(CMS).

MODULE -5:

E-commerce

Introduction to E-Commerce Technologies: Client-side programming: Important factors in client side, Web page design and production, Overview of HTML, Basic text formatting, Links, Images, Tables, Frames, form server side programming-I: Servlet fundamentals, Server-side Programming II: Database connectivity: Relational database systems, JDBC perspectives and JDBC program example. Business-oriented e-commerce: Feature of B2B e-Commerce, Business models, Integration E-services: Categories of

e-services, Web-enabled services, Matchmaking services.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each MODULE

Textbooks:

1. Waman S Jawadekar: Management Information System, 4th Edition, Tata McGraw Hill. (Chapters: 1,3,6,7,16,20)
2. Henry Chan, Raymond Lee, Tharam Dillon, Elizabeth Chang, E Commerce, Fundamentals & Applications, John Wiley & Sons, 2003. Part I & Part II. (chapters: 3, 4, 5, 12, 13)

Reference Books:

1. Ralph M Stair and George W Reynolds: Principles of Information Systems, 7th Edition, Thomson, 2010.
2. Steven Alter: Information Systems - The Foundation of E-Business, 4th Edition, Pearson Education, 2001.
3. Rahul De, Managing Information Systems in Business, Government and Society, Wiley India, 2012.

SEMESTER – IV ADVANCE JAVA PROGRAMMING

Semester	IV	CIE Marks : 40
Course Code	18MCA41	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA41) will enable student to

- Learn the concept of Servlet, JSP and its services.
- Learn how to create packages and interfaces
- Build Database connection
- Develop Enterprise Java Bean Applications

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Learn the concept of Servlet and its life cycle

CO2: Understand JSP tags and its services

CO3: Create packages and interfaces

CO4: Build Database connection

CO5: Develop Java Server Pages applications using JSP Tags.

CO6: Develop Enterprise Java Bean Applications

MODULE 1:

Servlet Structure, Servlet packaging, HTML building utilities, Lifecycle, SingleThreadModel interface, Handling Client Request: Form Data, Handling Client Request: HTTP Request Headers. Generating server Response: HTTP Status codes, Generating server Response: HTTP Response Headers, Handling Cookies, Session Tracking.

MODULE 2:

Introduction to JSP

Overview of JSP: JSP Technology, Need of JSP, Benefits of JSP, Advantages of JSP, Basic syntax, Invoking java code with JSP scripting elements, creating Template Text, Invoking java code from JSP, Limiting java code in JSP, using jsp expressions, comparing servlets and jsp, writing scriptlets. For example Using Scriptlets to make parts of jsp conditional, using declarations, declaration example.

MODULE 3:

Controlling the Structure of generated servlets and Java Beans

Controlling the Structure of generated servlets: The JSP page directive, import attribute, session attribute, isEligible attribute, buffer and autoflush attributes, info attribute, errorPage and isErrorPage attributes, isThreadSafe Attribute, extends attribute, language attribute, Including files and applets in jsp Pages, using java beans components in JSP documents

JAR files, manifest file, Working with Java Beans. Introspection, Customizers, Bean properties: Simple properties, Design Pattern events, creating bound properties, Bean Methods, Bean info class, Persistence.

MODULE 4:

Annotations and JDBC

Annotations: Built-in Annotations with examples, Custom Annotation.

Talking to Database, Immediate Solutions, Essential JDBC program, using prepared Statement Object, Interactive SQL tool. JDBC in Action Result sets, Batch updates, Mapping, Basic JDBC data types, Advanced JDBC data types, immediate solutions.

MODULE 5:

EJB and Server Side Component Models

Introduction to EJB: The Problem domain, Breakup responsibilities, Code Smart not hard, the Enterprise java bean specification, Components Types.

Server Side Component Types: Session Beans, MessageDriven Beans, Entity Beans, The Java Persistence Model. Container services: Dependency Injection, Concurrency, Instance pooling n caching, Transactions, security, Timers, Naming and object stores, Interoperability, Life Cycle Callbacks, Interceptors, platform integration. Developing your first EJB, Models: The Stateless Session Bean, the Stateful Session Bean, the Singleton Session Bean, Message-Driven Beans. EJB and PERSISTENCE. Persistence Entity manager Mapping Persistence objects, Entity Relationships.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Textbooks:

1. Marty Hall, Larry Brown. Core Servlets and Java Server Pages. Volume 1: Core Technologies. 2nd Edition. (Chapter 3,4,5,6,7,8,9,10,11,12,13,14).
2. Java 6 Programming Black Book, Dreamtech Press. 2012 (Chapter 17,18,19,20,21,22,27,28,29,30).
3. Andrew LeeRubinger, Bill Burke. Developing Enterprise Java Components. Enterprise JavaBeans 3.1.O'reilly. (Chapter 1,2,3,4,5,6,7,8,9,10,11).

Reference Books:

1. Michael Sikora, EJB 3 Developer Guide, A practical guide for developers and architects to the Enterprise Java Beans Standard, Shroff Publishers & Distributors PVT LTD. July 2008.
2. Herbert Schildt, Java The Complete Reference, 8th Edition. Comprehensive coverage of the Java Language. Tata McGraw-Hill Edition – 2011.

ADVANCED WEB PROGRAMMING

Semester	IV	CIE Marks : 40
Course Code	18MCA42	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA42) will enable student to

- Gain an understanding of building Web Applications using PHP, Ruby, Bootstrap, AJAX and XML.
- Provide a comprehensive introduction to Rich Internet Web Applications.
- Learn how to build responsive Web Applications.

Course Outcomes (CO): At the end of this course, the students will be able to:

CO1: Acquire knowledge of building the Web Applications using PHP, Ruby, Bootstrap, AJAX and XML.

CO2: Design the Asynchronous Web Applications using AJAX.

CO3: Understand the terminology of building Web Applications using MVC architecture.

CO4: Design responsive web applications using Bootstrap.

MODULE 1:

Introduction to PHP

Origins and uses of PHP, Overview of PHP, General syntactic characteristics, Primitives, operations and expressions, Output, Control statements, Arrays, Functions, Pattern matching.

Building Web applications with PHP

Form handling, Files, Tracking users, cookies, sessions, Using databases, Handling XML.

MODULE 2:

Introduction to Ruby and Introduction to Rails

Origins and uses of Ruby, Scalar types and their operations, Simple input and output, Control statements, Arrays, Hashes, Methods, Classes, Codeblocks and iterates, Pattern matching.

Overview of Rails, Document requests, Processing forms, Layouts. Rails applications with Databases.

MODULE 3:

Rich Internet Applications With Ajax: Limitations of Classic Web application model, AJAX principles, Technologies behind AJAX, Examples of usage of AJAX; Asynchronous communication and AJAX application model.

Ajax with XMLHttpRequest object: Part 1

Creating Ajax Applications: An example, Analysis of example ajax.html, Creating the JavaScript, Creating and opening the XMLHttpRequest object, Data download, Displaying the fetched data, Connecting to the server, Adding Server-side programming, Sending data to the server using GET and POST.

MODULE 4:

Ajax with XMLHttpRequest object: Part 2

Handling multiple XMLHttpRequest objects in the same page, Using two XMLHttpRequest objects, Using an array of XMLHttpRequest objects, AJAX Patterns – Predictive Fetch, Multi-stage download, Periodic Refresh and Fallback patterns, Submission throttling.

MODULE 5:

Introduction to Bootstrap.

What Is Bootstrap? Bootstrap File Structure, Basic HTML Template, Global Styles, Default Grid System, Basic Grid HTML, Offsetting Columns, Nesting Columns, Fluid Grid System, Container Layouts, Responsive Design. Typography, Emphasis Classes, Lists, Code, Tables, Optional Table Classes, Table Row Classes, Forms, Buttons, Images, Icons.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions from each Module.

- Each question will have questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each module

Textbooks:

1. RobertW.Sebesta: Programming the Worldwide Web, 4th Edn, Pearson, 2012
2. Professional AJAX – Nicholas C Zakas et al, Wrox publications, 2008.
3. Steven Holzner: Ajax: A Beginner's Guide, Tata McGraw Hill, 2014.
4. Jake Spurlock: "Bootstrap: Responsive Web Development", O'Reilly Media, 2014.

Reference Books:

1. Thomas A. Powel: Ajax The Complete reference, McGraw Hill, 2008.
2. Aravind Shenoy, Ulrich Sossou: Learning Bootstrap, Packt, Dec 2014.
3. Dana Moore, Raymond Budd, Edward Benson: Professional Rich Internet Applications: AJAX and Beyond, Wiley 2012.

OBJECT ORIENTED MODELING AND DESIGN

Semester	IV	CIE Marks : 40
Course Code	18MCA43	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA43) will enable student to

- Understand Basic UML Concepts and terminologies and Modeling concepts.
- Learn the basic principles of Software modeling and apply them in real world applications
- Develop Object Oriented Software models in terms of Static Dynamic behavior.

Course Outcomes (CO): At the end of this course, the students will be able to

CO1: Acquire knowledge of

- Basic UML Concepts and terminologies
- Life Cycle of Object oriented Development
- Modeling Concepts

CO2: Identify the basic principles of Software modeling and apply them in real world applications

CO3: Produce conceptual models for solving operational problems in software and IT environment using UML

CO4: Analyze the development of Object Oriented Software models in terms of Static behaviour

- Dynamic behaviour

CO5: Evaluate and implement various Design patterns

MODULE -1:

Modeling Concepts & Class Modeling

What is Object Orientation? What is OO development? OO themes; Evidence for usefulness of OO development; OO modeling history. Modeling as Design Technique: Modeling; abstraction; The three models.

Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models; Practical tips. Advanced object and class concepts; Association ends; N-array associations; Aggregation; Abstract

classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages;

MODULE -2:

State Modeling and Interaction Modeling

State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior; Practical tips. Advanced State Modeling: Nested state diagrams; Nested states; Signal generalization; Concurrency; A sample state model; Relation of class and state models;

Interaction Modeling: Use case models; Sequence models; Activity models. Use case relationships; Procedural sequence models; Special constructs for activity models

MODULE -3:

System Conception and Analysis

System Conception: Devising a system concept; elaborating a concept; preparing a problem statement.

Overview of analysis; Domain class model; Domain state model; Domain interaction model; Iterating the analysis. Application Analysis: Application interaction model; Application class model; Application state model; adding operations.

MODULE -4:

System Design and Class Design

Overview of system design; Estimating performance; Making a reuse plan; Breaking a system in to sub-systems; Identifying concurrency; Allocation of sub-systems; Management of data storage; Handling global resources; Choosing a software control strategy; Handling boundary conditions

Class Design: Overview of class design; Bridging the gap; Realizing use cases; Designing algorithms; Recording downwards, Refactoring; Design optimization; Reification of behavior.

MODULE -5:

Patterns and Design Patterns

What is a pattern and what makes a pattern? Pattern categories; Relationships between patterns; Pattern description.

Introduction, Model View Controller, Structural decomposition:

Whole-Part, Access Control: Proxy; Management Patterns: Command processor; Communication Patterns: Forwarder-Receiver; Client-Dispatcher-Server; Publisher-Subscriber

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Michael Blaha, James Rumbaugh, "Object-Oriented Modeling and Design with UML", 2nd Edition, Pearson Education / PHI, 2005. (Chapters 1 to 9, 11 to 14.10, 15.1 to 15.8)
2. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, "Pattern-Oriented Software Architecture, A System of Patterns", Volume 1, John Wiley and Sons, 2006. (Chapters 1, 2.4, 3)

Reference Books:

1. Grady Booch et al, "Object-Oriented Analysis and Design with Applications", 3rd Edition, Pearson, 2007.
2. Mark Priestley, "Practical Object-Oriented Design with UML", 2nd Edition, Tata McGraw-Hill, 2003.
3. K. Barclay, J. Savage, "Object-Oriented Design with UML and JAVA", Elsevier, 2008.
4. Booch, G., Rumbaugh, J., and Jacobson, I., "The Unified Modeling Language User Guide", 2nd Edition, Pearson, 2005.
5. E. Gamma, R. Helm, R. Johnson, J. Vlissides, "Design Patterns-Elements of Reusable Object- Oriented Software", Addison-Wesley, 1995.
6. Michael R Blaha, James R Rumbaugh, "Object Oriented Modeling and Design with UML", 2nd Edition, Prentice Hall, 2004

PROFESSIONAL COMMUNICATION & REPORT WRITING

Semester	:	IV	CIE Marks	: 40
Course Code	:	18MCA46	SEE Marks	: 60
Contact Periods (L:T:P)	:	2-0-0	Exam Hours	: 03
Credits : 02				

Course learning objectives:

This course (18MCA46) will enable student to

- Learn the role of communication and standard for grammar, numbers and spellings
- Realize the business Etiquettes and organizing the reports.
- Understand the aspects of group communication.
- Exposure to the various business and technological ethics.

Course Outcomes (CO):

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

CO1: Understand the professional communication at work place.

CO2: Acquire the knowledge technical writing and business reporting.

CO3: Develop the leadership qualities.

CO4: Understand and implement ethical behavior at work place.

