CHADALAWADA RAMANAMMA ENGINEERING COLLEGE

(Autonomous)

(Approved by AICTE | NAAC Accreditation with 'A' Grade | Permanently Affiliated to JNTUA)
Chadalawada Nagar, Tirupati - 517506, Andhra Pradesh.

OUTCOME BASED EDUCATION
WITH
CHOICE BASED CREDIT SYSTEM

MASTER OF COMPUTER APPLICATIONS

ACADEMIC REGULATIONS, COURSE STRUCTURE AND SYLLABI UNDER AUTONOMOUS STATUS

M.C.A Regular Three Year Degree Program
(for the batches admitted from the academic year 2017 - 18)

CONTENTS

	Preliminary Definitions and Nomenclatures & Foreword	i - iii
1	Choice Based Credit System	01
2	Medium of Instruction	01
3	Eligibility for Admission	02
4	Unique course identification code	02
5	Types of Courses	02
6	Semester Structure	03
7	Program Duration	04
8	Curriculum and Course structure	04
9	Evaluation Methodology	05
10	Attendance Requirements and Detention Policy	07
11	Conduct of Semester End Examinations and Evaluation	07
12	Scheme for the Award of Grade	08
13	Letter Grades and Grade Points	08
14	Computation of SGPA and CGPA	08
15	Illustration of Computation of SGPA and CGPA	09
16	Graduation Requirements	10
17	Award of Degree	10
18	Termination from the Program	11
19	With-holding of Results	11
20	Graduation Day	11
21	Discipline	11
22	Grievance Redressal Committee	11
23	Transitory Regulations	11
24	Revision of Regulations and Curriculum	12
25	General	12
26	Malpractices Rules	12
27	Course Structure of M.C.A	14
28	Syllabus	18
29	M.C.A - Program Outcomes (POs)	117
30	Frequently asked Questions and Answers about autonomy	118

PRELIMINARY DEFINITIONS AND NOMENCLATURES

Academic Council: The Academic Council is the highest academic body of the institute and is responsible for the maintenance of standards of instruction, education and examination within the institute. Academic Council is an authority as per UGC regulations and it has the right to take decisions on all academic matters including academic research.

Academic Autonomy: Means freedom to an institute in all aspects of conducting its academic programs, granted by UGC for Promoting Excellence.

Academic Year: It is the period necessary to complete an actual course of study within a year. It comprises two consecutive semesters i.e., Even and Odd semester.

AICTE: Means All India Council for Technical Education, New Delhi.

Autonomous Institute: Means an institute designated as autonomous by University Grants Commission (UGC), New Delhi in concurrence with affiliating University (Jawaharlal Nehru Technological University, Hyderabad) and State Government.

Backlog Course: A course is considered to be a backlog course if the student has obtained a failure grade (F) in that course.

Board of Studies (BOS): BOS is an authority as defined in UGC regulations, constituted by Head of the Organization for each of the departments separately. They are responsible for curriculum design and updation in respect of all the programs offered by a department.

Choice Based Credit System: The credit based semester system is one which provides flexibility in designing curriculum and assigning credits based on the course content and hours of teaching along with provision of choice for the student in the course selection.

Compulsory course: Course required to be undertaken for the award of the degree as per the program.

Commission: Means University Grants Commission (UGC), New Delhi.

Continuous Internal Examination: It is an examination conducted towards internal assessment.

Course: A course is a subject offered by the University for learning in a particular semester.

Course Outcomes: The essential skills that need to be acquired by every student through a course.

Credit: A credit is a unit that gives weight to the value, level or time requirements of an academic course. The number of 'Contact Hours' in a week of a particular course determines its credit value. One credit is equivalent to one lecture hour per week.

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

Cumulative Grade Point Average (CGPA): It is a measure of cumulative performance of a student over all the completed semesters. The CGPA is the ratio of total credit points secured 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.

Curriculum: Curriculum incorporates the planned interaction of students with instructional content, materials, resources and processes for evaluating the attainment of Program Educational Objectives.

Degree with Specialization: A student who fulfills the entire program requirements of her/his discipline and successfully completes a specified set of professional course is eligible to receive a degree with specialization.

Department: An academic entity that conducts relevant curricular and co-curricular activities, involving both teaching and non-teaching staff and other resources in the process of study for a degree.

Detention in a course: Student who does not obtain minimum prescribed attendance in a course shall be detained in that particular course.

Elective Course: An Elective can be chosen from a set of Professional Electives.

Evaluation: Evaluation is the process of judging the academic performance of the student in her/his courses. It is done through a combination of continuous internal assessment and semester end examinations.

Grade: It is an index of the performance of the students in a said course. Grades are indicated by alphabets.

Grade Point: It is a numerical weight allotted to each letter grade on a 10 point scale.

Institute: Means Chadalawada Ramanamma Engineering College, Tirupati unless indicated otherwise by the context.

Pre-requisite: A course, the knowledge of which is required for registration into higher level course.

Core: The courses that are essential constituents of each engineering discipline are categorized as professional core courses for that discipline.

Professional Elective: A course that is discipline centric. An appropriate choice of minimum number of such electives as specified in the program will lead to a degree with specialization.

Program: Means, Master of Computer Applications

Program Educational Objectives: The broad career, professional and personal goals that every student will achieve through a strategic and sequential action plan.

Project work: It is a design or research based work to be taken up by a student during his/her second year to achieve a particular aim. It is a credit based course and is to be planned carefully by the student.

Re-Appearing: A student can reappear only in the semester end examination for the theory component of a course, subject to the regulations contained herein.

Registration: Process of enrolling into a set of courses in a semester of a Program.

Regulations: The regulations for M.C.A programs offered by Institute are designated as “CREC-R17” and are binding on all the stakeholders.

Semester: It is a period of study consisting of 19 to 21 weeks of academic work equivalent to normally 90 working days. The odd semester starts usually in July and even semester in December.

Semester End Examinations: It is an examination conducted for all courses offered in a semester at the end of the semester.

S/he: Means “she” and “he” both.

Student Outcomes: The essential skill sets that need to be acquired by every student during her/his program of study. These skill sets are in the areas of employability, entrepreneurial, social and behavioral.

University: Means the Jawaharlal Nehru Technological University Anantapuramu, Anantapuramu.

Withdraw from a Course: Withdrawing from a course means that a student can drop from a course within the first two weeks of the odd or even semester (deadlines are different for summer sessions). However s/he can choose a substitute course in place of it by exercising the option within 5 working days from the date of withdrawal.

FOREWORD

The autonomy is conferred to Chadalawada Ramanamma Engineering College (CREC), Tirupati by University Grants Commission (UGC), New Delhi based on its performance as well as future commitment and competency to impart quality education. It is a mark of its ability to function independently in accordance with the set norms of the monitoring bodies like J N T University Anantapuramu (JNTUA), Anantapuramu and AICTE. It reflects the confidence of the affiliating University in the autonomous institution to uphold and maintain standards it expects to deliver on its own behalf and thus awards degrees on behalf of the college. Thus, an autonomous institution is given the freedom to have its own **curriculum, examination system** and **monitoring mechanism**, independent of the affiliating University but under its observance.

CREC is proud to win the credence of all the above bodies monitoring the quality in education and has gladly accepted the responsibility of sustaining, if not improving upon the standards and ethics for which it has been striving for more than a decade in reaching its present standing in the arena of contemporary technical education. As a follow up, statutory bodies like Academic Council and Boards of Studies are constituted with the guidance of the Governing Body of the institute and recommendations of the JNTUA to frame the regulations, course structure and syllabi under autonomous status.

The autonomous regulations, course structure and syllabi have been prepared after prolonged and detailed interaction with several expertise solicited from academics, industry and research, in accordance with the vision and mission of the institute to order to produce a quality Master of Computers Applications professionals to the society.

All the faculty, parents and students are requested to go through all the rules and regulations carefully. Any clarifications needed are to be sought at appropriate time and with principal of the college, without presumptions, to avoid unwanted subsequent inconveniences and embarrassments. The Cooperation of all the stake holders is sought for the successful implementation of the autonomous system in the larger interests of the college and brighter prospects of Master of Computers Applications professionals.