MODULE 1:

Communication in the workplace- Role of communication in Business, Process of Human Communication. Informal Communication- Listening, Non verbal Communication. Correctness Of Communication- Nature of correctness, Standard for Punctuation, Standards for Grammar, Standards for the use of numbers, Spelling, Capitalization

MODULE 2:

Writing for Effect- Business Etiquette, conversational style, view point, positive language, and courtesy. Basics Report Writing- Determine the Factors, Gathering the Information, Organizing the Report, Writing The Report. Physical Presentation Of Letters, Memos, And Reports- document preparation, Form of Business Letters. Strategies in the job search Process- Preparing application, Resume, Cover letter, Facing an Interview.

MODULE 3:

Group Communication- Introduction, Group discussion, Organizational group discussion, group discussion as part of selection process meetings. Effective Presentation strategies- Introduction, Defining purpose, Analyzing audience and place, organizing contents, preparing outline, Visual Aids, Understanding Nuances of delivery, Kinesics, Proxemics, Paralinguistics, Chronemics, Sample speech.

MODULE 4:

Motivation- Motivation and Motivators, Motivation: The Carrot and the stick, The Hierarchy of needs Theory, The Motivation – Hygiene Approach to Motivation. Leadership- Defining Leadership, Ingredients of Leadership, Trait Approaches to Leadership, Leadership behavior and styles, Situational or contingency, Approaches to Leadership.

MODULE 5:

Ethics: An overview of Ethics- What are Ethics? Ethics in the business World, Ethics in Information Technology (IT). Ethics for IT Professionals and IT users- IT professionals, The Ethical behavior of IT professionals, IT Users.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions from each Module.
- Each question will have questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each module

Textbooks

1. Meenakshi Raman and Sangeeta Sharma: Technical Communication-Principles and Practices, Oxford University Press, 2004.
2. George Reynolds: Ethics in Information Technology, 2nd Edition, Thomson Course Technology, 2007

Reference Books

1. Lesikar and Flatley: Communication-Basic Business

Communication Skills for Empowering the Internet Generation 9th Edition, Tata McGraw-Hill Edition, 2002

2. Harold Koonlz and Heinz Weihrich: Management-Essentials of Management, 9th Edition, McGraw-Hill International Edition, 2012

ADVANCED JAVA PROGRAMMING LAB

Semester	:	IV	CIE Marks	: 40
Course Code	:	18MCA47	SEE Marks	: 60
Contact Periods (L:T:P)	:	0-1-3	Exam Hours	: 03
Credits : 02				

Course Learning Objectives:

This course (18MCA47) will enable student to

- Design HTML pages using Java Servlets, JSP, Bean and EJB programs.
- Understand and demonstrate JSP with EJB application using business logic.

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Designing HTML pages to demonstrate Java Servlets, JSP, Bean and EJB programs.

CO2: Implementing Dynamic HTML using Servlet and demonstration of service methods, auto web page refresh, Session tracking using cookie and Http Session in Servlet.

CO3: Learn the fundamental of connecting to the database.

CO4: Demonstrate JSP (page attributes, action tags and all basic tags) and types of EJB application.

Program Statements

1. Write a JAVA Servlet Program to implement a dynamic HTML using Servlet (user name and Password should be accepted using HTML and displayed using a Servlet).
2. Write a JAVA Servlet Program to Auto Web Page Refresh (Consider a webpage which is displaying Date and time or stock market status. For all such type of pages, you would need to refresh your web page regularly; Java Servlet makes this job easy by providing refresh automatically after a given interval).
3. Write a JAVA Servlet Program to implement and demonstrate GET and POST methods (Using HTTP Servlet Class).
4. Write a JAVA Servlet Program using cookies to remember user preferences.
5. Write a JAVA Servlet program to track HttpSession by

accepting user name and password using HTML and display the profile page on successful login.

6. Write a JSP Program which uses jsp:include and jsp:forward action to display a Webpage.
7. Write a JSP Program which uses <jsp:plugin> tag to run an applet
8. Write a JSP Program to get student information through a HTML and create a JAVA Bean class, populate Bean and display the same information through another JSP
9. Write a JSP program to implement all the attributes of page directive tag.
10. Write a JAVA Program to insert data into Student DATA BASE and retrieve info based on particular queries (For example update, delete, search etc...).
11. An EJB application that demonstrates Session Bean (with appropriate business logic).
12. An EJB application that demonstrates MDB (with appropriate business logic).
13. An EJB application that demonstrates persistence (with appropriate business logic).

Note 1: In the practical Examination student has to execute one program from a lot of all the 13 questions.

Note 2: Change of program is not permitted in the Practical Examination.

ADVANCED WEB PROGRAMMING LAB

Semester	:	IV	CIE Marks : 40
Course Code	:	18MCA48	SEE Marks : 60
Contact Periods (L:T:P)	:	0-1-3	Exam Hours : 03
Credits : 02			

Course learning objectives: This course (18MCA48) will enable student to

- Learn the role of server side scripting languages.
- Develop the web application using PHP, Ruby, XML, Ajax
- Build the responsive web application using Bootstrap and store values in MYSQL.

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Understand, analyze and apply the role of server side scripting languages.

CO2: Build web application using PHP, Ruby, Bootstrap, XML and store values in MYSQL.

CO3: Build MVC based web applications using Ruby and Rails.

PART-A

1. Write a PHP program to process a file which contains English words, where each word is separated from the next word on a line by one space. The file is specified on the command line. The output of your program is a table in which the first column has unique words from the input file and the second column has the number of times the word appeared in the file; no word can appear twice in the table. Use two arrays to store the table, one for the words and one for the frequency values.
2. A file contains lines of employee data, where each line has name:age:department code:salary.
Write a PHP program to generate the following output:
 - i. The names of all the employee whose names end with “son”
 - ii. Percentage of employees under 40 years old
 - iii. Average salary of employees under 40 years old
 - iv. An alphabetical list of employees who are under 40 years old and who have salaries more than \$40,000.

3.
 - a) Write a PHP program to store current date-time in a COOKIE and display the ‘Last visited on’ date-time on the web page upon reopening of the same page.
 - b) Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.
4. Write an XHTML document to create a form with the following capabilities:
 - a. A text widget to collect the user’s name
 - b. B four checkboxes, one each for the following items
 - I. Four 100-watt light bulbs for \$2.39
 - II. Eight 100-watt light bulbs for \$4.29
 - III. Four 100-watt long-life light bulbs for \$3.95
 - IV. Eight 100-watt long-life light bulbs for \$7.49
 - c. A collection of three radio buttons they are labeled as follows:
 - I. Visa
 - II. MasterCard
 - III. Rupy

Write a PHP program that computes the total cost of the ordered light bulbs by a customer. The program must inform the buyer of exactly what was ordered in a table.

5. Write a PHP program to insert name and age information entered by the user into a table created using MySQL and to display the current contents of this table.
6. Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text fields. On submitting, store the values in MySQL table using PHP. Retrieve and display the data based on Name.
7. Write a PHP program to read student data from an XML file and store into the MySQL database. Retrieve and display.
8. Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text fields. On submitting, store the values in MySQL table using PHP. Provide buttons to update and delete data for the same.
9. Build a Rails application to accept book information and store into the date base. search a book with title specified by the and display the result with proper headings.

PART B

Develop a web application project using the languages and concepts learnt in the theory and exercises listed in part A with a good look and feel effects. You can use any web technologies and frameworks and databases.

Note (Part A : 40 Marks and Part B : 20 Marks):

1. In the examination student has to execute one question from part A which is evaluated for 40 Marks.
2. A team of maximum two students must develop the web application project. However during the examination, each student must demonstrate the project individually.
3. The team must submit a brief project report (25-30 pages) that must include the following
 - a. Introduction
 - b. Requirement Analysis
 - c. Software Requirement Specification
 - d. Analysis and Design
 - e. Implementation
 - f. Testing
 - g. Conclusion

OBJECT ORIENTED MODELING AND DESIGN LAB

Semester	:	IV	CIE Marks	: 40
Course Code	:	18MCA49	SEE Marks	: 60
Contact Periods (L:T:P)	:	0-1-3	Exam Hours	: 03
Credits : 02				

Course learning objectives:

This course (18MCA49) will enable student to

- Develop object oriented design model with the help of modern tool, Rational software Architect
- Implement Various design patterns for applicability, reasonableness, and relation to other design criteria.

Course Outcomes (CO):

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

CO1: Understand the fundamental principles of Object-Oriented analysis, design, development and programming

CO2: Demonstrate and represent the UML model elements, to enable visual representation of the system being developed

CO3: Implement object oriented design model with the help of modern tool, Rational software Architect

CO4: Analyze and differentiate the static and dynamic behavior of the system for achieving the intended functionalities of the system

CO5: Evaluate Various design patterns for applicability, reasonableness, and relation to other design criteria.

Laboratory Experiments:

The student has to draw the necessary UML diagrams using any suitable UML drawing tool and implement a program in Java or C++ or C# to demonstrate the design pattern specified by the examiner. For analysis and design models -, Class Diagram, Use-Case, Sequence diagrams should be drawn.

NOTE: ANY SUPPORTING TOOL MAY BE USED.

1. Publisher-Subscriber: Define a one-to-many dependency between objects so that when one object changes state all its dependents are notified and updated automatically. The Observer pattern is also known as Dependents, Publisher-Subscriber. The need

to maintain consistency between related objects without making classes tightly coupled.

2. Command Processor: The command processor design pattern separates the request for Service from its execution. A command processor component manages request as separate objects, schedules their execution and provides additional service.

3. Forwarder-Receiver: Distributed peers collaborate to solve a particular problem. A peer may act as a Client, requesting services, as a server, providing services, or both. The forwarder-receiver design pattern provides transparent inter process communication for software system with a peer-to-peer interaction model. It introduces forwarder and receiver to decouple peers from the underlying communication mechanism.

4. Client-Dispatcher: Provide a dispatcher component to act as an intermediate layer between clients and servers. The dispatcher implements a name service that allows client to refer to servers by name instead of physical locations, thus providing transparency. Each server is uniquely identified by its name and is connected to clients by the dispatcher. Client relay on the dispatcher to locate a particular server and to establish a communication like with the server.

5. Proxy: A proxy object can act as the intermediary between the client and target object. The proxy object has the same interface as the target object. The proxy holds a reference to the target object and can forward requests to target as required. In effect the proxy object has the authority the act on behalf of client to interact with the target object.

6. Polymorphism: When related alternatives or behaviors vary by type (class), assign responsibility for the behavior using polymorphic operations to the types for which the behavior varies. This interface will provide the behavior which varies according to the class type. All classes implementing this interface will write the method accordingly.

7. Whole-Part: The idea of the Whole-Part pattern is to introduce a component that encapsulates smaller objects, and prevents clients from accessing these constituent parts directly. Define an interface for the aggregate that is the only means of access to the functionality

of the encapsulated objects, allowing the aggregate to appear as a semantic unit.

8. Controller Design Pattern: Use a controller as the initial point of contact for handling a request. The controller manages the handling of the request, including invoking security services such as authentication and authorization, delegating business processing, managing the choice of an appropriate view, handling errors, and the selection of content creation strategies.

Note 1: In the practical Examination each student has to pick one question from a lot of all the 8 questions.

Note 2: Change of program is not permitted in the Practical Examination.

Note 3: In the Examination partial marks can be given for Class Diagram, Use-Case, Sequence diagrams

WIRELESS COMMUNICATION & MOBILE TECHNOLOGIES

Semester	:	IV	CIE Marks	: 40
Course Code	:	18MCA441	SEE Marks	: 60
Contact Periods (L:T:P)	:	3-0-0	Exam Hours	: 03
Credits : 03				

Course learning objectives:

This course (18MCA441) will enable student to

- Understand the various mobile communication systems.
- Learn various multiplexing systems used in mobile computing.
- Realize the use and importance of data synchronization in mobile computing.

Course outcomes:

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

CO1: Understand the classification of devices, limitations of devices, interfaces, protocol and system Architecture

CO2: Understand the importance of Wireless Medium Access control and CDMA based communication and its Applications.

CO3: Understand the concepts of Network layer, Transport layer.

CO4: Analyze the working of Data Dissemination and Broadcasting Systems.

CO5: Understand and apply the Data Synchronization Server and Management Application languages (XML, Java, J2ME and JavaCard, Mobile Operating Systems).

MODULE -1:

Mobile Devices and Systems, Architectures Mobile phones, Digital Music Players, Handheld Pocket Computers, Handheld Devices, Operating Systems, Smart Systems, Limitations of Mobile Devices, Automotive Systems GSM – Services and System Architectures, Radio Interfaces, Protocols, Localization, Calling, Handover.

MODULE -2:

Wireless Medium Access Control and CDMA – based Communication Medium Access Control, Introduction to CDMA – based Systems. OFDM.

MODULE – 3:

Mobile IP Network Layer Mobile Transport Layer and Databases

Packet Delivery and Handover Management, Registration, Tunneling and Encapsulation, Route Optimization, Dynamic Host Configuration Protocol Indirect TCP, Snooping TCP, Mobile TCP, Other Methods of TCP – layer Transmission for Mobile Networks. Databases Database Hoarding Techniques, Data Caching, Client – Server Computing and Adaptation, Transactional Models, Query Processing, Data Recovery Process

MODULE -4

Data Dissemination and Broadcasting Systems Communication Asymmetry, Classification of Data – Delivery Mechanisms, Data Dissemination Broadcast Models, Selective Tuning and Indexing Techniques, Digital Audio Broadcasting. Digital video Broadcasting.