PRINCIPAL

CHADALAWADA RAMANAMMA ENGINEERING COLLEGE

(Autonomous)

ACADEMIC REGULATIONS

M.C.A Regular Three Year Degree Program
(for the batches admitted from the academic year 2017 - 18)

For pursuing three year postgraduate Master Degree program of study in computer applications offered by Chadalawada Ramanamma Engineering College under Autonomous status and herein after referred to as CREC.

1.0 CHOICE BASED CREDIT SYSTEM

The Indian Higher Education Institutions (HEI's) are changing from the conventional course structure to Choice Based Credit System (CBCS) along with introduction to semester system at first year itself. The semester system helps in accelerating the teaching learning process and enables vertical and horizontal mobility in learning.

The credit based semester system provides flexibility in designing curriculum and assigning credits based on the course content and hours of teaching. The choice based credit system provides a "Cafeteria" type approaches, in which the students can take courses of their choice, learn at their own pace, undergo additional courses and acquire more than the required credits and adopt an interdisciplinary approach to learning.

Choice Based Credit System (CBCS) is a flexible system of learning and provides choice for students to select from the prescribed elective courses. A course defines learning objectives and learning outcomes and comprises of lectures / tutorials / laboratory work / field work / project work / comprehensive examination / viva / seminars / assignments / presentations / self-study etc. or a combination of some of these.

Under the CBCS, the requirement for awarding a degree is prescribed in terms of number of credits to be completed by the students.

The CBCS permits students to:

1. Choose electives from a wide range of elective courses offered by the department of the Institute.
2. Undergo additional courses of interest.
3. Adopt an inter-disciplinary approach in learning.
4. Make the best use of expertise of the available faculty.

2.0 MEDIUM OF INSTRUCTION

The medium of instruction shall be English for all courses, examinations, seminar presentations and project work. The curriculum will comprise courses of study as given in course curriculum in accordance with the prescribed syllabi.

3.0 ELIGIBILITY FOR ADMISSION

The admissions for category A and B seats shall be as per the guidelines of Andhra Pradesh State Council for Higher Education (APSCHE) in consonance with government reservation policy.

a) Under Category A: 70% of the seats are filled based on ICET ranks.

b) Under Category B: 30% seats are filled on merit basis as per guidelines of APSCHE.

4.0 UNIQUE COURSE IDENTIFICATION CODE: F0

5.0 TYPES OF COURSES

Courses in a programme may be of two kinds: **Core and Elective.**

5.1 Core Course:

There may be a core course in every semester. This is the course which is to be compulsorily studied by a student as a core requirement to complete the requirement of a programme in said discipline of study.

5.2 Elective Course:

Electives provide breadth of experience in respective branch and applications areas. Elective course is a course which can be chosen from a pool of courses. It may be:

□ Supportive to the discipline of study

□ Providing an expanded scope

□ Enabling an exposure to some other discipline/domain

□ Nurturing student’s proficiency/skill.

An elective may be discipline centric (Professional Elective) focusing on those courses which add generic proficiency to the students.

There shall be two professional elective groups out of which students can choose not more than two courses from each group. Overall, students can opt for two professional elective courses which suit their project work in consultation with the faculty advisor/mentor.

6.0 SEMESTER STRUCTURE

The institute shall follow semester pattern. An academic year shall consist of a first semester and a second semester. Each semester shall be of 21weeks (Table 2) duration and this period includes time for course work, examination preparation and conduct of examinations. Each main semester shall have a minimum of 90 working days; out of which number of contact days for teaching / practical shall be 83 and 7 days shall be for examination preparation. The duration for each semester shall be a minimum of 16 weeks of instructions. The Academic Calendar is declared at the beginning of the academic year as given in below Table 1.

Table 1: Academic Calendar

I-YEAR FIRST SEMESTER (21 weeks)	I Spell Instructions	8 weeks	19 weeks
	I Mid Term Examinations	5 days	
	II Spell Instructions	8 weeks	
	II Mid Term Examinations	5 days	
	Preparation and Practical Examinations	1 week	
	Semester End Examinations		2 weeks
Semester Break			1 week
I-YEAR SECOND SEMESTER (21 weeks)	I Spell Instructions	8 weeks	19 weeks
	I Mid Term Examinations	5 days	
	II Spell Instructions	8 weeks	
	II Mid Term Examinations	5 days	
	Preparation & Practical Examinations	1 week	
	Semester End Examinations		2 weeks
Summer Vacation			4 weeks
II-YEAR FIRST SEMESTER (21 weeks)	I Spell Instructions	8 weeks	19 weeks
	I Mid Term Examinations	5 days	
	II Spell Instructions	8 weeks	
	II Mid Term Examinations	5 days	
	Preparation and Practical Examinations	1 week	
	Semester End Examinations		2 weeks
Semester Break			1 week
II-YEAR SECOND SEMESTER (21 weeks)	I Spell Instructions	8 weeks	19 weeks
	I Mid Term Examinations	5 days	
	II Spell Instructions	8 weeks	
	II Mid Term Examinations	5 days	
	Preparation & Practical Examinations	1 week	
	Semester End Examinations		2 weeks
Summer Vacation			4 weeks
III-YEAR FIRST SEMESTER (21 weeks)	I Spell Instructions	8 weeks	19 weeks
	I Mid Term Examinations	5 days	
	II Spell Instructions	8 weeks	
	II Mid Term Examinations	5 days	
	Preparation and Practical Examinations	1 week	
	Semester End Examinations		2 weeks
Semester Break			
III-YEAR	Project Work Phase - I		--
	Project Work Phase - II		--

SECOND SEMESTER (21 weeks)	Pre-submission seminar	--

7.0 PROGRAM DURATION

A student shall be declared eligible for the award of M.C.A degree, if s/he pursues a course of study and completes it successfully in not less than three academic years and not more than six academic years. A student, who fails to fulfill all the academic requirements for the award of the degree within three academic years from the year of his/her admission, shall forfeit his/her seat in M.C.A course.

a) A student will be eligible for the award of M.C.A degree on securing a minimum of 5.0/10.0 CGPA.

b) In the event of non-completion of project work and/or non-submission of the project report by the end of the final semester, the candidate shall re-register by paying the semester fee for the project. In such a case, the candidate will not be permitted to submit the report earlier than three months and not later than six months from the date of registration.

8.0 CURRICULUM AND COURSE STRUCTURE

The curriculum shall comprise Core Courses, Elective Courses (Professional/open), Laboratory Course, Internship and Project Work. The list of elective courses may include subjects from allied disciplines also.

Each Theory and Laboratory course carries credits based on the number of hours/week as follows:

□ **Lecture Hours (Theory):** 1 credit per lecture hour per week.

□ **Laboratory Hours (Practical):** 1 credit for 2 practical hours, 2 credits for 3 or 4 practical hours per week.

□ **Project Work:** 1 credit for 4 hours of project work per week.

8.1 Credit distribution for courses offered is shown in Table2.

Table 2: Credit distribution

S. No	Course	Hours	Credits
1	Core Courses	4	4
2	Elective Courses	4	4
3	Laboratory Courses	3	2
4	Seminar	--	2
5	Project Work	--	8

8.2 Course wise break-up for the total credits:

Total Theory Courses (12) Core Courses (21) + Professional Electives (04)	25 @ 4 credits	100
Total Laboratory Courses (15)	15 @ 2 credits	30

Seminar (01)	1 @ 2 credits	02
Project Work	1 @ 8 credits	08
TOTAL CREDITS		**140**

9.0 EVALUATION METHODOLOGY

Theory Course:

Each theory course will be evaluated for a total of 100 marks, with 40 marks for Continuous Internal Assessment (CIA) and 60 marks for Semester End Examination (SEE).

9.1.1 Semester End Examination (SEE):

The SEE shall be conducted for 60 marks of 3 hours duration. The syllabus for the theory courses shall be divided into FIVE units and each unit carries equal weightage in terms of marks distribution. The question paper pattern shall be as defined below. Two full questions with "either" or choice will be drawn from each unit. Each question carries 12 marks. There could be a maximum of three sub divisions in a question.

The emphasis on the questions is broadly based on the following criteria:

50 %	To test the objectiveness of the concept
30 %	To test the analytical skill of the concept
20 %	To test the application skill of the concept

Continuous Internal Assessment (CIA):

For each theory course the CIA shall be conducted by the faculty/teacher handling the course as given in Table 3. CIA is conducted for a total of 40 marks.

Table 3: Assessment pattern for Theory Courses

COMPONENT	THEORY	TOTAL MARKS
Type of Assessment	CIE Exam (Sessional)	
Max. CIA	40	40

Continuous Internal Examination (CIE):

Two Internal Examinations shall be conducted at the end of the 9th and 17th week of the semester respectively. The Internal Examination is conducted for 40 marks of 2 hours duration, consisting of 5 questions, the student has to answer any 4 questions, each question shall be of equal weightage. For a total of 40 marks, 80% of marks from better one of the two and 20% of marks from the other one are added and finalized.

9.2 Laboratory Course:

9.2.1 Each lab will be evaluated for a total of 100 marks consisting of 40 marks for internal assessment and 60 marks for semester end lab examination. Out of 40 marks of internal assessment, continuous lab assessment will be done for 30 marks for the day to day performance and 10 marks for the final internal lab assessment.

The semester end lab examination for 60 marks shall be conducted by two examiners, one of them being a internal examiner and another is external examiner, both nominated by Head of the Department.

9.3 Project Work

Every candidate shall be required to submit thesis or dissertation after taking up a topic approved by the college/ institute. Registration of Project work: A candidate is permitted to register for the project work after satisfying the attendance requirement of all the courses (theory and practical courses of I to V Sem)

An Internal Departmental Committee (I.D.C) consisting of HOD, Supervisor and one internal senior expert shall monitor the progress of the project work. The work on the project shall be initiated in the penultimate semester and continued in the final semester. The candidate can submit project thesis with the approval of I.D.C. at the end of the VI semester Instruction as per the schedule. Extension of time within the total permissible limit for completing the programme is to be obtained from the Head of the Institution.

The student must submit status report at least in two different phases during the project work period. These reports must be approved by the I.D.C before submission of the Project Report. The viva-voce examination may be conducted for all the candidates as per the VI semester examination schedule. Three copies of the Thesis / Dissertation certified in the prescribed form by the supervisor and HOD shall be presented to the HOD. One copy is to be forwarded to the University and one copy to be sent to the examiner.

The Head of the Department shall submit a panel of three experts for a maximum of every 15 students. However, the viva-voce examiners will be nominated by the Principal. If the report of the examiner is favorable viva-voce examination shall be conducted by a board consist of the Supervisor, Head of the Department and the examiner who adjudicated the thesis / dissertation.

The board shall jointly report candidates work as:

1.	Excellent	Grade	A
2.	Good	Grade	B
3.	Satisfactory	Grade	C
4.	Not satisfactory	Grade	D

If the report of the viva-voce is not satisfactory (Grade D) the candidate will retake the viva-voce examination after three months. If he fails to get a satisfactory report at the second viva-voce examination, he will not be eligible for the award of the degree unless the candidate is permitted to revise and resubmit the thesis.

10.0 ATTENDANCE REQUIREMENTS AND DETENTION POLICY

10.1 It is desirable for a candidate to put on 100% attendance in each course. In every course (theory/laboratory), student has to maintain a minimum of 75% attendance including the days of attendance in sports, games, NCC and NSS activities to be eligible for appearing in Semester End Examination of the course.

10.2 For cases of medical issues, deficiency of attendance in each course to the extent of 10% may be condoned by the College Academic Committee (CAC) on the recommendation of Head of the Department if his/her attendance is between 75% to 65%, subjected to submission of medical certificate and other needful documents to the department concerned.

10.3 The basis for the calculation of the attendance shall be the period prescribed by the institute by its calendar of events. For late admission, attendance is reckoned from the date of admission to the program.

10.4 A prescribed fee shall be payable towards Condonation of shortage of attendance.

10.5 However, in case of a student having less than 65% attendance, s/he shall be detained in the course and in no case such process will be relaxed.

10.6 If the candidate doesn’t satisfy the attendance requirement he is detained for want of attendance and shall reregister for that semester. He / she shall not be promoted to the next semester.

11.0 CONDUCT OF SEMESTER END EXAMINATIONS AND EVALUATION

11.1 Semester End Examination shall be conducted by the Controller of Examinations (COE) by inviting Question Papers from the External Examiners.

11.2 Question papers may be moderated for the coverage of syllabus, pattern of questions by Semester End Examination Committee chaired by Head of the Department one day before the commencement of Semester End Examinations. Internal Examiner shall prepare a detailed scheme of evaluation.

11.3 The answer papers of semester end examination should be evaluated by the internal examiner immediately after the completion of exam and the award sheet should be submitted to COE in a sealed cover and evaluated by the external examiner.

11.4 In case of difference is more than 15% of marks, the answer paper shall be re-evaluated by a third examiner appointed by the Examination Committee and marks awarded by him/her shall be taken as final.

11.5 COE shall invite required number of external examiners to evaluate all the end semester answer scripts on a prescribed date(s).

11.6 Examination Control Committee shall consolidate the marks awarded by internal and external examiners to award grades.

12.0 SCHEME FOR THE AWARD OF GRADE

12.1 A student shall be deemed to have satisfied the minimum academic requirements and earn the credits for each theory course, if s/he secures:

I. Not less than 40% marks for each theory course in the semester end examination, and

II. A minimum of 50% marks for each theory course considering both CIA and SEE.

12.2 A student shall be deemed to have satisfied the minimum academic requirements and earn the credits for each Laboratory if s/he secures

I. Not less than 40% marks for each Laboratory in the semester end examination,

II. A minimum of 50% marks for each Laboratory considering both internal and semester end examination.

12.3 If a candidate fails to secure a pass in a particular course, it is mandatory that s/he shall register and reappear for the examination in that course during the next semester when examination is conducted in that course. It is mandatory that s/he should continue to register and reappear for the examination till s/he secures a pass.

13.0 LETTER GRADES AND GRADE POINTS

13.1 Performances of students in each course are expressed in terms of marks as well as in Letter Grades based on absolute grading system. The UGC recommends a 10 point grading system with the following letter grades as given below:

Range of Marks	Grade Point	Letter Grade
90 – 100	10	S (Superior)
80 – 89	9	A+ (Excellent)
70 – 79	8	A (Very Good)
60 – 69	7	B+ (Good)
55 – 59	6	B (Average)
50 – 54	5	P (Pass)
Below 50	0	F (Fail)
Absent	0	Ab (Absent)
Authorized Break of Study	0	ABS

13.2 A student obtaining Grade F shall be declared as failed and will be required to reappear in the examination.

13.3 At the end of each semester, the institute issues grade sheet indicating the SGPA and CGPA of the student. However, grade sheet will not be issued to the student if she/he has any outstanding dues.

14.0 COMPUTATION OF SGPA AND CGPA

The UGC recommends computing the Semester Grade Point Average (SGPA) and Cumulative Grade Point Average (CGPA). The credit points earned by a student are used for calculating the Semester Grade Point Average (SGPA) and the Cumulative Grade Point Average (CGPA), both of which are important performance indices of the student. SGPA is equal to the sum of all the total points earned by the student in a given semester divided by the number of credits registered by the student in that semester. CGPA gives the sum of all the total points earned in all the previous semesters and the current semester divided by the number of credits registered in all these semesters. Thus,

$$SGPA = \sum_{I=1}^{N} (C_I G_I) / \sum_{I=1}^{N} C_I$$

Where, C_i is the number of credits of the i^{th} course and G_i is the grade point scored by the student in the i^{th} course and n represent the number of courses in which a student is registered in the concerned semester.

$$CGPA = \sum_{J=1}^{M} \left(C_J S_J\right) / \sum_{J=1}^{M} C_J$$

Where, S_j is the SGPA of the j^{th} semester and C_j is the total number of credits up to the semester and m represent the number of semesters completed in which a student registered up to the semester.

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

15.0 ILLUSTRATION OF COMPUTATION OF SGPA AND CGPA

15.1 Illustration for SGPA

Course Name	Course Credits	Grade letter	Grade point	Credit Point (Credit x Grade)
Course 1	4	A	8	4 x 8 = 32
Course 2	4	B+	7	4 x 7 = 28
Course 3	4	B	6	4 x 6 = 24
Course 4	4	S	10	4 x 10 = 40
Course 5	4	P	5	4 x 5 = 20
	20			144

Thus, SGPA = 144/ 20 = 7.2

15.2 Illustration for CGPA

Semester 1	Semester 2	Semester 3	Semester 4	Semester 5	semester 6
Credit: 20 SGPA: 7.2	Credit: 20 SGPA: 7.2	Credit: 20 SGPA: 7.2	Credit: 20 SGPA: 7.2	Credit: 20 SGPA: 7.2	Credit: 10 SGPA: 5.6

$$Thus, CGPA = \frac{26 x 6.9 + 26 x 7.8 + 26 x 5.6 + 26 x 6.0 + 26 x 6.4 + 10 x 6.2}{140} = 6.51$$

16.0 GRADUATION REQUIREMENTS

The following academic requirements shall be met for the award of M.C.A degree.