MODULE -5:

Data Synchronization in Mobile Computing Systems

Synchronization, Synchronization Protocols, SyncML – Synchronization Language for Mobile Computing. Mobile Devices, Server and Management, Wireless LAN, Mobile Internet Connectivity and Personal Area Network Mobile agent, Application Server, Gateways, Portals, Service Discovery, Device Management, Mobile File Systems. Wireless LAN (WiFi) Architecture and Protocol Layers, WAP 1.1 and WAP 2.0 Architectures, Bluetooth – enabled Devices Network, Zigbee. Mobile Application languages – XML, Java, J2ME and JavaCard, Mobile Operating Systems Introduction, XML, JAVA, Java 2 Micro Edition (J2ME), JavaCard Operating System, PalmOS, Windows CE, Symbian OS.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.

- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbook:

1. Raj Kamal: Mobile Computing, Oxford University Press, 2007.

Reference Books:

1. Ashok Talukdar, Roopa R Yavagal: Mobile Computing – Technology, Applications and Service Creation, Tata McGraw Hill, 2005.
2. Reza B'Far: Mobile Computing Principles – Designing and Developing Mobile Applications with UML and XML, 5th Edition, Cambridge University press, 2006.
3. Uwe Hansmann, Lothar Merk, Martin S Nicklous and Thomas Stober: Principles of Mobile Computing, 2nd Edition, Springer International Edition, 2003.
4. Schiller: Mobile Communication, Pearson Publication, 2004.

CYBER SECURITY

Semester	:	IV	CIE Marks : 40
Course Code	:	18MCA442	SEE Marks : 60
Contact Periods (L:T:P)	:	3-0-0	Exam Hours : 03
Credits : 03			

Course learning objectives:

This course (18MCA442) will enable student to

- Learn cyber security concepts, applications and Analyze the working of cyber security principles to system design
- Study appropriate techniques to solve cyber security threats
- Learn how to evaluate cyber security through network security protocols

Course outcomes (CO):

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

CO1: Define and illustrate cyber security concepts and applications

CO2: Analyze the working of cyber security principles to system design

CO3: Illustrate appropriate techniques to solve cyber security threats

CO4: Evaluate and implement cyber security through network security protocols

MODULE -1:

Introduction to Cybercrime and Laws

Introduction, Cybercrime: Definition and Origins of the word, Cybercrime and information Security, Who are Cybercriminals? Classifications of Cybercrimes. How Criminals Plan Them – Introduction, How Criminals Plan the Attacks, Cybercafe and Cybercrimes, Botnets, Attack Vector, The Indian IT ACT 2000.

MODULE -2:

Tools and Methods used in Cybercrime

Introduction, Proxy Server and Anonymizers, Password Cracking, Keyloggers and Spyware, Virus and Worms, Trojan and backdoors, Steganography, DOS and DDOS attack, SQL injection, Buffer Overflow.

MODULE -3:

Phishing and Identity Theft

Introduction, Phishing – Methods of Phishing, Phishing Techniques, Phishing Toolkits and Spy Phishing. Identity Theft – PII,

Types of Identity Theft, Techniques of ID Theft. DigitalForensics Science, Need for Computer Cyberforensics and Digital Evidence, DigitalForensics Life Cycle.

MODULE -4:

Cybercrime: Mobile and Wireless Devices Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing, Attacks on Mobile/Cell Phones.

MODULE -5:

Network Defense tools

Firewalls and Packet Filters: Firewall Basics, Packet Filter Vs Firewall, How a Firewall Protects a Network, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls, NetworkAddress Translation (NAT) and Port Forwarding, the basic of Virtual Private Networks, Linux Firewall, Windows Firewall, Snort: Introduction Detection System.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Anti-Hacker Tool Kit (Indian Edition) by Mike Shema, Publication McGraw Hill. **(Chapters: 2, 7, 8, 11)**
2. Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Nina Godbole and SunitBelpure, Publication Wiley. **(Chapters: 1.1, 1.2, 1.3, 1.4, 1.5, 2.1, 2.2, 2.5, 2.6, 2.7, 6.4, 5.2.1, 5.2.2, 5.2.5, 5.3.1, 5.3.2, 5.3.3, 4.2,4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 4.10, 4.11)**

Reference Books:

1. Marjie T. Britz - Computer Forensics and Cyber Crime: An Introduction - Pearson
2. Chwan-Hwa (John) Wu, J. David Irwin - Introduction to Computer Networks and Cyber security - CRC Press
3. Bill Nelson, Amelia Phillips, Christopher Steuart - Guide to Computer Forensics and Investigations - Cengage Learning

COMPUTER GRAPHICS

Semester	:	IV	CIE Marks	: 40
Course Code	:	18MCA443	SEE Marks	: 60
Contact Periods (L:T:P)	:	3-0-0	Exam Hours	: 03
Credits : 03				

Course Learning Objectives:

This course (18MCA443) will enable students to:

- Familiarize basics of graphics, coordinate systems (in 2D and 3D)
- Understanding various algorithms for drawing shapes, windowing, clipping etc.
- Apply Graphics Library – OpenGL on various shape-drawing algorithms
- Use 2D and 3D transformations in coordinate systems
- Take user input in graphical environment

Course outcomes: The students should be able to

CO1: Design and implement algorithms for 2D graphics primitives and attributes.

CO2: Illustrate Geometric transformations on both 2D and 3D objects.

Co3: Understand the concepts of clipping and visible surface detection in 2D and 3D viewing, and Illumination Models.

CO4: Discuss about suitable hardware and software for developing graphics packages using OpenGL.

MODULE 1:

Overview: Computer Graphics and OpenGL: Computer Graphics: Basics of computer graphics, Application of Computer Graphics, Video Display Devices: Random Scan and Raster Scan displays, color CRT monitors, Flat panel displays. Raster-scan systems: video controller, raster scan Display processor, graphics workstations and viewing systems, Input devices, graphics networks, graphics on the internet, graphics software. OpenGL: Introduction to OpenGL, coordinate reference frames, specifying two-dimensional world coordinate reference frames in OpenGL, OpenGL point functions, OpenGL line functions, point attributes, line attributes, curve attributes, OpenGL point attribute functions, OpenGL line

attribute functions, Line drawing algorithms(DDA, Bresenham's), circle generation algorithms(Bresenham's).

MODULE 2:

Fill area Primitives, 2D Geometric Transformations and 2D viewing: Fill area Primitives: Polygon fill-areas, OpenGL polygon fill area functions, fill area attributes, general scan line polygon fill algorithm, OpenGL fill-area attribute functions. 2D Geometric Transformations: Basic 2D Geometric Transformations, matrix representations and homogeneous coordinates. Inverse transformations, 2D Composite transformations, other 2D transformations, raster methods for geometric transformations, OpenGL raster transformations, OpenGL geometric transformations function, 2D viewing: 2D viewing pipeline, OpenGL 2D viewing functions.

MODULE 3:

Clipping 3D Geometric Transformations, Color and Illumination Models: Clipping: clipping window, normalization and viewport transformations, clipping algorithms, 2D point clipping, 2D line clipping algorithms: Cohen-Sutherland line clipping polygon fill area clipping: Sutherland-Hodgeman polygon clipping algorithm only. 3D Geometric Transformations: 3D translation, rotation, scaling, composite 3D transformations, other 3D transformations, affine transformations, OpenGL geometric transformations functions. Color Models: Properties of light, color models, RGB and CMY color models. Illumination Models: Light sources, basic illumination models-Ambient light, diffuse reflection, specular and Phong model, Corresponding OpenGL functions.

MODULE 4:

3D Viewing and Visible Surface Detection: 3D Viewing: 3D viewing concepts, 3D viewing pipeline, 3D viewing coordinate parameters, Transformation from world to viewing coordinates, Projection transformation, orthogonal projections, perspective projections, The viewport transformation and 3D screen coordinates. OpenGL 3D viewing functions. Visible Surface Detection Methods:

Classification of visible surface Detection algorithms, back face detection, depth buffer method and OpenGL visibility detection functions.

MODULE 5:

Input& interaction, Curves and Computer Animation: Input and Interaction: Input devices, clients and servers, Display Lists, Display Lists and Modelling, Programming Event Driven Input, Menus Picking, Building Interactive Models, Animating Interactive programs, Design of Interactive programs, Logic operations .Curved surfaces, quadric surfaces, OpenGL Quadric-Surface and Cubic-Surface Functions, Bezier Spline Curves, Bezier surfaces, OpenGL curve functions. Corresponding openGL functions.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Donald Hearn & Pauline Baker: Computer Graphics with OpenGL Version,3rd/ 4 thEdition, Pearson Education,2011
Chapter -1: 1-1 to 1-9,2-1 to 2-9 (Excluding 2-5),3-1 to 3-5,3-9,3-20.
Chapter 3-14 to 3-16,4-9,4-10,4-14,5-1 to 5-7,5-17,6-1,6-4.
Chapter: 7-1 to 7-10(Excluding 7-7), 9-1 to 9-3, 9-14.Chapter :8-3 to 8-6 (Excluding 8-5),8-9,8-10,8-11,3-8,8-18,13-11,3-2,13-3,13-4,13-10.Chapter :6-2 to 6-08 (Excluding 6-4),5-9 to 5-17(Excluding 5-15),12- 1,12-2,12-4,12-6,10-1,10-3
2. Edward Angel: Interactive Computer Graphics- A Top Down approach with OpenGL, 5 th edition. Pearson Education, 2008
Chapter 3: 3-1 to 3.11: Input& interaction

Reference Books:

1. James D Foley, Andries Van Dam, Steven K Feiner, John F

- Huges Computer graphics with OpenGL: pearson education
2. Xiang, Plastock : Computer Graphics , sham's outline series, 2nd edition, TMG.
3. Kelvin Sung, Peter Shirley, steven Baer : Interactive Computer Graphics, concepts and applications, Cengage Learning
4. M MRaiker, Computer Graphics using OpenGL, Filip learning/Elsevier

CLOUD COMPUTING

Semester	:	IV	CIE Marks	: 40
Course Code	:	18MCA444	SEE Marks	: 60
Contact Periods (L:T:P)	:	3-0-0	Exam Hours	: 03
Credits : 03				

Course learning objectives:

This course (18MCA444) will enable student to

- Understand the technology and principles involved in building a cloud environment.
- Contrast various programming models used in cloud computing
- Choose appropriate cloud model for a given application.

Course outcomes:

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

CO1: Describe fundamental and core concepts of cloud computing.

CO2: Understand the fundamentals concepts of computing paradigms like parallel and distributed computing.

CO3: Investigate the system virtualization and outline its role in enabling the cloud computing system model.

CO4: Configure and Demonstrate different deployment and service models of cloud to develop different variety of applications.

MODULE -1:

Introduction to Cloud Computing

Eras of Computing, The Vision of Cloud Computing, Defining a cloud, A Closer Look, Cloud computing Reference Model, Historical developments: Distributed Systems; Virtualization; Web 2.0; Service Oriented Computing; Utility Oriented Computing.

MODULE -2:

Architectures for parallel and distributed computing

Parallel Vs Distributed computing, Elements of distributed computing, Technologies for distributed computing.

MODULE -3:

Virtualization

Introduction, Characteristics of virtualized Environments, Taxonomy of Virtualization techniques, Virtualization and Cloud computing, Pros and Cons of virtualization, Technology Examples: Xen: Para virtualization, VmWare : Full Virtualization, Microsoft Hyper-V.

MODULE -4:

Cloud Computing Architecture

Introduction, Cloud reference Model: Architecture, IaaS, PaaS, SaaS.

Types of Clouds: Public, Private, Hybrid and Community Clouds, Economics of the Cloud, Open Challenges.

MODULE -5:

Cloud Tools and Applications

Aneka PaaS; Open Stack: Introduction to open stack; Components of open stack; Amazon Web Services; Google AppEngine; Microsoft Azure; Scientific applications: Health Care; Biology; Geo- Science, Business And Consumer Applications: CRM & ERP; Productivity; Social Networking;

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Textbook:

1.Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, Mastering Cloud Computing, Tata McGraw Hill, New Delhi, India, 2013.

Reference Books:

1.Cloud Computing for Dummies by Judith Hurwitz, R.Bloor, M.Kanfman, F.Halper (Wiley India Edition)

2.Cloud Computing: A Practical Approach by J.Vette, Toby J. Vette, Robert Elsenpeter (Tata McGraw Hill)

3.Cloud Computing: Principles and Paradigms, Rajkumar Buyya, Andrzej M. Goscinski, James Broberg John Wiley & Sons, 17-Dec-2010

3.“Cloud Computing for Dummies”, J. Hurwitz, ISBN 978-0-470-484-8

4.“Web-Based Applications that Change the Way You Work and Collaborate Online”, Michael Miller, Pearson Publication, 2012.

ENTERPRISE RESOURCE PLANNING (ERP)

Semester	:	IV	CIE Marks	: 40
Course Code	:	18MCA451	SEE Marks	: 60
Contact Periods (L:T:P)	:	3-0-0	Exam Hours	: 03
Credits : 03				

Course learning objectives:

This course (18MCA451) will enable student to

- Understand and build the fundamental concepts of ERP systems, their architecture, and working of different modules in ERP.
- Develop and design the modules used in ERP systems, and can customize the existing modules of ERP systems.