16.0 Student shall register and acquire minimum attendance in all courses.

16.1 Student, who fails to secure a minimum CGPA of 5.0 from all the semesters from his /her admission, shall forfeit his/her degree and his/her admission stands cancelled.

17.0 AWARD OF DEGREE

The degree shall be conferred and awarded by Jawaharlal Nehru Technological University Anantapur, Ananthapuramu on the recommendations of the Chairman, Academic Council of CREC(Autonomous).

17.1 AWARD OF DIVISION:

Classification of degree will be as follows:

CGPA ≥ 7.5	**CGPA ≥ 6.5 and < 7.5**	**CGPA ≥ 5.5and < 6.5**	**CGPA ≥ 5.0and < 5.5**	**CGPA < 5.0**
First Class with Distinction	First Class	Second Class	Pass Class	Fail

17.1.1 In case a student takes more than one attempt in clearing a course, the final marks secured shall be indicated by * mark in the grade sheet.

17.1.2 All the candidates who register for the semester end examination will be issued grade sheets by the Institute. Apart from the semester wise grade sheet, the institute will issue the provisional certificate subject to the fulfillment of all the academic requirements.

17.1.3 In order to extend the benefit to the students with one/two backlogs after either IV semester or VI semester, GRAFTING option is provided to the students enabling their placements and fulfilling graduation requirements. Following are the guidelines for Grafting:

a. Grafting will be done among the courses within the semester shall draw a maximum of 7 marks from the any one of the cleared courses in the semester and will be grafted to the failed course in the same semester.

b. Students shall be given a choice of grafting only once in the 3 years program, either after IV semester (Option #1) or after V semester (Option #2).

c. Option#1: Applicable to students who have maximum of TWO theory courses in III and / or IV semesters.

   Option#2: Applicable to students who have maximum of TWO theory courses in V semester.

d. Eligibility for grafting:

   I. Prior to the conduct of the supplementary examination after the declaration of IV or V semester results.

   II. S/he must appear in all regular or supplementary examinations as per the provisions laid down in regulations for the courses s/he appeals for grafting.

   III. The marks obtained by her/him in latest attempt shall be taken into account for grafting of marks in the failed course(s).

17.1.4 Student, who clears all the courses up to IV semester, shall have a chance to appear for Quick Supplementary Examination to clear the failed courses of V semester.

17.1.5 By the end of VI semester, all the students (regular and lateral entry students) shall complete one of the audit course and mandatory course with acceptable performance.

17.1.6 All the candidates who register for the semester end examination will be issued grade sheet by the institute. Apart from the semester wise grade sheet, the institute will issue the provisional certificate and consolidated grade sheet subject to the fulfillment of all the academic requirements.

18.0 TERMINATION FROM THE PROGRAM

The admission of a student to the program may be terminated and the student may be asked to leave the institute in the following circumstances:

18.1.1 The student fails to satisfy the requirements of the program within the maximum period stipulated for that program.

18.1.2 The student fails to satisfy the norms of discipline specified by the institute from time to time.

19.0 WITH-HOLDING OF RESULTS

If the candidate has not paid any dues to the college / if any case of indiscipline / malpractice is pending against him/her, the results of the candidate will be withheld. The issue of the degree is liable to be withheld in such cases.

20.0 GRADUATION DAY

The institute shall have its own Annual Graduation Day for the award of Degrees to students completing the prescribed academic requirements in each case, in consultation with the University and by following the provisions in the Statute.

The college shall institute prizes and medals to meritorious students annually on Graduation Day. This will greatly encourage the students to strive for excellence in their academic work.

21.0 DISCIPLINE

Every student is required to observe discipline and decorum both inside and outside the institute and not to indulge in any activity which will tend to bring down the honor of the institute. If a student indulges in malpractice in any of the theory / practical examination, continuous assessment examinations he/she shall be liable for punitive action as prescribed by the Institute from time to time.

22.0 GRIEVANCE REDRESSAL COMMITTEE

The institute shall form a Grievance Redressal Committee for each course in each department with the Course Teacher and the HOD as the members. This Committee shall solve all grievances related to the course under consideration.

23.0 TRANSITORY REGULATIONS

23.1 A student who has been detained in any semester of previous regulations for not satisfying the attendance requirements shall be permitted to join in the corresponding semester of this regulation.

23.2 Semester End Examination in each course under the regulations that precede immediately these regulations shall be conducted three times after the conduct of last regular examination under those regulations. Thereafter, the failed students, if any, shall take examination in the equivalent papers of these regulations as suggested by the Chairman, BOS concerned.

24.0 REVISION OF REGULATIONS AND CURRICULUM

The Institute from time to time may revise, amend or change the regulations, scheme of examinations and syllabi if found necessary and on approval by the Academic Council and the Governing Body shall come into force and shall be binding on the students, faculty, staff, all authorities of the Institute and others concerned.

25.0 GENERAL

i. The academic regulations should be read as a whole for purpose of any interpretation.

ii. Disciplinary action for Malpractice/improper conduct in examinations is appended.

IV. Where the words "he", "him", "his", occur in the regulations, they include "she", "her", "hers". iv. In the case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Principal is final.

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

26.0 MALPRACTICES RULES

RULES FOR DISCIPLINARY ACTION FOR MALPRACTICE / IMPROPER CONDUCT IN EXAMINATIONS

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

		University.
3	Comes in a drunken condition to the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year.
4	Smuggles in the Answer book or additional sheet or takes out or arranges to send out the question paper during the examination or answer book or additional sheet, during or after the examination.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all University examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
5	Leaves the exam hall taking away answer script or intentionally tears of the script or any part thereof inside or outside the examination hall.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all University examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
6	Possess any lethal weapon or firearm in the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat.

29. M.C.A - Program Outcomes (POs)

1. Ability to apply knowledge of mathematics, computing fundamentals and specialization
2. Ability to identify, formulate and analyze complex computing problems.
3. Ability to design, solve and evaluate solution for complex computing problems.
4. Ability to conduct systematic investigations of systems and data during design & development to derive valid conclusion.
5. A sense of professional, ethical, legal, security and social issues and responsibilities.
6. Ability to use the techniques, skills, and modern tools necessary for complex computing techniques.
7. Ability to apply and commit professional ethics and cyber regulations in a global economic environment.
8. Ability to engage in independent learning for continual development with proactive measures.
9. Ability to understand financial and management principle in multidisciplinary environment
10. Ability to comprehend and write effective reports, design documentation and make effective presentation.
11. Ability to analyze the global and local impact of business solutions on individuals, organizations and the society.
12. Ability to act as a member or leader in diverse teams in multidisciplinary environments.
13. Ability to use creativity and entrepreneurial vision to create value and wealth for betterment of individual and society at large.

30. Frequently asked Questions and Answers about autonomy

1. Who grants Autonomy? UGC, Govt., AICTE or University

In case of Colleges affiliated to a university and where statutes for grant of autonomy are ready, it is the respective University that finally grants autonomy but only after concurrence from the respective state Government as well as UGC. The State Government has its own powers to grant autonomy directly to Govt. and Govt. aided Colleges.

2 Shall CREC award its own Degree?

No. Degree will be awarded by Jawaharlal Nehru Technological University, Anantapuramu with a mention of the name CREC on the Degree Certificate.

3 What is the difference between a Deemed University and an Autonomy College?

A Deemed University is fully autonomous to the extent of awarding its own Degree. A Deemed University is usually a Non-Affiliating version of a University and has similar responsibilities like any University. An Autonomous College enjoys Academic Autonomy alone. The University to which an autonomous college is affiliated will have checks on the performance of the autonomous college.

4 How will the Foreign Universities or other stake – holders know that we are an Autonomous College?

Autonomous status, once declared, shall be accepted by all the stake holders. The Govt. of Andhra Pradesh mentions autonomous status during the First Year admission procedure. Foreign Universities and Indian Industries will know our status through our website.

5 What is the change of Status for Students and Teachers if we become Autonomous?

An autonomous college carries a prestigious image. Autonomy is actually earned out of our continued past efforts on academic performances, our capability of self- governance and the kind of quality education we offer.

6 Who will check whether the academic standard is maintained / improved after Autonomy? How will it be checked?

There is a built in mechanism in the autonomous working for this purpose. An Internal Committee called Academic Programme Evaluation Committee, which will keep a watch on the academics and keep its reports and recommendations every year. In addition the highest academic council also supervises the academic matters. The standards of our question papers, the regularity of academic calendar, attendance of students, speed and transparency of result declaration and such other parameters are involved in this process.

7 Will the students of CREC as an Autonomous College qualify for University Medals and Prizes for academic excellence?

No. CREC has instituted its own awards, medals, etc. for the academic performance of the students. However for all other events like sports, cultural on co-curricular organized by the University the students shall qualify.

8 Can CREC have its own Convocation?

No. Since the University awards the Degree the Convocation will be that of the University, but there will be Graduation Day at CREC.

9 Can CREC give a provisional degree certificate?

Since the examinations are conducted by CREC and the results are also declared by CREC, the college sends a list of successful candidates with their final Grades and Grade Point Averages including CGPA to the University. Therefore with the prior permission of the University the college will be entitled to give the provisional certificate.

10 Will Academic Autonomy make a positive impact on the Placements or Employability?

Certainly, the number of students qualifying for placement interviews is expected to improve, due to rigorous and repetitive class room teaching and continuous assessment. Also the autonomous status is more responsive to the needs of the industry. As a result therefore, there will be a lot of scope for industry oriented skill development built-in into the system. The post graduates from an autonomous college will therefore represent better employability.

11 What is the proportion of Internal and External Assessment as an Autonomous College?

Presently, it is 60 % external and 40% internal. As the autonomy matures the internal assessment component shall be increased at the cost of external assessment.

12 Is it possible to have complete Internal Assessment for Theory or Practicals?

Yes indeed. We define our own system. We have the freedom to keep the proportion of external and internal assessment component to choose.

13 Why Credit based Grade System?

The credit based grade system is an accepted standard of academic performance the world over in all Universities. The acceptability of our graduates in the world market shall improve.

14 What exactly is a Credit based Grade System?

The credit based grade system defines a much better statistical way of judging the academic performance. One Lecture Hour per week of Teaching Learning process is assigned One Credit. One hour of laboratory work is assigned half credit. Letter Grades like S,A+,A, B+,B,C,F etc. are assigned for a Range of Marks. (e.g. 90% and above is S, 80 to 89 % could be A+ etc.) in Absolute Grading System while grades are awarded by statistical analysis in relative grading system. We thus dispense with sharp numerical boundaries. Secondly, the grades are associated with defined Grade Points in the scale of 1 to 10. Weighted Average of Grade Points is also defined Grade Points are weighted by Credits and averaged over total credits in a Semester. This process is repeated for all Semesters and a CGPA defines the Final Academic Performance

15 What are the norms for the number of Credits per Semester and total number of Credits for PG programme?

These norms are usually defined by UGC or AICTE. Usually around 26 Credits per semester is the accepted norm.

16 What is a Semester Grade Point Average (SGPA)?

The performance of a student in a semester is indicated by a number called SGPA. The SGPA is the weighted average of the grade points obtained in all the courses registered by the student during the semester.

$$SGPA = \sum_{i=1}^{n} (C_i G_i) / \sum_{i=1}^{n} C_i$$

Where, C_i is the number of credits of the i^{th} course and $G_i$is the grade point scored by the student in the i^{th} course and i represent the number of courses in which a student registered in the concerned semester. SGPA is rounded to two decimal places.

17 What is a Cumulative Grade Point Average (CGPA)?

An up-to-date assessment of overall performance of a student from the time of his first registration is obtained by calculating a number called CGPA, which is weighted average of the grade points obtained in all the courses registered by the students since he entered the Institute.

$$CGPA = \sum_{j=1}^{m} (C_j S_j) / \sum_{j=1}^{m} C_j$$

Where, S_j is the SGPA of the j^{th} semester and $C_j$is the total number of credits up to the semester and m represent the number of semesters completed in which a student registered up to the semester. CGPA is rounded to two decimal places.

18 Is there any Software available for calculating Grade point averages and converting the same into Grades?

Yes, the institute has its own MIS software for calculation of SGPA, CGPA, etc.

19 Will the teacher be required to do the job of calculating SGPAs etc. and convert the same into Grades?

No. The teacher has to give marks obtained out of whatever maximum marks as it is. Rest is all done by the computer.

20 Will there be any Revaluation or Re-Examination System?

No. There will double valuation of answer scripts. There will be a makeup Examination after a reasonable preparation time after the End Semester Examination for specific cases mentioned in the Rules and Regulations. In addition to this, there shall be a ‘summer term’ (compressed term) followed by the End Semester Exam, to save the precious time of students.

21 How fast Syllabi can be and should be changed?

Autonomy allows us the freedom to change the syllabi as often as we need.

22 Will the Degree be awarded on the basis of only final year performance?

No. The CGPA will reflect the average performance of all the semester taken together.

23 What are Statutory Academic Bodies?

Governing Body, Academic Council, Examination Committee and Board of Studies are the different statutory bodies. The participation of external members in everybody is compulsory. The institute has nominated professors from IIT, NIT, University (the officers of the rank of Pro-vice Chancellor, Deans and Controller of Examinations) and also the reputed industrialist and industry experts on these bodies.

24 Who takes Decisions on Academic matters?

The Governing Body of institute is the top academic body and is responsible for all the academic decisions. Many decisions are also taken at the lower level like Boards of Studies. Decisions taken at the Board of Studies level are to be ratified at the Academic Council and Governing Body.

25 What is the role of Examination committee?

The Examinations Committee is responsible for the smooth conduct of internal, End Semester and makeup Examinations. All matters involving the conduct of examinations spot valuations, tabulations preparation of Grade Cards etc, fall within the duties of the Examination Committee.

26 Is there any mechanism for Grievance Redressal?

The institute has grievance redressal committee, headed by Dean - Student affairs and Dean - IQAC.

27 How many attempts are permitted for obtaining a Degree?

All such matters are defined in Rules & Regulation

28 Who declares the result?

The result declaration process is also defined. After tabulation work wherein the SGPA, CGPA and final Grades are ready, the entire result is reviewed by the Moderation Committee. Any unusual deviations or gross level discrepancies are deliberated and removed. The entire result is discussed in the Examinations and Result Committee for its approval. The result is then declared on the institute notice boards as well put on the web site and Students Corner. It is eventually sent to the University.

29 Who will keep the Student Academic Records, University or CREC?

It is the responsibility of the Dean, Academics of the Autonomous College to keep and preserve all the records.

30 What is our relationship with the JNT University?

We remain an affiliated college of the JNT University. The University has the right to nominate its members on the academic bodies of the college.

31 Shall we require University approval if we want to start any New Courses?

Yes, It is expected that approvals or such other matters from an autonomous college will receive priority.

32 Shall we get autonomy for PG and Doctoral Programmes also?

Yes, presently our PG programmes also enjoying autonomous status.