Course Outcomes (CO):

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

CO1: Acquire knowledge of

- Benefits of ERP, Process Re-engineering
- Project management and Monitoring

CO2: Analyze the performance of

- Project implementations
- Quality management

CO3: Know how ERP evolves in market place

CO4: Develop the ERP system, ERP with E-Commerce & Internet

MODULE -1:

Introduction To ERP

Overview, Benefits of ERP, ERP and Related Technologies, Business Process Reengineering, Data Warehousing, Data Mining, On-line Analytical Processing, Supply Chain Management

MODULE -2:

ERP Implementation

Implementation of Life Cycle, Implementation Methodology, Hidden Costs, Organizing Implementation, Vendors, Consultants and Users, Contracts, Project Management and Monitoring

MODULE – 3:

Business Modules

Business Modules in an ERP Package, Finance, Manufacturing,

Human Resource, Plant Maintenance, Materials Management, Quality Management, Sales and Distribution

MODULE -4:

ERP Market

ERP Market Place, SAP AG, PeopleSoft ,Baan Company , JD Edwards World Solutions Company, Oracle Corporation, QAD , System Software Associates.

MODULE -5:

ERP–Present And Future

Turbo Charge the ERP System, EIA, ERP and E–Commerce, ERP and Internet, Future Directions in ERP.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Alexis Leon, “ERPDemystified”, Tata McGraw Hill, 1999.
2. Joseph A. Brady, Ellen F. Monk, Bret J. Wangner, “Concepts in Enterprise Resource Planning”, Thomson Learning, 2001.

Reference Books:

1. Vinod Kumar Garg and N.K .Venkata Krishnan, “Enterprise Resource Planning concepts and Planning”, Prentice Hall, 1998.
2. Jose Antonio Fernandz, “ The SAP R /3 Hand book”, Tata McGraw Hill

DATA WAREHOUSING AND DATA MINING

Semester	:	IV	CIE Marks : 40
Course Code	:	18MCA452	SEE Marks : 60
Contact Periods (L:T:P)	:	3-0-0	Exam Hours : 03
Credits : 03			

Course learning objectives:

This course (18MCA452) will enable student to

- Learn the concept of Data warehousing and OLAP and Understand storage and retrieval technique of data from DATA CUBE.
- Study different types of data and different preprocessing techniques.
- Understand various Association algorithms and its applications.
- Learn how to apply different Classification technique and to evaluate different types of classifiers.
- Distinguish different clustering techniques and their applications.

Course Outcome (CO):

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

CO1: Learn the concept of Data warehousing and OLAP.

CO2: Understand storage and retrieval technique of data from DATA CUBE.

CO3: Analyze different types of data and different preprocessing techniques.

CO4: Evaluate various Association algorithms and its applications.

CO5: Apply different Classification technique.

CO6: Evaluate different types of classifiers.

CO7: Analyze different clustering techniques and their applications

MODULE -1:

Data warehousing and OLAP

Data Warehouse basic concepts, Data Warehouse Modeling, Data Cube and OLAP : Characteristics of OLAP systems, Data Cube Implementations, Data Cube operations, Implementation of OLAP and overview on OLAP Software.

MODULE -2:

Data Mining and its Applications

Introduction, What is Data Mining, Motivating Challenges, Data Mining Tasks, Which technologies are used for data mining, Kinds of pattern that can be mined, Data Mining Applications, Data Preprocessing, Data cleaning, data integration, data reduction and data transformation.

MODULE-3:

Association Analysis: Basic Concepts and Algorithms

Frequent Item set Generation, Rule Generation, Compact Representation of Frequent Item sets, Alternative methods for generating Frequent Item sets, FP Growth Algorithm, Evaluation of Association Patterns

MODULE-4:

Classification : Methods, Improving accuracy of classification

Basics, General approach to solve classification problem, Decision Trees, Rule Based Classifiers, Nearest Neighbor Classifiers. Bayesian Classifiers, Estimating Predictive accuracy of classification methods, Improving accuracy of classification methods, Evaluation criteria for classification methods, Multiclass Problem.

MODULE-5:

Clustering Techniques

Overview, Features of cluster analysis, Types of Data and Computing Distance, Types of Cluster Analysis Methods, Partitional Methods, Hierarchical Methods, Density Based Methods, Quality and Validity of Cluster Analysis

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one

full question from each Module.

Textbooks:

1. Jiawei Han and Micheline Kamber: Data Mining - Concepts and Techniques, 2nd Edition, Morgan Kaufmann Publisher, 2006.
2. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Addison- Wesley, 2005.

Reference Books:

1. Arun K Pujari: Data Mining Techniques University Press, 2nd Edition, 2009.
2. G. K. Gupta: Introduction to Data Mining with Case Studies, 3rd Edition, PHI, New Delhi, 2009.
3. Alex Berson and Stephen J.Smith: Data Warehousing, Data Mining, and OLAP Computing McGrawHill Publisher, 1997.

ADVANCED DATABASE MANAGEMENT SYSTEMS

Semester	:	IV	CIE Marks : 40
Course Code	:	18MCA453	SEE Marks : 60
Contact Periods (L:T:P)	:	3-0-0	Exam Hours : 03
Credits : 03			

Course learning objectives:

This course (18MCA453) will enable student to

- Learn the knowledge on Storage and Indexing
- Realize the concept of Transaction Management
- Understand the concept of Query Evaluation, External Sorting, Evaluating Relational Operators and Relational Query Optimizer
- Understand the concept of query optimization for solving real world problem

Course Outcome (CO):

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

CO1: Acquire knowledge on Storage and Indexing

CO2: Learn the concept of Transaction Management

CO3: Understand the concept of Query Evaluation, External Sorting, Evaluating Relational Operators and Relational Query Optimizer

CO4: Apply the concept of query optimization into real world problem

MODULE-1:

Over view of Storage and Indexing, Disks and Files

Data on external storage; File organizations and indexing; Index data structures; Comparison of file organizations; Indexes and performance tuning Memory hierarchy; RAID; Disk space management; Buffer manager; Files of records; Page formats and record formats.

Transaction Management

Introduction to Transaction Processing; Transaction and System Concepts; Desirable Properties of Transactions; Characterizing Schedules based on Recoverability; Characterizing Schedules based on Serializability; Two-Phase Locking Techniques for Concurrency Control; Concurrency Control based on Timestamp Ordering;

Recovery Concepts, Recovery Techniques based on Deferred Update; Recovery Techniques based on Immediate Update; Shadow Paging; The ARIES Recovery Algorithms; Database Backup and Recovery from Catastrophic Failures.

MODULE-2:

Tree Structured Indexing

Intuition for tree indexes; Indexed sequential access method; B+trees, Search, Insert, Delete, Duplicates, B+trees in practice

Hash-Based Indexing

Static hashing, Extendible hashing, Linear hashing, comparisons

MODULE-3:

Overview of Query Evaluation, External Sorting

The system catalog, Introduction to operator evaluation; Algorithm for relational operations; Introduction to query optimization; Alternative plans; A motivating example; what a typical optimizer does. When does a DBMS sort data? A simple two-way merge sort; External merge sort

Evaluating Relational Operators

The Selection operation; General selection conditions; The Projection operation; The Join operation; The Set operations; Aggregate operations; The impact of buffering.

MODULE-4:

A Typical Relational Query Optimizer

Translating SQL queries in to Relational Algebra; Estimating the cost of a plan; Relational algebra equivalences; Enumeration of alternative plans; Nested sub-queries; other approaches to query optimization.

MODULE-5:

Physical Database Design and Tuning

Introduction; Guidelines for index selection ,examples; Clustering and indexing; Indexes that enable index-only plans, Tools to assist in index selection; Overview of database tuning; Choices in tuning the conceptual schema; Choices in tuning queries and views; Impact of concurrency; DBMS benchmarking.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions from each Module.
- Each question will have questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each module

Textbooks:

1. Raghu Ramakrishnan and Johannes Gehrke: Database Management Systems,3rd Edition, McGraw-Hill,2003, (Chapters 8,9,10,11,12,13.1 to 13.3, 14,15,20)
2. Elmasri and Navathe:Fundamentals of Database Systems,5th Edition, Pearson Education, 2007. (Chapter 30)

Reference Book:

1. Conolly and Begg: Database Systems,4th Edition, Pearson Education,2002.

BIG DATA ANALYTICS

Semester	:	IV	CIE Marks : 40
Course Code	:	18MCA454	SEE Marks : 60
Contact Periods (L:T:P)	:	3-0-0	Exam Hours : 03
Credits : 03			

Course learning objectives:

This course (18MCA454) will enable student to

- Learn the Map Reduce technique for solving Big Data problems
- Realize algorithms for Big Data by deciding on the apt Features set
- Learn how to apply algorithms for handling peta bytes of datasets
- Realize the main memory consumption for Big Data analytics and analyze the usage of map reduce techniques for solving big data problems

Course outcomes (CO):

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

CO1: Understand the Map Reduce technique for solving Big Data problems

CO2: Understand algorithms for Big Data by deciding on the apt Features set

CO3: Apply algorithms for handling peta bytes of datasets

CO4: Analyze main memory consumption for Big Data analytics

CO5: Understand and analyze the usage of map reduce techniques for solving big data problems

MODULE -1:

Big Data and Analytics

Example Applications, Basic Nomenclature, Analysis Process Model, Analytical Model Requirements , types of Data Sources, Sampling, Types of data elements, data explorations, exploratory statistical analysis, missing values, outlier detection and Treatment, standardizing data labels, categorization

MODULE -2:

Big Data Technology

Hadoop's Parallel World – Data discovery – Open source technology

for Big Data Analytics – cloud and Big Data –Predictive Analytics – Mobile Business Intelligence and Big Data – Crowd Sourcing Analytics – Inter- and Trans-Firewall Analytics

MODULE -3:

Meet Hadoop

Data, Data Storage and Analysis ,Comparison with Other Systems,RDBMS,Grid Computing Volunteer Computing, A Brief History of Hadoop,Apache Hadoop and the Hadoop Ecosystem Hadoop Releases Response

MODULE -4:

The Hadoop Distributed File system

The Design of HDFS, HDFS Concepts, Blocks, Namenodes and Datanodes, HDFS Federation, HDFS High-Availability, The Command-Line Interface, Basic Filesystem Operations, Hadoop Filesystems Interfaces ,The Java Interface, Reading Data from a Hadoop URL, Reading Data Using the FileSystem API, Writing Data, Directories, Querying the Filesystem, Deleting Data, Data Flow Anatomy of a File Read ,Anatomy of a File Write, Coherency Model, Parallel Copying with distcp Keeping an HDFS Cluster Balanced, Hadoop Archives

MODULE -5:

Map Reduce

A Weather Dataset ,Data Format, Analyzing the Data with Unix Tools, Analyzing the Data with Hadoop, Map and Reduce, Java MapReduce, Scaling Out, Data Flow, Combiner functions, Running a Distributed MapReduce Job, Hadoop Streaming, Hadoop Pipes, Compiling and Running, Developing a MapReduce Application, The Configuration API, Combining Resources, Variable Expansion, Configuring the Development Environment, Managing Configuration, GenericOptionsParser, Tool and ToolRunner, Writing a Unit Test, Mapper, Reducer, Running Locally on Test Data, Running a Job in a Local Job Runner, Testing the Driver, Running on a Cluster, Packaging, Launching a Job, The MapReduce Web UI, Retrieving the Results, Debugging a Job, Hadoop Logs, Remote Debugging.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Bart Baesens, “ Analytics in a Big Data World : The Essential Guide to Data Science and its Applications” Wiley
2. Michael Minelli, Michele Chambers, “Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today’s Businesses”, 1st Edition, Michael Minelli, Michele Chambers, Ambiga Dhiraj, Wiley CIO Series, 2013.
3. Tom White, “Hadoop: The Definitive Guide”, 3rd Edition, O’reilly, 2012

Reference Books:

1. Boris Iubinskyy, Kevin T. Smith, Alexey Yakubovich, “Professional Hadoop Solutions”, Wiley, ISBN: 9788126551071, 2015.
2. Chris Eaton, Dirk Derouos et al. , “Understanding Big data ”, McGraw Hill, 2012.
3. Vignesh Prajapati, “Big Data Analytics with R and Hadoop”, Packet Publishing 2013.
4. Tom Plunkett, Brian Macdonald et al, “Oracle Big Data Handbook”, Oracle Press, 2014.

SEMESTER – V PROGRAMMING USING C# .NET

Semester	V	CIE Marks : 40
Course Code	18MCA51	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course(18MCA51) will enable the students to:

- Familiarize .NET Framework and OOPs concepts in C#
- Analyze the concepts of Delegates, Events and ADO.NET
- Illustrate Windows applications using C# .NET; web applications using ASP.NET and AJAX

Course Outcomes (CO):

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

CO1: Understand C# and client-server concepts using .Net Framework Components.

CO2: Apply delegates, event and exception handling to incorporate with ASP, Win Form, ADO.NET.

CO3: Analyze the use of .Net Components depending on the problem statement.

CO4: Implement & develop a web based and Console based application with Database connectivity

MODULE -1:

Getting started with .NET Framework 4.0 and C#

Understanding Previous Technologies, Benefits of .NET Framework, Architecture of .NET Framework 4.0, .NET Execution Engine, Components of .NET Framework 4.0: CLR, CTS, Metadata and Assemblies, .NET Framework Class Library, Windows Forms, ASP .NET and ASP .NET AJAX, ADO .NET, Windows Workflow Foundation, Windows Presentation Foundation, Windows Communication Foundation, Windows Card Space and LINQ.