CHADALAWADA RAMANAMMA ENGINEERING COLLEGE

(Autonomous)

MASTER OF COMPUTER APPLICATIONS

COURSE STRUCTURE

I SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per WEEK			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
17CF54101	Probability and Statistics	BS	Foundation	4		-	4	40	60	100
17CF52101	Technical Communication Skills	HS	Foundation	4	-	-	4	40	60	100
17CF53101	Accounting and Financial Management	HS	Foundation	4	-	-	4	40	60	100
17CF00101	Mathematical Foundations of Computer Science	CS	Foundation	4	-	-	4	40	60	100
17CF00102	Introduction to Problem Solving and Programming	CS	Core	4	-	-	4	40	60	100
PRACTICAL										
17CF52102	English Language Communication Skills Lab	HS	Foundation	-	-	4	2	40	60	100
17CF00103	Computer Programming Lab	CS	Core	-	-	4	2	40	60	100
17CF00104	IT Workshop	CS	Core	-	-	4	2	40	60	100
TOTAL				**20**	**-**	**12**	**26**	**320**	**480**	**800**

II SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per WEEK			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
17CF53201	Organization Structure and Human Resource Management	HS	Foundation	4	-	-	4	40	60	100
17CF00201	Data Structures	CS	Core	4	-	-	4	40	60	100
17CF00202	Computer Organization	CS	Core	4	-	-	4	40	60	100
17CF00203	Operating Systems	CS	Core	4	-	-	4	40	60	100
17CF00204	Object Oriented Programming Through C++	CS	Core	4	-	-	4	40	60	100
PRACTICAL										
17CF52201	Advanced Communication Skills Lab	HS	Foundation	-	-	4	2	40	60	100
17CF00205	Data Structures through C++ Lab	CS	Core	-	-	4	2	40	60	100
17CF00206	Python Programming Lab	CS	Core	-	-	4	2	40	60	100
TOTAL				**20**	**-**	**12**	**26**	**320**	**480**	**800**

III SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per WEEK			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
17CF00301	Database Management Systems	CS	Core	4	-	-	4	40	60	100
17CF00302	Computer Networks	CS	Core	4	-	-	4	40	60	100
17CF00303	Linux Programming	CS	Core	4	-	-	4	40	60	100
17CF00304	Software Engineering	CS	Core	4	-	-	4	40	60	100
17CF00305	Java Programming	CS	Core	4	-	-	4	40	60	100
PRACTICAL										
17CF00306	Database Management Systems Lab	CS	Core	-	-	4	2	40	60	100
17CF00307	Linux Programming Lab	CS	Core	-	-	4	2	40	60	100
17CF00308	Java Programming Lab	CS	Core	-	-	4	2	40	60	100
TOTAL				**20**	**-**	**12**	**26**	**320**	**480**	**800**

IV SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per WEEK			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
17CF00401	Object Oriented Analysis and Design	CS	Core	4	-	-	4	40	60	100
17CF00402	Web Technologies	CS	Core	4	-	-	4	40	60	100
17CF00403	Data Warehousing and Mining	CS	Core	4	-	-	4	40	60	100
17CF00404 17CF00405 17CF00406	**Elective – I** a. .NET Technologies b. Information Security c. Computer Graphics	CS	Elective	4	-	-	4	40	60	100
17CF00407 17CF00408 17CF00409	**Elective – II** a. Artificial Intelligence b. Distributed Systems c. Cloud Computing	CS	Elective	4	-	-	4	40	60	100
PRACTICAL										
17CF00410	Object Oriented Analysis and Design Lab	CS	Core	-	-	4	2	40	60	100
17CF00411	Web Technologies Lab	CS	Core	-	-	4	2	40	60	100
17CF00412	Data Warehousing and Mining Lab	CS	Core	-	-	4	2	40	60	100
TOTAL				**20**	**-**	**12**	**26**	**320**	**480**	**800**

V SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per WEEK			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
17CF00501	Fundamentals of Data Science	CS	Core	4	-	-	4	40	60	100
17CF00502	Mobile Application Development	CS	Core	4	-	-	4	40	60	100
17CF00503	Scripting Languages	CS	Core	4	-	-	4	40	60	100
 17CF00504 17CF00505 17CF00506	**Elective – III** a. Software Project Management b. Machine learning c. Internet of Things	CS	Elective	4	-	-	4	40	60	100
 17CF00507 17CF00508 17CF00509	**Elective – IV** a. Web Services b. Cyber Security c. Software Testing	CS	Elective	4	-	-	4	40	60	100
PRACTICAL										
17CF00510	R and Analytics Lab	CS	Core	-	-	4	2	40	60	100
17CF00511	Mobile Application Development Lab	CS	Core	-	-	4	2	40	60	100
17CF00512	Scripting Languages Lab	CS	Core	-	-	4	2	40	60	100
TOTAL				**20**	**-**	**12**	**26**	**320**	**480**	**800**

Course Code	Course Name	Subject Area	Category	Periods per WEEK			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
17CF00501	Fundamentals of Data Science	CS	Core	4	-	-	4	40	60	100
17CF00502	Mobile Application Development	CS	Core	4	-	-	4	40	60	100
17CF00503	Scripting Languages	CS	Core	4	-	-	4	40	60	100
 17CF00504 17CF00505 17CF00506	**Elective – III** a. Software Project Management b. Human Computer Interaction c. Internet of Things	CS	Elective	4	-	-	4	40	60	100
 17CF00507 17CF00508 17CF00509	**Elective – IV** a. Web Services b. Design Patterns c. Software Testing	CS	Elective	4	-	-	4	40	60	100
PRACTICAL										
17CF00510	R and Analytics Lab	CS	Core	-	-	4	2	40	60	100
17CF00511	Mobile Application Development Lab	CS	Core	-	-	4	2	40	60	100
17CF00512	Scripting Languages Lab	CS	Core	-	-	4	2	40	60	100
TOTAL				**20**	**-**	**12**	**26**	**320**	**480**	**800**

VI SEMESTER (Non-FSI Model)

Course Code	Course Name	Subject Area	Category	Periods per WEEK	Credits	Scheme of Examination Max. Marks

				L	**T**	**P**		**CIA**	**SEE**	**Total**
PRACTICAL										
17CF00601	Seminar	CS	Skill	-	3	-	2	40	60	100
17CF00602	Main Project	CS	Core	-	-	-	8	-	-	-
TOTAL				**-**	**-**	**-**	**10**	**40**	**60**	**100**

SYLLABUS
(Semesters: I - V)

PROBABILITY & STATISTICS

I – Semester

Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
17CF54101	Foundation	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		4	-	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		

Course Objectives:

- To help the students in getting a thorough understanding of the fundamentals of probability and usage of statistical techniques like testing of hypothesis, curve fitting and correlation, regressions.

Course Outcomes:

- The student will be able to analyze the problems of computer & industry using the techniques of testing of hypothesis, curve fitting.

Unit-I		**Classes: 08**

Basic Concepts of Probability - Conditional probability – Baye's theorem. Random variables – Expectation Discrete and continuous – Distribution – Distribution functions. Binomial and Poison distributions Normal distribution – Related properties.

Unit-II		**Classes: 10**

Test of hypothesis: Populations and samples- confidence interval of mean from normal distribution – Statistical hypothesis – Null and Alternative Hypothesis –Level of Significance –Test of significance – Test based on the normal distribution –Z-test for means and proportions: small simples –t-test for one sample and two sample problem and paired t-test, F-test and chi-square test (testing of goodness of fit and independence).

Unit-III		**Classes: 08**

Analysis of variance one way classification and two-way classification. Latin square Design and RBD.

Unit-IV		**Classes: 10**

Statistical Quality control : Concept of quality of a manufactured Defectives- Causes of variations-Random and assignable – the principle of Schwartz control charts for attribute and Variable quality characteristics- Constructions and operation of X-bar chart, R-chart , P-chart and C-chart.

Unit-V		**Classes: 09**

Curve fitting: The method of least squares- Inferences based on the least squares estimations-curvilinear regression-Multiple regression-correlation for univariate and bivariate distributions.

Text Books:

1. Probability & Statistics for engineers by Dr.J.Ravichandran WILEY-INDIA publishers.
2. Probability &statistics by E.Rukmagadachari &E.keshava Reddy, Pearson publisher.

Reference Books:

1. Probability & Statistics by T.K.V.Iyengar, B.Krishna Gandhi and S.Ranganatham and M.V.S.S.N.Prasad, S.Chand publications.
2. Mathematical Statistics by B.Rama Bhupal Reddy Research India Publications (DELHI), 2016).
3. Stastical methods byS.P.Gupta,S.Chand Publications.
4. Probability & Statistics for Science and Engineering by G.Shankarrao, Universities Press.
5. Probability & Statistics for Engineering and Sciences by Jay L.Devore, Cengage.
6. Probability & Statistics by R.A.Johnson and Gupta C.B.