Introducing C#

Creating a Simple C# Console Application, Identifiers and Keywords. System Data Types, Variables and Constants: Value Types, Reference Types, Understanding Type Conversions, Boxing and UnBoxing. Namespaces, The System namespace, .NET Array Types

MODULE -2:

Classes, Objects and Object Oriented Programming

Classes and Objects: Creating a Class, Creating an Object, Using this Keyword, Creating an Array of Objects, Using the Nested Classes, Defining Partial Classes and Method, Returning a Value from a Method and Describing Access Modifiers. Static Classes and StaticMembers, Properties: Read-only Property, Static Property, Indexers, Structs: Syntax of a struct and Access Modifiers for structs, System.Object Class

Encapsulation: Encapsulation using accessors and mutators, Encapsulation using Properties. Inheritance: Inheritance and Constructors, Sealed Classes and Sealed Methods, Extension methods.

Polymorphism: Compile time Polymorphism/ Overloading, Runtime Polymorphism/ Overriding. Abstraction: Abstract classes, Abstract methods. Interfaces: Syntax of Interfaces, Implementation of Interfaces and Inheritance.

MODULE -3:

Delegates, Events, Exception Handling and ADO.NET

Delegates: Creating and using Delegates, Multicasting with Delegates. Events: Event Sources, Event Handlers, Events and Delegates, Multiple Event Handlers.

Exception Handling: The try/catch/throw/finally statement, Custom Exception.System.Exception, Handling Multiple Exception

Data Access with ADO.NET : Understanding ADO.NET: Describing the Architecture of ADO.NET, ADO.NET,ADO.NET Entity Framework. Creating Connection Strings: Syntax for Connection Strings.Creating a Connection to a Database: SQL Server Database, OLEDB Database, ODBC Data Source. Creating a Command Object. Working with DataAdapters: Creating DataSet from DataAdapter.

MODULE -4:

Graphical User Interface with Windows Forms and WPF

Windows Forms : Introduction, Windows Forms, Event Handling: A Simple Event- Driven GUI, Control Properties and Layout, Labels,

TextBoxes and Buttons, GroupBoxes and Panels, CheckBoxes and RadioButtons, ToolTips, Mouse-Event Handling, Keyboard-Event Handling. Menus, Month Calendar Control, LinkLabel Control, ListBox Control, ComboBox Control, TreeView Control, ListView Control, TabControl and Multiple Document Interface (MDI) Windows.

WPF: New WPF Controls, WPF Architecture: Presentation Framework, Presentation Core, WindowsBase, MIL or Milcore, Working with WPF Windows: Using XAML in WPF 4.0 Applications : Contents of XAML and WPF Applications: XAML Elements Namespace and XAML, XAML Property Syntax, Markup Extensions.

MODULE -5:

Web App Development and Data Access using ADO.NET

Introduction, Web Basics, Multitier Application Architecture,

Your First Web Application: Building Web-Time Application, Examining Web-Time.aspx's Code-Behind File, Understanding Master pages, Standard Web Controls: Designing a Form, Validation Controls, GridView Control, DropDownList, Session Tracking, ASP.NET AJAX : Exploring AJAX,Need for AJAX, AJAX and other Technologies, AJAX Server Controls, ScriptManager control, Update Panel, UpdateProgress Control, Creating Simple Application using AJAX Server Controls

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. .NET 4.0 Programming (6-in-1), Black Book, Kogent Learning Solutions Inc., Wiley- Dream Tech Press.

(Chapters: 1,10,11,12,13,14 and 19).

2. Paul Deitel and Harvey Deitel: C# 2010 for Programmers, 4th Edition, Pearson Education.

(Chapters: 14,15,19 and 27.3)

Reference Books:

1. **Andrew Trolsen:** Pro C# 5.0 and the .NET 4.5 Framework, 6th Edition, Wiley-Appress.
2. **Bart De Smet:** C# 4.0 Unleashed, Pearson Education- SAMS Series.
3. **Herbert Schildt:** Complete Reference C# 4.0, Tata McGraw Hill, 2010.

MOBILE APPLICATIONS

Semester	V	CIE Marks : 40
Course Code	18MCA52	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course(18MCA52) will enable students to:

- Familiarize with the advantages and limitations of mobile application development
- Learn business trends impacting mobile application development
- Characterize the architecture of mobile applications
- Develop, deploy and test mobile applications

Course Outcomes (CO):

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

CO1: Illustrate effective user interfaces that leverage evolving mobile device capabilities

CO2: Develop applications using software development kits (SDKs), frameworks and toolkits

CO3: Establish various methods to integrate database and server-side technologies

CO4: Design and develop open source software based mobile applications

CO5: Build and deploy competent mobile development solutions

MODULE -1:

Introduction

Preliminary Considerations – Cost of Development – Importance of Mobile Strategies in the Business World – Effective use of Screen Real Estate –

Understanding Mobile Applications

Understanding Mobile Applications Users – Understanding Mobile Information Design – Understanding Mobile Platforms – Using the Tools of Mobile Interface Design.

MODULE -2:

Getting Started with Android Programming

What is Android – Obtaining the required tools– Anatomy of an

Android Application – Components of Android Applications – Activities – Fragments – Utilizing the Action Bar

MODULE -3:

Android UI Design and Location Based Services

Views and View Groups – Basic Views – Fragments – Displaying Maps – Getting Location Data – Publishing for Publishing – Deploying APK Files

MODULE -4

Android Messaging and Networking

SMS Messaging – Sending Email – Networking – Downloading Binary Data, Text files – Accessing Web Services – Performing Asynchronous Call – Creating your own services – Communicating between a service and an activity – Binding activities to services

MODULE -5:

Feedback and Oscillator Circuits

iOS – Obtaining the tools and SDK – Components of XCODE – Architecture of iOS – Building Derby App in iOS – Other useful iOS things – Windows Phone: Getting the tools you need – Windows Phone 7 Project – Building Derby App in Windows Phone 7 – Distribution – Other useful Windows Phone Thing

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Jeff McWherter and Scott Gowell, “Professional Mobile Application Development”, 1st Edition, 2012, ISBN: 978-1-118-20390-3

2. Wei-Meng Lee, “Beginning Android Application Development”, Wiley 2011.

Reference Book:

1. Reto Meier, “Professional Android 4 Application Development”, Wrox Publications 2012

MACHINE LEARNING

Semester	V	CIE Marks : 40
Course Code	18MCA53	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course(18MCA53) will enable the students to:

- Explain the basic concepts of instance based learning and decision trees.
- Understand the ANN, genetic algorithms, and Bayesian methods.
- Familiarize the analytical and reinforcement learning

Course Outcomes:

After studying this course, students will be able to

CO1: Develop an appreciation for what is involved in learning models from data.

CO2: Differentiate supervised, unsupervised and reinforcement learning.

CO3: Apply neural networks, Bayes classifier and k nearest neighbor, for real world problems.

CO4: Perform statistical analysis of machine learning techniques.

CO5: Understand theory of probability and statistics related to machine learning.

CO6: Understand a wide variety of learning algorithms.

MODULE – 1

Introduction: Well posed learning problems, Designing a Learning system, Perspective and Issues in Machine Learning. Concept Learning: Concept learning task, Concept learning as search, Find-S algorithm, Version space, Candidate Elimination algorithm, Inductive Bias.

MODULE – 2

Decision Tree Learning: Decision tree representation, Appropriate problems for decision tree learning, Basic decision tree learning algorithm, hypothesis space search in decision tree learning, Inductive bias in decision tree learning, Issues in decision tree learning.

MODULE – 3

Artificial Neural Networks: Introduction, Neural Network representation, Appropriate problems, Perceptrons, Backpropagation algorithm.

MODULE – 4

Bayesian Learning: Introduction, Bayes theorem, Bayes theorem and concept learning, ML and LS error hypothesis, ML for predicting probabilities, MDL principle, Naive Bayes classifier, Bayesian belief networks, EM algorithm.

MODULE – 5

Evaluating Hypothesis: Motivation, Estimating hypothesis accuracy, Basics of sampling theorem, General approach for deriving confidence intervals, Difference in error of two hypothesis, Comparing learning algorithms. Instance Based Learning: Introduction, k-nearest neighbor learning, locally weighted regression, radial basis function, cased-based reasoning, Reinforcement Learning: Introduction, Learning Task, Q Learning.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbook:

1. Tom M. Mitchell, Machine Learning, India Edition 2013, McGraw Hill Education.
(Sections: 1.1 – 1.3, 2.1-2.5, 2.7 3.1-3.7, 4.1 – 4.6, 6.1 – 6.6, 6.9, 6.11, 6.12, 5.1-5.6, 8.1-8.5, 13.1-13.3)

Reference Books:

1. Trevor Hastie, Robert Tibshirani, Jerome Friedman, h The Elements of Statistical Learning, 2nd edition, springer series in statistics.
2. Ethem Alpaydın, Introduction to machine learning, second edition, MIT press.

C# .NET LABORATORY

Semester	V	CIE Marks : 40
Course Code	18MCA56	SEE Marks : 60
Contact Periods (L:T:P)	0-1-3	Exam Hours : 03
Credits : 02		

Course Learning Objectives:

This course(18MCA56) will enable the students to:

- Implement basic concepts of C#.
- Debug and execute windows and web applications.

Course Outcomes :

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

CO1: Understand C# and client-server concepts using .Net Framework Components

CO2: Apply delegates, event and exception handling to incorporate with ASP, Win Form, ADO.NET

CO3: Analyze the use of .Net Components depending on the problem statement

CO4: Implement & develop a web based and Console based application with Database connectivity

PART A

1. Write a Program in C# to demonstrate Command line arguments processing for the following.
 - a) To find the square root of a given number.
 - b) To find the sum & average of three numbers.
2. Write a Program in C# to demonstrate the following
 - a) Boxing and Unboxing
 - b) Invalid Unboxing.
3. Write a program in C# to add Two complex numbers using Operator overloading .
4. Write a Program in C# to find the sum of each row of given jagged array of 3 inner arrays.
5. Write a Program in C# to demonstrate Array Out of Bound Exception using Try, Catch and Finally blocks.
6. Write a Program to Demonstrate Use of Virtual and override key words in C# with a simple program.
7. Write a Program in C# to create and implement a Delegate for any two arithmetic operations
8. Write a Program in C# to demonstrate abstract class and

- abstract methods in C#.
- Write a program to Set & Get the Name & Age of a person using Properties of C# to illustrate the use of different properties in C#.
 - Write a Program in C# Demonstrate arrays of interface types (for runtime polymorphism).

PART-B

I. Consider the Database db_EMS (Employee Management System)

consisting of the following tables :

tbl_Designations (IdDesignation: int, Designation: string)

tbl_EmployeeDetails (IdEmployee: int, EmployeeName: string, ContactNumber: string, IdDesignation: int, IdReportingTo: int)

Develop a suitable window application using C#.NET having following options.

- Enter new Employee details with designation & Reporting Manager.
- Display all the Project Leaders (In a Grid) reporting to selected Project Managers (In a Combo box).
- Display all the Engineers (In a Grid) reporting to selected Project Leader (In a Combo box).
- Display all the Employees (In a Grid) with their reporting Manager (No Value for PM).

NOTE: tbl_Designation is a static table containing the following Rows in it.

- | | |
|---|-----------------|
| 1 | Project Manager |
| 2 | Project Leader |
| 3 | Engineer |

II. Consider the Database db_LSA (Lecturer Subject Allocation) consisting of the following tables:

tbl_Subjects (IdSubject: int, SubjectCode: string, SubjectName: string)

tbl_Lecturers (IdLecturer: int, LecturerName: string, ContactNumber: string)

tbl_LecturerSubjects (IdSubject: int, SubjectCode: string,

IdLecturer: int)

Develop a suitable window application using C#.NET having following options.

- Enter new Subject Details.
- Enter New Lecturer Details.
- Subject Allocation with Lecturer Name in a Combo box and subjects to be allocated in Grid with checkbox Column.
- Display all the subjects allocated (In a Grid) to the selected Lecturer (In a Combo Box).

III. Consider the database db_VSS (Vehicle Service Station) consisting of the following tables:

tbl_VehicleTypes (IdVehicleType: int, VehicleType: string, ServiceCharge: int)

tbl_ServiceDetails (IdService: int, VehicleNumber: string, ServiceDetails: string, IdVehicleType: int)

Develop a suitable window application using C#.NET having following options.

- Enter new Service Details for the Selected Vehicle Type (In a Combo Box).
- Update the Existing Service Charges to Database.
- Total Service Charges Collected for the Selected Vehicle (In a Combo box) with total amount displayed in a text box.

NOTE: tbl_VehicleType is a static table containing the following Rows in it.

- | | | |
|---|---------------|------|
| 1 | Two Wheeler | 500 |
| 2 | Four Wheeler | 1000 |
| 3 | Three Wheeler | 700 |

IV. Develop a web application using C#.NET and ASP.NET for the Postal System Management. The master page should contain the hyper links for adding Area Details, Postman details, Letter distributions and View Letters.