Web References:

1. https://ox.libguides.com/statistics-resources
2. https://stattrek.com/m/
3. https://www.oakdaleengr.com

E-Text Books:

1. https://www.e-booksdirectory.com/details.php?ebook=10166
2. https://www.e-booksdirectory.com/details.php?ebook=7400re

TECHNICAL COMMUNICATION SKILLS

I - Semester

Course Code	Category	Hours / Week			Credits	Maximum Marks		
17CF52101	Foundation	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		4	-	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil			Total Classes: 60			

Course Objectives:

- To develop awareness in students of the relevance and importance of technical communication and presentation skills.
- To prepare the students for placements
- To sensitize the students to the appropriate use of non-verbal communication
- To train students to use language appropriately for presentations and interviews
- To enhance the documentation skills of the students with emphasis on formal and informal writing

COURSE OUTCOMES

- Become effective technical communicators
- Be job-ready and able to face interviews confidently
- Sensitive use of non-verbal language suitable to different situations in professional life
- Learn and use keys words, phrases and sentence structures making a mark in interviews and presentation skills
- Effective writing skills with the ability to use different styles for different situations

Unit-I		**Classes: 08**

Basics of Technical Communication – Introduction – Objectives & Characteristics of Technical Communication – Importance and need for Technical communication - LSRW Skills – Barriers to effective communication

Unit-II		**Classes: 10**

Informal and Formal Conversation - Verbal and Non-verbal communication –Kinesics, Proxemics, Chronemics, Haptics, Paralanguage

Unit-III		**Classes: 08**

Written communication – Differences between spoken and written communication – Features of effective writing –Advantages and disadvantages of spoken and written communication- Art of condensation- summarizing and paraphrasing

Unit-IV		**Classes: 10**

Presentation Skills – Nature and importance of oral presentation – Defining the purpose – Analyzing the audience - Planning and preparing the presentation, organizing and rehearsing the presentation – Individual and group presentations - Handling stage fright

Unit-V		**Classes: 09**

Interview Skills – The Interview process –Characteristics of the job interview – Pre-interview preparation techniques – Projecting the positive image – Answering Strategies

Text Books:

1. Effective Technical Communication, Ashrif Rizvi, TataMcGrahill, 2011
2. Technical Communication by Meenakshi Raman & Sangeeta Sharma,3rd Edition, O U Press 2015

Reference Books:

1. Communication Skills by Pushpa latha & Sanjay Kumar, Oxford University Press
2. Books on TOEFL/GRE/GMAT/CAT/ IELTS by Barron’s/DELTA/Cambridge University Press.2012.
3. Soft Skills for Everyone, Butterfield Jeff, Cengage Publications, 2011.
4. Management Shapers Series by Universities Press (India) Pvt Ltd., Himayath nagar, Hyderabad 2008.
5. Successful Presentations by John Hughes & Andrew Mallett, Oxford.
6. Winning at Interviews by Edgar Thorpe and Showick Thorpe, Pearson

Web References:

1. https://www.vocabulary.com/lists/291470
2. https://amarit04.wordpress.com/2008/02/17/5-types-of-kinesics/
3. https://www.monster.com/career-advice/article/boost-your-interview-iq
4. https://www.skillsyouneed.com/writing-skills.html
5. https://www.hanselman.com/blog/11TopTipsForASuccessfulTechnicalPresentation.aspx

E-Text Books:

1. eBook *of* Word Power Made Easy
2. Effective Communications for the Technical Professions. *by Jennifer MacLennan.* eBook

Course Home Page:

1. https://docs.moodle.org/32/en/Course_homepage
2. https://community.canvaslms.com/docs/DOC-10557

ACCOUNTING AND FINANCIAL MANAGEMENT

I – Semester

Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF53101	Foundation	-	4	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		

Course Objectives:

- The objective of the course is to familiarize the student with the fundamentals of Accounting principles and Financial Management for making sound financial decisions.

Course Outcomes:

- After completion of this course, the student will be able to understand the basic accounting principles, gets exposure to the fundamental concepts, techniques and tools of Financial Management, also enables to prepare and analyze financial statements of business enterprises for taking sound financial decisions.

Unit-I **Classes: 08**

Introduction to Accounting :Definition of Accounting- Accounting concepts– Principles Double entry system of accounting classification of accounts - Books of accounts – Journal entries Ledger books – preparation of financial statements and accounts-Trial Balance- Trading account-Profit and Loss account - Balance sheet(Simple problems with adjustments)

Unit-II **Classes: 10**

Cost Accounting and Marginal Costing: Nature- importance- Scope- difference between financial accounting and cost accounting- principles-Absorption costing- Marginal Costing - Concept of Break Even Analysis - Margin of Safety and P/V ratio- Break Even Point-Determination of BEP- Cost Volume-Profit Analysis – managerial applications of BEP and application of marginal costing techniques (Simple problems).

Unit-III **Classes: 08**

Financial Analysis and Interpretations: Funds flow and cash flow statements meaning importance-statement of changes in working capital - sources and application of funds - Funds Flow and Cash flow analysis-Financial analysis through Ratios–liquidity ratios- solvency ratios – Profitability ratio, Activity ratio (Simple problems).

Unit-IV **Classes: 10**

Financial Management: Definition-Course Objectives- finance functions-importance-Profit and wealth maximization- Sources of capital- concept of Leverage and types of Leverage- Over Capitalization and Under Capitalization- Time Value of money -Present value of Money and Future Value of Money.

Unit-V **Classes: 09**

Capital Budgeting and Budgeting Techniques: Definition- Features- Significance methods of evaluation of capital budgeting proposals - Payback Period-Accounting Rate of Return (ARR)- Net Present Value Method (NPV) and Internal Rate of Return (IRR)- (Simple problems)

Text Books:

1. M.N.Arora, Accounting for Management, , HPH, 2012.
2. T.S.Reddy and Y.Hari Prasad Reddy, Accounting and Financial Management, Margham Publications.

Reference Books:

1. Khan M.Y, Jain P.K, Management Accouting, 5th Edition , Tata McGraw Hill, 2012.
2. S.N.Maheshwari, Financial Accounting, 4th Edition, Vikas Publications, 2012.
3. Khan M.Y, Jain P.K, Financial Statement Analysis, PHI, 2009.
4. I.M.Pandey, Financial Management,10th Edition, Vikas Publications, 2011.
5. Financial Management, 7th Edition, TMH, 2011.

Web References:

http://www.icsi.edu/Webmodules/Publications/Company%20Accounts,%20Cost%20and%20Management%20Accounting.pdf

E-Text Books:

https://sjctnc.edu.in/wp-content/uploads/2015/08/EPCA915S1.pdf

MATHEMATICAL FOUNDATION OF COMPUTER SCIENCE

I – Semester

Course Code	Category	Hours / Week			Credits	Maximum Marks		
17CF00101	Core	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		

Course Objectives:

- Apply logical reasoning to solve a variety of problems.
- Understand and apply methods of discrete mathematics such as proofs, counting principles, number theory, logic and set theory to mathematical problems in a creative way.
- To apply the abstract concepts of graph theory in modeling and solving non-trivial problems in different fields of study.

Course Outcomes:

- Able to apply mathematical concepts and logical reasoning to solve problems in different fields of Computer science and information technology.
- Able to apply the concepts in courses like Computer Organization, DBMS, Analysis of Algorithms, Theoretical Computer Science, Cryptography, Artificial Intelligence

Unit-I		**Classes: 10**

Sets and Propositions: Introduction, Combination of Sets, Finite and Infinite Sets, Uncountable Infinite Sets, Mathematical Induction, Principle of Inclusion and Exclusion, Multisets, Propositions, Logical Connectives, Conditional and Bi-conditionals, Well-Formed Formulas, Tautologies, Logical Equivalences. Relations and Functions: Introduction, Properties of Binary Relations, Closure of Relations.

Unit-II		**Classes: 10**

Groups: Introduction, Groups, Subgroups, Generators and Evaluations of Powers, Co-sets and Lagrange's Theorem, Permutations Groups and Burnside's Theorem, Codes and Group Codes, Isomorphism's and Automorphisms, Homomorphism's and Normal Subgroups.

Unit-III		**Classes: 10**

Permutations, Combinations, and Discrete Probability: Introduction, the Rules of Sum and Product, Permutations, Combinations, Generation of Permutations and Combinations, Discrete Probability, Conditional Probability.

Recurrence Relations and Recursive Algorithms: Introduction, Recurrence Relations, Linear Recurrence Relations with Constant Coefficients, Homogeneous Solutions, Particular Solutions, Total Solutions

Unit-IV		**Classes: 08**

Graphs: Introduction, Basic Terminology, Multigraphs and Weighted Graphs, Digraphs and Relations, Representation of Graphs, Operations on Graphs, Paths and Circuits, Graph Traversals, Shortest Paths in Weighted Graphs, Eulerian Paths and Circuits, Hamiltonian Paths and Circuits

Unit-V		**Classes: 07**

Trees: Trees, Rooted Trees, Path Lengths in Rooted Trees, Prefix Codes, Binary Search Trees, Spanning Trees and Cut sets, Minimum Spanning Trees, Kruskal's Algorithm, Prim's Algorithm.