Consider the database db_PSM (Postal System Management) consisting of the following tables:

tbl_AreaDetails (IdArea: int, AreaName: string)

tbl_PostmanDetails (IdPostman: int, PostmanName: string,

ContactNumber: string, IdArea: int)
tbl_AreaLetters(IdLetter: int, LetterAddress: string, IdArea:
int)

Develop the suitable content pages for the above created 4 hyper links with the following details:

1. Enter New Area Details
2. Enter New Postman Details with the Area he/she is in-charge of (display Area in a Combo box)
3. Enter all the Letters distributed to the selected Area (display Area in a Combo box)
4. Display all the Letter addresses (In a Grid) to be distributed by the selected Postman (In a Combo box)

Note 1: In the practical Examination each student has to pick one question from Part A and Part B each.

Note 2: Change of program is not permitted in the Practical Examination.

Note 3: Part A has to be evaluated for 30 marks and Part B has to be evaluated for 30 marks.

MOBILE APPLICATIONS LABORATORY

Semester	V	CIE Marks : 40
Course Code	18MCA57	SEE Marks : 60
Contact Periods (L:T:P)	0-1-3	Exam Hours : 03
Credits : 02		

Course Learning objective: This Course(18MCA57) enable the students to:

- Create a project and develop mobile application in Android studio using XML and Java programming language.
- Test and Debug a mobile application for a reliable output
- Develop a project and emphasize its applications and uses to the real world.

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

CO1: Illustrate effective user interfaces that leverage evolving mobile device capabilities

CO2: Develop applications using software development kits (SDKs), frameworks and toolkits

CO3: Establish various methods to integrate database and server-side technologies

CO4: Design and develop open source software based mobile applications

CO5: Build and deploy competent mobile development solutions

Laboratory Experiments:

The laboratory can be carried out only using any mobile application software.

1. Develop a mobile application to display a background image of your college and print “Hello MCA” in green color.
2. Develop a standard calculator application to perform basic calculations like addition, subtraction, multiplication and division using 2 EditText and 1 TextView for displaying result.
3. Develop a mobile application to display user profile with 3 UI activities using intents.
4. Develop a mobile application to register a form in first activity and display the registered information in second activity using intents.

5. Develop a mobile application to list the tourist places of Karnataka using ListView.
6. Design an application that contains Phone Contacts in vertical linear manner. Selected contact appears at the top of the list with a large italicized font and a blue background.
7. Create an application that uses Layout Managers and Event Listeners.
8. Devise an application that draws basic graphical primitives (rectangle, circle) on the screen.
9. Build a mobile application that create, save, update and delete data in a database.
10. Devise an application that implements Multi threading.
11. Develop a mobile application that uses GPS location information.
12. Create an application that writes data to the SD card.
13. Implement an application that creates an alert upon receiving a message.
14. Devise a mobile application that creates alarm clock.

Note:

1. Students are required to execute one question from lot of all 14 Questions.
2. Change of program is not permitted in the examination.

MINI PROJECT

Semester	V	CIE Marks : 40
Course Code	18MCA58	SEE Marks : 60
Contact Periods (L:T:P)	0-2-4	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course (18MCA58) will enable students to:

- Acquire practical knowledge within the chosen area of technology for project development
- Identify, analyze, formulate and handle programming projects with a comprehensive and systematic approach
- Contribute as an individual or in a team in development of technical projects
- Develop effective communication skills for presentation of project related activities

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

CO1: Identify a suitable problem making use of the technical and engineering knowledge gained from previous courses with the awareness of impact of technology on the society and their ethical responsibilities.

CO2: Ability to segregate work and execute/implement projects using appropriate tools.

CO3: Develop skills to disseminate technical and general information by means of oral as well as written presentation skills.

Develop an application using the languages and concepts learnt in the theory and exercises listed upto V semester with a good look and feel effects.

Note :

1. In the examination student has to demonstrate the project.
2. A team of maximum two students must develop the project. However during the examination, each student must demonstrate the project individually.
3. The team must submit a brief project report (20-30 pages) that must include the following

- a. Introduction
- b. Requirement Analysis
- c. Software Requirement Specification
- d. Analysis and Design
- e. Implementation
- f. Testing
- g. Conclusion

MULTIMEDIA SYSTEMS

Semester	V	CIE Marks : 40
Course Code	18MCA541	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course (18MCA541) will enable students to:

- Synthesize the key components of multimedia technologies including text, graphics, voice, video and animation
- Appraise on online and web development environment
- Familiarize each media type and their significance with its applications and formats
- Edit interactive web pages that incorporate a variety of digital media

Course Outcomes:

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

CO1: Understand the concepts of Multimedia technology

CO2: Learn the concepts of various digital media

CO3: Know the fundamentals of data compression

CO4: Acquire the knowledge of optical storage media and data and file formats

MODULE 1:

Introduction, Media and Data Streams, Audio Technology

Multimedia Elements; Evolving Technologies for Multimedia Systems; Defining Objects for Multimedia Systems; Multimedia Data Interface Standards; The need for Data Compression; Multimedia Databases. Media: Perception Media, Representation Media, Presentation Media, Storage Media, Transmission Media, Information Exchange Media, Presentation Spaces & Values, and Presentation Dimensions; Key Properties of a Multimedia System: Discrete & Continuous Media, Independence Media, Computer Controlled Systems, Integration; Characterizing Data Streams: Asynchronous Transmission Mode, Synchronous Transmission Mode, Isochronous Transmission Mode; Characterizing Continuous Media Data Streams. Sound: Frequency, Amplitude, Sound Perception and Psychoacoustics; Audio Representation on Computers; Three Dimensional Sound Projection; Music and MIDI Standards; Speech Signals; Speech Output; Speech Input; Speech Transmission.

MODULE 2 :

Graphics and Images, Video Technology, Computer-Based Animation

Capturing Graphics and Images Computer Assisted Graphics and Image Processing; Reconstructing Images; Graphics and Image Output Options. Basics; Television Systems; Digitalization of Video Signals; Digital Television; Basic Concepts; Specification of Animations; Methods of Controlling Animation; Display of Animation; Transmission of Animation; Virtual Reality Modeling Language.

MODULE 3:

Data Compression

Storage Space; Coding Requirements; Source, Entropy, and Hybrid Coding; Basic Compression Techniques; JPEG: Image Preparation, Lossy Sequential DCT-based Mode, Expanded Lossy DCT-based Mode, Lossless Mode, Hierarchical Mode. H.261 (Px64) and H.263: Image Preparation, Coding Algorithms, Data Stream, H.263+ and H.263L; MPEG: Video Encoding, Audio Coding, Data Stream, MPEG-2, MPEG-4, MPEG-7; Fractal Compression.

MODULE 4 :

Optical Storage Media

History of Optical Storage; Basic Technology; Video Discs and Other WORMs; Compact Disc Digital Audio; Compact Disc Read Only Memory; CD-ROM Extended Architecture; Further CD-ROM-Based Developments; Compact Disc Recordable; Compact Disc Magneto-Optical; Compact Disc Read/Write; Digital Versatile Disc.

MODULE 5:

Data and File Format Standards

Rich-Text Format; TIFF File Format; Resource Interchange File Format (RIFF); MIDI 106 File Format; JPEG DIB File Format for Still and Motion Images; AVI Indeo File Format; MPEG Standards; TWAIN

Multimedia Application Design

Multimedia Application Classes; Types of Multimedia Systems; Virtual Reality Design; Components of Multimedia Systems; Organizing Multimedia Databases; Application Workflow Design Issues; Distributed Application Design Issues.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbook:

1. **Ralf Steinmetz, Klara Narstedt:** Multimedia Fundamentals: Vol 1-Media Coding and Content Processing, 2nd Edition, Pearson Education, 2003. (Chapters 2, 3, 4, 5, 6, 7, 8, 9) 2. Prabhat K. Andleigh, Kiran Thakrar: Multimedia Systems Design, PHI, 2003. (Chapters 1, 3, 7)

Reference Book:

1. K.R Rao, Zoran S. Bojkovic and Dragorad A. Milovanovic: Multimedia Communication Systems: Techniques, Standards, and Networks, Pearson Education, 2002. 2. Nalin K Sharad: Multimedia information Networking, PHI, 2002.

INTERNET OF THINGS

Semester	V	CIE Marks : 40
Course Code	18MCA542	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course learning objectives:

This course (18MCA542) will enable student to

- Learn the impact and challenges posed by IoT networks leading to new architectural models.
- Compare and contrast the deployment of smart objects and the technologies to connect them to network.
- Appraise the role of IoT protocols for efficient network communication.
- Understand the need for Data Analytics and Security in IoT.
- Learn different sensor technologies for sensing real world entities

Course Outcomes (CO):

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

CO1: Understand constraints and opportunities of wireless and mobile networks for Internet of Things

CO2: Analyze the societal impact of IoT security events.

CO3: Develop critical thinking skills.

CO4: Analyze, design or develop parts of an Internet of Things solution and map it toward selected business model(s)

CO5: Evaluate ethical and potential security issues related to the Internet of Things.

MODULE -1:

M2M to IoT

Introduction: The Vision-Introduction, From M2M to IoT, M2M towards IoT-the global context, A use case example, Differing Characteristics

MODULE -2:

M2M to IoT- A Market Perspective– Introduction, Some Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT, The international driven global value chain and global information monopolies. M2M to IoT-An Architectural Overview– Building an architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations.

MODULE – 3:

M2M and IoT Technology Fundamentals

Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, Everything as a Service(XaaS), M2M and IoT Analytics, Knowledge Management

MODULE -4:

IoT Architecture-State of the Art

Introduction, State of the art, Architecture Reference Model-Introduction, Reference Model and architecture, IoT reference Model.

IoT Reference Architecture

Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views.

MODULE-5:

Real-World Design Constraints- Introduction, Technical Design constraints-hardware is popular again, Data representation and visualization, Interaction and remote control. Industrial Automation- Service-oriented architecture-based device integration, SOCRADES: realizing the enterprise integrated Web of Things, IMC-AESOP: from the Web of Things to the Cloud of Things, Commercial Building Automation- Introduction, Case study: phase one-commercial building automation today, Case study: phase two-commercial building automation in the future.

Question Paper Pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Book:

1. Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, “From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence”, 1st Edition, Academic Press, 2014.

Reference Books:

1. Vijay Madiseti and ArshdeepBahga, “Internet of Things (A Hands-on-Approach)”, 1st Edition, VPT, 2014.
2. Francis daCosta, “Rethinking the Internet of Things: A Scalable Approach to Connecting Everything”, 1st Edition, Apress Publications, 2013

IMAGE PROCESSING

Semester	V	CIE Marks : 40
Course Code	18MCA543	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course(18MCA543) will enable students to:

- Familiarize basics of images like relationship between pixels, neighbors etc. and types of images like binary, grayscale and color images
- Represent images as multi-dimensional arrays, applying enhancement and filtering techniques
- Understand image transformation techniques like DCT, DFT etc. and Segment the image based on pattern, color etc.
- Applying morphological operations like erosion, dilation, opening, closing on images Understand compression techniques

Course Outcomes:

At the end of the course the students should be able to:

CO1: Explain how digital images are represented and manipulated in a computer, including reading and writing from storage, and displaying.

CO2: Be conversant with the mathematical description of image processing techniques and know how to go from the equations to code.

CO3: Know the image enhancement, segmentation and compression techniques.

MODULE – 1:

Teaching Hours Introduction Fundamental Steps in Digital Image Processing, Components of an Image Processing System, Sampling and Quantization, Representing Digital Images (Data structure), Some Basic Relationships Between Pixels- Neighbors and Connectivity of pixels in image, Applications of Image Processing: Medical imaging, Robot vision, Character recognition, Remote Sensing

MODULE – 2:

Image Enhancement In The Spatial Domain: Some Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement Methods.

MODULE – 3:

Image Enhancement In Frequency Domain: Introduction, Fourier Transform, Discrete Fourier Transform (DFT), properties of DFT , Discrete Cosine Transform (DCT), Image filtering in frequency domain.

MODULE – 4:

Image Segmentation: Introduction, Detection of isolated points, line detection, Edge detection, Edge linking, Region based segmentation- Region growing, split and merge technique, local processing, regional processing, Hough transform, Segmentation using Threshold.

MODULE – 5:

Image Compression: Introduction, coding Redundancy , Inter-pixel redundancy, image compression model, Lossy and Lossless compression, Huffman Coding, Arithmetic Coding, LZW coding, Transform Coding, Sub-image size selection, blocking, DCT implementation using FFT, Run length coding.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Rafael C G., Woods R E. and Eddins S L, Digital Image Processing, Prentice Hall, 3rd edition, 2008.

Reference Books:

1. Milan Sonka, "Image Processing, analysis and Machine Vision", Thomson Press India Ltd, Fourth Edition.
2. Fundamentals of Digital Image Processing- Anil K. Jain, 2nd Edition, Prentice Hall of India.
3. S. Sridhar , Digital Image Processing, Oxford University Press, 2nd Ed, 2016.

PARALLEL COMPUTING

Semester	V	CIE Marks : 40
Course Code	18MCA544	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning objectives:

This Course(18MCA544) enable the students to:

- Understand the need of parallel computing
- Familiarize concepts of Distributed Memory Programming with MPI
- Explore OpenMP and Parallel algorithms

Course Outcomes: At the end of the course the students should be able to:

CO1: Know the fundamentals of parallel processing

CO2: Analyze the hardware and software required for parallel computing

CO3: Understand the distributive memory programming with MPI and OpenMP

MODULE 1:

Introduction to Parallel Computing

Need of Performance, Building Parallel Systems, Why to Write Parallel Programs? How to Write Parallel Programs? Approach: Concurrent, Parallel, Distributed.

Parallel Hardware and Parallel Software

Background, Modifications to the von Neumann Model, Parallel Hardware, Parallel Software, Input and Output, Performance, Parallel Program Design and Writing and Running Parallel Programs

MODULE 2:

Distributed Memory Programming with MPI

Getting Started, The Trapezoidal Rule in MPI, Dealing with I/O, Collective Communication, MPI Derived Data types, A Parallel Sorting Algorithm Shared Memory Programming with Pthreads Processes, Threads and Pthreads, Hello, World program ,Matrix-Vector Multiplication, Critical Sections Busy-Waiting, Mutexes, Producer-Consumer Synchronization and Semaphores, Barriers and

Condition Variables, Read-Write Locks, Caches, CacheCoherence, and False Sharing and Thread-Safety

MODULE 4:

Shared Memory Programming with

Introduction to OpenMP, The Trapezoidal Rule, Scope of Variables, The Reduction Clause, The Parallel For Directive.

MODULE 5:

OpenMP, Parallel Program Development and Parallel Algorithms

More About Loops in OpenMP: Sorting, Scheduling Loops, Producers and Consumers, Caches, Cache-Coherence, and False Sharing and Thread-Safety, Two N-Body Solvers, Tree Search.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbook:

1. An introduction to parallel programming by peter s. Pacheco. 2011. I Edition, morgan kaufmann publishers

Reference Books :

1. Using OpenMP: Portable Shared Memory Parallel Programming, Gabriele Jost and Ruud van der Pas The MIT Press (October 12, 2007)
2. Using MPI - 2nd Edition: Portable Parallel Programming with the Message Passing Interface, William Gropp and Ewing Lusk, 1999, 2nd edition, MIT Press
3. Pthreads Programming: A Posix Standard for Better Multiprocessing, Dick Buttlar, Jacqueline Farrell & Bradford Nichols. 1996, I Edition, Oreilly

SYSTEM SIMULATION AND MODELING

Semester	V	CIE Marks : 40
Course Code	18MCA551	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning objective:

This Course(18MCA551) enable the students to:

- Understand functional modeling.
- Analyze the behavior of a dynamic systems.
- Familiarize statistical sampling techniques and Performance measurement of simulated systems.

Course outcome:

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

CO1: Understand the basics of simulation and various simulation models

CO2: Distinguish different types of random number and random variate generation techniques for solving problems through statistical functions

CO3: Explore verification, validation and optimization on simulation models

CO4: Estimate the performance of system simulation models

CO5: Understand the method of applying computational knowledge to solve specific problems

MODULE 1:

Introduction

When simulation is the appropriate tool and when it is not appropriate; Advantages and disadvantages of Simulation; Areas of application; Systems and system environment; Components of a system; Discrete and continuous systems; Model of a system; Types of Models; Discrete-Event System Simulation; Steps in a Simulation Study. Queuing Models : Characteristics of queuing systems; Queuing notation Simulation Examples: Queuing, Inventory System

MODULE 2:

General Principles and Statistical Models in Simulation :Concepts in Discrete-Event Simulation: The Event-Scheduling / Time-Advance Algorithm, World Views, Manual simulation Using Event

Scheduling; List processing. Statistical Models in Simulation: Review of terminology and concepts; Random Variables, Probability Distribution, Probability distribution function, Useful statistical models; discrete distributions; Continuous distributions;

MODULE 3:

Random-Number Generation, Random-Variate Generation: Properties of random numbers; Generation of pseudo-random numbers; Techniques for generating random numbers; Tests for Random Numbers, Random-Variate Generation: Inverse transformation technique; Acceptance-Rejection technique; Special properties.

MODULE 4:

Input Modeling and Verification and Validation :Data Collection; Identifying the distribution with data; Parameter estimation; Goodness of Fit Tests; Fitting a non-stationary Poisson process; Selecting input models without data; Multivariate and Time-Series input models, uniformity and independence, Chi-Square test, K-S Test. Verification and Validation :Verification, Calibration, and Validation; Optimization: Model building, verification and validation; Verification of simulation models; Calibration and validation of models.

MODULE 5:

Estimation of Absolute Performance & Computer System Simulation 6 Hours Types of simulations with respect to output analysis; Stochastic nature of output data; Absolute measures of performance and their estimation; Output analysis for terminating simulations; Output analysis for steady-state simulations.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.

- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbook:

1. Jerry Banks, John S. Carson II, Barry L. Nelson, David M. Nicol: Discrete-Event System Simulation, 5th Edition, Pearson, 2010. (Listed topics only from Chapters 1 to 12)

Reference Books:

1. **Lawrence M. Leemis, Stephen K. Park:** Discrete – Event Simulation: A First Course, Pearson / Prentice-Hall, 2006.
2. **Averill M. Law:** Simulation Modeling and Analysis, 4th Edition, Tata McGraw-Hill, 2007.
3. Simulation 5ed Ross Elsevier
4. Theory of modeling and simulation, Zeiglar, Elsevier

PRINCIPLES OF USER INTERFACE DESIGN

Semester	V	CIE Marks : 40
Course Code	18MCA552	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course (18MCA552) will enable students to:

- Demonstrate the knowledge of theories and guidelines of designing the user interfaces
- Familiarize the need of interactive design principles and their patterns
- Explore the various interaction styles of user interfaces
- Analyze the user interfaces from both communication perspective and historical perspective

Course Outcome (CO):

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

CO1: use the new technologies that provide interactive devices and interfaces.

CO2: apply the process and evaluate UID.

CO3: understand Direct Manipulation and Virtual Environment

CO4: discuss the command, natural languages and issues in design for maintaining QoS.

CO5: persuade user documentations and information search.

MODULE 1:

Introduction

Usability of Interactive Systems: Introduction, Usability Goals and Measures, Usability Motivation, Universal Usability, Goals for our profession. Guideline, principles, and theories: Introduction, Guidelines, principles, Theories,

MODULE 2:

Development process, evaluating interface

Managing Design Processes: Introduction, Organizational Design to support Usability, The Four Pillars of Design, Development methodologies: Ethnographic Observation, Participatory Design, Scenario Development, Social Impact statement for Early Design Review, Legal Issues. Introduction, Expert Reviews, Usability

Testing and Laboratories, Survey Instruments, Acceptance tests, Evaluation during Active Use,

MODULE 3:

Interaction Styles

Direct Manipulation and Virtual Environments: Introduction, 3D Interfaces, Tele-operation, Virtual and Augmented Reality Menu Selection, Form Filling and Dialog Boxes: Introduction, Task-Related Menu Organization, Single Menus, Combination of Multiple Menus, Content Organization, Fast Movement Through Menus, Data Entry With Menus, Form Filling, Dialog Boxes and Alternatives, Audio Menus and Menus for Small Displays.

MODULE 4:

Command and Natural Languages, Design Issues

Introduction, Command-organization functionality strategies and structure, Naming and Abbreviations, Natural Language in computing. Interaction Devices: Introduction, Keyboards and Keypads, Pointing Devices, Speech and Auditory interfaces, Displays-Small and Large.

Quality of Service: Introduction, Models of Response-Time Impacts, Expectations and Attitudes, User Productivity, Variability in Response time, Frustrating Experiences.

MODULE 5:

User Documentation and Online Help, Information Search and Visualization

Introduction, Online versus paper documentation, Reading from paper versus Displays, Shaping the content of the Manuals, Accessing the Documentation, Online Tutorials and animated demonstrations, Online Communities for User Assistance, The Development Process.

Introduction, Search in Textual Documents and Database Querying, Multimedia document searches, advanced filtering and Search Interfaces, Information Visualization: Introduction, Data type by task taxonomy, Challenges for information visualization.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Ben Shneiderman, Plaisant, Cohen, Jacobs: Designing the User Interface, 5th Edition, Pearson ,Education, 2010

Reference Books:

1. Alan Dix, Janet Finalay, Gregory D AbiwdmRussel Bealel: Human-Computer Interaction, III Edition, Pearson ,Education, 2008.
2. Eberts: User Interface Design, Prentice Hall, 1994
3. Wilber O Galitz: The Essential Guide to User Interface Design- An Introduction to GUI Design, Principles and Techniques, Wiley-Dreamtech India Pvt Ltd, 2011

SOFTWARE ARCHITECTURE

Semester	V	CIE Marks : 40
Course Code	18MCA553	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This course (18MCA553) will enable students to:

- Familiarize different Architectural Structures and views
- Analyze Requirements gathering process
- Implement the different types of software architecture and design patterns

Course Outcome (CO):

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

CO1: Acquire knowledge of

- working principles, characteristics and basic applications of Architectural patterns.
- project life cycle context.
- how the architecture is influenced.
- the quality attributes of architecture.

CO2: Modeling quality attributes through

- check lists.
- experiments.
- back-of-the envelope analysis.

CO3: Understand the techniques of requirements gathering through interviewing stake holders, etc.

CO4: Understand different types of design patterns.

MODULE -1:

Introduction and Context of Software Architecture

What software architecture is and what it is not; Architectural Structures and views; Architectural patterns; What makes a “good” architecture? Why is software important? Technical Context; Project life-cycle context; Business context; Professional context; Stake holders; How is Architecture influenced? What Do Architecture influence?

MODULE -2:

Understanding Quality Attributes

Architecture & Requirements; Functionality; quality attribute considerations; Specifying and achieving Quality attribute requirements; Guiding quality design decisions; Availability; Interoperability; Modifiability; Performance; Security; Testability; Usability.

MODULE – 3:

Quality Attribute modeling and Analysis, Architecture and requirements Gathering

Modeling Architecture to enable quality attribute analysis; Quality attribute check lists; Through experiments and Back-of-the-envelope analysis; Experiments; Simulations and prototypes; Analysis at different stages of the life cycle. Architecture and requirements Gathering ASRs from requirements documents; ASRs by interviewing stake holders; ASRs by understanding the business; capturing ASRs in a utility tree; Typing the methods together.

MODULE – 4:

Designing an Architecture, Documenting Software Architecture and Architecture, Implementation & Testing

Design strategy; the attribute driven design methods; the steps of ADD,

Uses and Audiences for architecture documentation; Notations, View and Behavior; Documentation and quality attributes, Architecture and implementation; Architecture and testing.

MODULE -5:

Architectural Patterns

Introduction to patterns; From Mud to structure; Layers; Pipes and filters; Blackboard; Distributed systems; Broucker; Interactive systems; Model-view-control; Presentation-abstraction- control; Adaptable systems; Microkernel

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks
- There will be 2 full questions (with a maximum of four sub questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
- The students will have to answer 5 full questions, selecting one full question from each Module.

Textbooks:

1. Len Bass, Paul Clements, Rick Kazman: Software Architecture in Practice, 3rd Edition, Pearson Education, 2013 (**Listed Topics only from Chapters 1,2,3,4,5,6,7,8,9,10,11,14,16,17,18,19**)
2. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal: Pattern-Oriented Software Architecture, A System of Patterns, Volume 1, John Wiley and Sons, 2012 (**chapter 2**).

Reference Books:

1. Richard N. Taylor, Nenad Medvidovic and Eric M. Dashofy: Software Architecture: Foundations, Theory, and Practice, Wiley-India 2012
2. Mary Shaw and David Garlan : Software Architecture - Perspectives on an Emerging Discipline, Prentice Hall of India, 2007.

STORAGE AREA NETWORKS

Semester	V	CIE Marks : 40
Course Code	18MCA554	SEE Marks : 60
Contact Periods (L:T:P)	3-0-0	Exam Hours : 03
Credits : 03		

Course Learning Objectives:

This Course (18MCA554) enable the students to:

- Discuss fundamentals of storage networking concepts
- Understand need of Network Attached and Storage Area Networks.
- Familiarize SAN and NAS solutions for enterprise requirement

Course Outcome (CO):

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

CO1: Understand the fundamentals of storage and storage networking concepts

CO2: Analyze Network Attached and Storage Area Networks Requirements

CO3: Apply and Integrate SAN and NAS solutions for an enterprise requirements

CO4: Design a secured, scalable SAN/NAS enterprise solutions

MODULE 1:

Introduction to Information Storage and Management, Storage System Environment

Information Storage, Key Challenges in Managing Information, Information Lifecycle Components of Storage System Environment, Disk Drive Components, Disk Drive Performance, Fundamental Laws Governing Disk Performance, Logical Components of the Host, Application Requirements and Disk Performance.

MODULE 2 :

Data Protection, Intelligent Storage system, Direct-Attached Storage

Implementation of RAID, RAID Array Components, RAID Levels, RAID Comparison, RAID Impact on Disk Performance, Hot Spares Components of an Intelligent Storage System, Intelligent Storage

Array, Types of DAS, DAS Benefits and Limitations.

MODULE 3 :

SCSI, and Storage Area Networks, NAS

Disk Drive Interfaces, Introduction to Parallel SCSI, Overview of Fibre Channel, The SAN and Its Evolution, Components of SAN, FC Connectivity, Fibre Channel Ports, Fibre Channel Architecture, Zoning, Fibre Channel Login Types, FC Topologies. General – Purpose Service vs. NAS Devices, Benefits of NAS,

MODULE 4:

IPSAN, Content-Addressed Storage, Storage Virtualization

NAS File I / O, Components of NAS, NAS Implementations, NAS File-Sharing Protocols, NAS I/O Operations, Factors Affecting NAS Performance and Availability. iSCSI, FCIP. Fixed Content and Archives, Types of Archive, Features and Benefits of CAS, CAS Architecture, Object Storage and Retrieval in CAS, CAS Examples. Forms of Virtualization, SNIA Storage Virtualization Taxonomy, Storage Virtualizations Configurations, Storage Virtualization Challenges, Types of Storage Virtualization.

MODULE 5:

Business Continuity, Backup and Recovery

Information Availability, BC Terminology, BC Planning Lifecycle, Failure Analysis, Business Impact Analysis, BC Technology Solutions. Backup Purpose, Backup Considerations, Backup Granularity, Recovery Considerations, Backup Methods, Backup Process, Backup and restore Operations, Backup Topologies, Backup in NAS Environments, Backup Technologies.

Local Replication, Remote Replication

Source and Target, Uses of Local Replicas, Data Consistency, Local Replication Technologies, Modes of Remote Replication Technologies.

Question paper pattern:

- The question paper will have ten questions.
- Each full question will be for 20 marks.
- There will be 2 full questions (with a maximum of four sub

- questions) from each Module.
- Each full question will have sub questions covering all the topics under a Module.
 - The students will have to answer 5 full questions, selecting one full question from each Module.

Textbook:

1. G. Somasundaram, Alok Shrivastava (Editors): Information Storage and Management: Storing, Managing & Protecting Digital Information in Classic, Visualized and Cloud Environments, 2nd edition, EMC Education Services, WileyIndia, 2009. ISBN 978-1-1180-9483-9

Reference Books:

1. Ulf Troppens, Rainer Erkens and Wolfgang Muller: Storage Networks Explained, Wiley India, 2003.
2. Rebert Spalding: Storage Networks, The Complete Reference, Tata McGraw Hill, 2003.
3. Richard Barker and Paul Massiglia: Storage Area Networks Essentials A Complete Guide to Understanding and Implementing SANs, Wiley India, 2002.

Semester VI

Internship: 18MCA61

Seminar: 18MCA62

Major Project: 18 MCA63

Course Learning Objectives: This course (18MCA61, 62, 63) will enable students to:

- Acquire practical knowledge within the chosen area of technology for project development
- Identify, analyze, formulate and handle programming projects with a comprehensive and systematic approach
- Contribute as an individual or in a team in development of technical projects
- Develop effective communication skills for presentation of project related activities
- Have industrial environment exposure

Course Outcomes(CO):

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

CO1: Identify a suitable problem making use of the technical and engineering knowledge gained from previous courses with the awareness of impact of technology on the society and their ethical responsibilities.

CO2: Ability to segregate work and execute/implement projects using appropriate tools.

CO3: Develop skills to disseminate technical and general information by means of oral as well as written presentation and professional skills.

General Rules

- 1) Students are required to take up individual project in companies/Respective Colleges other than the mini project standards already taken up during previous semesters.
- 2) Project should be real time work, for total of 5 months duration.
- 3) Project work may be application oriented or research oriented as per student and guide's interest. Therefore the project reports will vary depending on whether it is application oriented project or research based project.
- 4) Regular project work weekly dairy should be maintained by the students, signed by the external guide and internal guide in order to verify the regularity of the student. (Enclosing the format.
- 5) Seminars / presentation should be given at Synopsis, SRS, Design and Project Completion levels.
- 6) Project verification at the place of project work must mandatory by the internal guide, for completion of the work.
- 7) If project report is not as per the format and not a real time project, external guides will have every right to reject the project.
- 8) Students are encouraged and appreciated to show their project code demo along with their power point slide show during their viva-voce exams as an added advantage.

(COVER PAGE)

PROJECT TITLE

A Dissertation submitted in partial fulfillment of the requirement for the award of degree of

MASTER OF COMPUTER APPLICATIONS

of

Visvesvaraya Technological University



By

<Student Name>

<USN>

Under the Guidance of

Internal Guide:

External Guide:

<Guide Name>

<Guide Name>

Institute LOGO

Department of Master of Computer Applications,

Institute Name

Address – Pincode

Month Year

(Inner Title Page)

PROJECT TITLE

A Dissertation submitted in partial fulfillment of the requirement
for the award of degree of

MASTER OF COMPUTER APPLICATIONS

of

Visvesvaraya Technological University

By

<Student Name>

<USN>

Under the Guidance of

Internal Guide:

<Guide Name>

<Affiliation>

<Address>

<E-mail>

External Guide:

<Guide Name>

<Affiliation>

<Address>

<E-mail>

Institute LOGO

Department of Master of Computer Applications,

Institute Name

Address – Pincode

Month Year

(Certificate)

Institute LOGO

Department of Master of Computer Applications,

Institute Name

Address – Pincode

CERTIFICATE

This is to certify that <Student Name> has completed his/her
final semester project work entitled “*****”
as a partial fulfillment for the award of Master of Computer
Applications degree, during the academic year 20XX under our joint
supervision.

Signature of Internal Guide

<Guide Name>

<Affiliation>

<Address>

Signature of External Guide

<Guide Name>

<Affiliation>

<Address>

Head of the Department

Principal

Declaration

I, <Name of Student>, student of 6th MCA, Institute Name , bearing USN < USN of student> hereby declare that the project entitled < Project Title> has been carried out by me under the supervision of External Guide < Name of the Guide>, <Designation of Guide> and Internal Guide <Name of the Guide>, <Designation of the Guide> and submitted in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications by the Vivesvaraya Technological University during the academic year 20XX. This report has not been submitted to any other Organization/University for any award of degree or certificate.

Name:

Signature:

Guidelines for the Preparation of Project Reports

1. Printing Area: The margins should be: Left: 1.25”, Right: 1.00”, Top and Bottom-1.00”. The text should be justified to occupy the full line width, so that the right margin is not ragged, with words hyphenated as appropriate. Please fill pages so that the length of the text runs to the right margin.

2. The report must be printed on one side only. Please use a high-resolution printer, preferably a laser printer with at least 300 dpi. Project reports must be printed neatly on one side of the paper on a A4 size bond paper. The reports submitted to the department/guide(s) must be hard bounded with dry tone Xerox.

3. Abstract: The abstract should summarize the contents of the report and should contain at least 150 and at most 350 words. It should be set in 12-point font size. There should be two blank (10-point) lines before and after the title ABSTRACT.

4. Layout, Typeface, Font Sizes, and Numbering: For the main text, please use 12-point type and 1.5 line spacing. We recommend using Times New Roman fonts. Italic type may be used to emphasize words in running text. Bold type and underlining should be avoided.

5. Headings

The chapter headings should be in capitals and must be separated from the other text by 24point line space.

Headings should be in the form where each word is capitalized (i.e., nouns, verbs, and all other words except articles, prepositions, and conjunctions should be set with an initial capital) and should, with the exception of the title, be aligned to the left. The font sizes are given in Table 1. Here are some examples of headings: “Criteria to Disprove Context-Freeness of Collage Languages”, “On Correcting the Intrusion of Tracing Non-deterministic Programs by Software”, “A User-Friendly and Extendable Data Distribution System”, “Multi-flip Networks: Parallelizing GenSAT”, “Self-determinations of Man”.

Table 1 Font sizes of headings. Table captions should always be positioned above the tables. The final sentence of a table caption should end without a period

Heading	Example	Font Size and Style
Title	Chapter 1 Introduction	16 Point Bold
First Level Heading	1.1. Preamble	14 Point Bold
Second Level Heading	2.3.1. Mandatory or Regulatory Signs	12 Point Bold
Third Level Heading	Stop and Give away signs	12 Point Bold
Fourth Level Heading	<i>Creation of database</i>	12 Point Bold Italicized

Figures and Photographs

Check that in line drawings, lines are not interrupted and have constant width. Grids and details within the figures must be clearly readable and may not be written one on top of the other. The lettering in figures should have a height of 2 mm (10-point type). Figures should be scaled up or down accordingly.

Figures should be numbered and should have a caption which should always be positioned under the figures, in contrast to the caption belonging to a table, which should always appear above the table. Please center the captions between the margins and set them in 9-point type (Fig. 1 shows an example). The distance between text and figure should be about 12 point spacing, the distance between figure and caption about 6 point spacing.

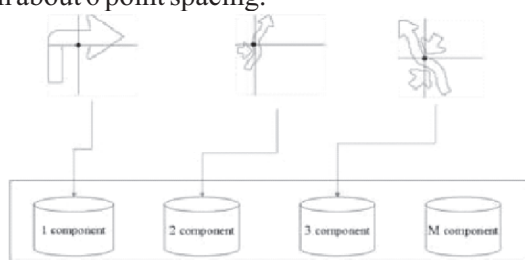


Fig 1.1. The last sentence of a figure caption should generally end without a period.

6. Formulas

Displayed equations or formulas are centered and set on a separate line (with an extra line or halfline space above and below). Displayed expressions should be numbered for reference. The numbers should be consecutive within each section or within the contribution, with numbers enclosed in parentheses and set on the right margin. For example, a correlation matrix is computed using equation (5),

$$c_{ij} = \frac{d_{ij}(\Sigma AB) - (\Sigma A)(\Sigma B)}{\sqrt{d_{ij}(\Sigma A^2) - (\Sigma A)^2} \sqrt{d_{ij}(\Sigma B^2) - (\Sigma B)^2}} \quad (5)$$

7. Program Code

Program listings or program commands or algorithms in the text are normally set in typewriter font, e.g., CMTT10 or Courier.

Example of an Algorithm is

Algorithm-1: Database Creation (Mean and Standard Deviation based approach)

Input: Static images of potential traffic sign

Output: Database created.

Methodology:

For each input image do

Step1: Preprocess the image as explained in section 4.3.1

Step2: Calculate the number of components in a sign as explained in section 4.3.1.

Step3: calculate a feature vector as mentioned in section 4.3.2.1.

Step4: Store the feature vector computed in step 3 in the corresponding database, based on number of components present in the sign. For End.

Algorithm End.

8. Footnotes/ Header

Footnotes/Header should appear at the bottom of the normal text area, with a line of about 5 cm in Word set immediately below/above the text.

Header sample: (Project title is left aligned and page number is right aligned)

<<Project Title>>

<<Page Number>>

<College Name>

Department of MCA

2018-2019

9. The list of references is headed “References” and is assigned a number with square brackets in the decimal system of headings. The list should be set in small print and placed at the end of the dissertation, in front of the appendix, if any exists. Please do not insert a page break before the list of references if the page is not completely filled. An example is given at the end of this information sheet. For citations in the text please use square brackets and consecutive numbers: [1], [2], [3] etc.

10. Page Numbering

Reports must be printed with page numbers on the top right corner.

11. The total number of reports to be prepared are three

- One copy to the concerned guide
- One copy for University
- One copy to candidate
- Two CD's having soft copy of Project report (for department

purpose)

12. Before taking the final printout, the approval of the concerned guide is mandatory and suggested corrections, if any, must be incorporated.

13. Every copy of the report must contain (See formats towards the end of this document)

- Outer title page (parrot green) with a plastic cover
- Inner title page (White)
- Certificate in the format enclosed, only certificate will be signed by following:
 - o Principal
 - o HOD
 - o Internal guide and External guide (if project is carried out in company)
 - o Guide and/or Co-guide (if project is carried out in college)

14. The organization of the report should be as follows

- o Inner title page
- o Certificate
- o Project Completion certificate from Company / College
- o Declaration (by student)
- o Acknowledgement
- o Abstract
- o Table of Contents
- o List of table and figures

Contents (for Application Oriented Projects)

Contents (for Application Oriented Projects)

1. Introduction	00
1.1 Project Description (2-4 Pages)	00
1.2 Company Profile (1-2-3 Pages)	00
2. Literature Survey	00
2.1 Existing And Proposed System (2-3 Pages)	00
2.2 Feasibility Study (2-3 Pages)	00
2.3 Tools And Technologies Used (2-4 Pages)	00
2.4 Hardware And Software Requirements (1 Page)	00
3. Software Requirements Specification	00
3.1 Users (2-3 Pages)	00
3.2 Functional Requirements (2-3 Pages)	00
3.3 Non-functional Requirements (2-3 Pages)	00
4. System Design (high Level Or Architectural Design)	00
4.1 System Perspective (1-2 Pages)	00
4.2 Context Diagram (1-2 Pages)	00
5. Detailed Desing	00
(various Design Diagrams According To Project)	00
5.1 Use Case Diagram (4-6 Pages)	00
5.2 Sequence Diagrams (4-6 Pages)	00
5.3 Collabration Diagrams (3-5 Pages)	00
5.4 Activity Diagram (4-6 Pages)	00
5.5 Database Desing (er And/or Conceptual Schema)	00
(3-4 Pages)	00
6. Implementation	00
(no Full Code, Code Snippet May Be Included)	00
6.1 Screen Shots (15-20 Pages)	00
7. Software Testing (test Cases Etc.) (6-8 Pages)	00
8. Conclusion (1 Page)	00
9. Future Enhancements (1 Page)	00
Appendix A	Bibliography (1 Page) 00
Appendix B	User Manual (2-10 Pages) 00

Note: For research oriented or other type of projects table of content structure may be altered as per need.