Discrete Numeric Functions: Introduction, Manipulation of Numeric Functions, Asymptotic Behavior of Numeric Functions.

Text Books:

1. C L Liu and D Mohapatra “Elements of Discrete Mathematics”, Tata Mcgraw Hill, 2009.
2. J. P. Tremblay, R. Manohar, “Discrete Mathematical Structures with Applications to Computer Science”, Tata Mc Graw Hill, India, 1st Edition, 1997.
3. Joe L. Mott, Abraham Kandel, Theodore P. Baker, “Discrete Mathematics for Computer Scientists and Mathematicians”, Prentice Hall of India Learning Private Limited, New Delhi, India, 2nd Edition, 2010.

Reference Books:

1. Discrete and Combinatorial Mathematics, Fifth Edition, R. P. Grimaldi, B.V. Ramana, Pearson
2. Discrete Mathematics Theory and Applications, D.S Malik and M.K. Sen, Cengage Learning
3. J .L.Mott, A.Kandel, T.P .Baker, Discrete Mathematics for Computer Scientists and Mathematicians, second edition 1986, Prentice Hall of India
4. C.L.Liu, Elements of Discrete Mathematics, Second Edition 1985, McGraw-Hill Book Company. Reprinted 2000
5. Discrete Mathematics, Norman L. Biggs, Second Edition, OXFORD Indian Edition.

Web References:

1. http://www.web.stanford.edu/class/cs103x
2. http://www.cs.odu.edu/~cs381/cs381content/web_course.html
3. http://www.cse.iitd.ernet.in/~bagchi/courses/discrete-book
4. http://www.saylor.org/course/cs202/
5. http://www.nptel.ac.in/courses/106106094/
6. http://www.tutorialspoint.com/discrete_mathematics
7. http://www.dmtcs.org/dmtcs-ojs/index.php/dmtcs

E-Text Books:

1. https://people.eecs.berkeley.edu/~daw/teaching/cs70-s05/
2. http://home.anadolu.edu.tr/~eakyar/dersler/ayrik/kitap/kitap.pdf
3. http://45.63.83.30/graph-theory-keijo-ruohonen-pdf-tut.pdf
4. http://www.zib.de/groetschel/teaching/WS1314/BondyMurtyGTWA.pdf

INTRODUCTION TO PROBLEM SOLVING AND PROGRAMMING

I- Semester

Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF00102	Core	3	1	-	4	40	60	100
Contact Classes:45	Tutorial Classes:15	Practical Classes: Nil				Total Classes:60		

Course Objectives:

- To understand the various steps in Program development.
- To understand the basic concepts in C Programming Language.
- To learn how to write modular and readable C Programs
- To understand the basic concepts such as Abstract Data Types, Linear and Non Linear Data structures.
- To understand the notations used to analyze the Performance of algorithms.
- To understand and analyze various searching and sorting algorithms.

Course Outcomes:

- Able to design the flowchart and algorithm for real world problems
- Able to learn and understand new programming languages
- Able to construct modular and readable programs
- Able to write C programs for real world problems using simple and compound data types
- Adapt programming experience and language knowledge to other programming language contexts
- Employee good programming style, standards and practices during program development

Unit – I **Classes:10**

Introduction of Computer Programing: Introduction – The Problem-solving Aspect – Top-down Design – Implementation of Algorithms –flowcharts-programming development steps-programming languages. **Fundamental Algorithms** – Introduction – Exchanging the Values of Two Variables – Counting – Summation of a Set of Numbers – Factorial Computation – Sine Functional Computation – Generation of the Fibonacci Sequence – Reversing the Digits of an Integer.

Unit – II **Classes:08**

Overview of C Language: Features – Components – Structure – Process of Executing a ‘ C’ Program - Data Types – Variables – Constants – Operators - Type Modifiers – Expressions – Type Definitions using typedef.

Unit –III **Classes:08**

Control Statements, Iterations and Arrays: Control Statements – Conditional Statements – Loops – Infinite Loops – Nested Loops – Break Statement – Continue Statement – exit() Function – goto Statement – Introduction to Arrays – One-dimensional Array – Strings – Two-dimensional Array

Unit – IV **Classes:08**

Functions - Introduction to Functions – Function Declaration and Prototypes – Definition – Storage Classes – Scope and Lifetime of Declaration – Passing Parameters of Functions – Command Line Arguments – Recursion in Function.

Structures – Definition – Bit Fields – Giving Values to Members – Structure Initialization – Comparison of Structures Variables – Arrays of Structures – Array within Structures – Structures within Structures – Passing Structures to Functions – Structure Pointers.

Unions – Definition and Declaration – Accessing a Union Member – Union of Structures –

Initialization of a Union Variable – Use of Union – Use of User-defined Type Declarations.

Unit – V		**Classes:08**

Pointers – Introduction to Pointers – Pointer Notation – Declaration and Initialization – Accessing a Variable through a Pointer – Difference between Array and Pointer – Pointer Expressions – Pointers and One-dimensional Arrays – malloc Library Function – calloc Library Function – Pointers and Multi-dimensional Arrays – Arrays of Pointers – Pointer to Pointers – Pointers and Functions – Functions with a Variable Number of Arguments.

File Handling In 'C' – File – Defining and Opening a File – Closing a File – Input / Output Operations on Files – Functions for Random Access to Files – Example Programs.

Text Books:

1. R. G. Dromey, How to Solve it by Computer, Pearson Education, 2007.
2. ISRD Group, Programming and Problem Solving Using C, Tata McGraw-Hill.

References:

1. Herbert Schildt, Osborne, C- The Complete Reference, Mcgraw Hill, Inc.
2. Brian W. Kerningham and Dennis Ritchie, C Programming Language (ANSI C), Pearson Edition.
3. B.S. Gottfried, Programming with C, Schaum Series, TMH.
4. Alfred V. Aho, Foundations of Computer Science(C Edition).

Web References

1. https://www.tutorialspoint.com/cprogramming/
2. www.studytonight.com/c/
3. fresh2refresh.com/c-programming/
4. www.cprogramming.com/tutorial/c/

E-Text Books:

1. bookboon.com/en/c-cpp-csharp-ebooks
2. electronicsforu.com › Resources › Cool Stuff
3. https://en.wikibooks.org/wiki/C_Programming
4. www.e-booksdirectory.com › Computers & Internet

ENGLISH LANGUAGE AND COMMUNICATION SKILLS LAB

I - Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF52102	Foundation	-	-	3	2	40	60	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 42				Total Classes: 42		

Course Objectives:

- To develop awareness in students of the relevance and importance of technical communication and presentation skills.
- To prepare the students for placements
- To sensitize the students to the appropriate use of non-verbal communication
- To train students to use language appropriately for presentations and interviews
- To enhance the documentation skills of the students with emphasis on formal and informal writing

Course Outcomes:

- Become effective technical communicators
- Be job-ready and able to face interviews confidently
- Sensitive use of non-verbal language suitable to different situations in professional life
- Learn and use keys words, phrases and sentence structures making a mark in interviews and presentation skills

LIST OF EXPERIMENTS

Exp.1

Phonetics- introduction to sounds of English – vowels – diphthongs – consonants – phonetic transcription & Orthographic Transcription

Exp.2

Syllabification – Word Stress – Rules of word stress – Intonation – Falling tone and Rising tone

Exp.3

Situational Dialogues – Role-play – Expressions in various situations – Self Introduction – Introducing others – Greetings – Apologies – Requests – Giving directions -Social and Professional etiquettes – Telephone Etiquettes

Exp.4

JAM – Describing Pictures, Photographs, Products, and Process – Talking about Wishes-Information Transfer

Exp.5

Debates - Group Discussions-1

MINIMUM REQUIREMENT FOR ELCS LAB:

The English Language Lab shall have two parts:

1. Computer Assisted Language Learning (CALL) Lab:
   The Computer aided Language Lab for 60 students with 60 systems, one master console, LAN facility and English language software for self- study by learners.

2. The Communication Skills Lab with movable chairs and audio-visual aids with a P.A. system, Projector, a digital stereo-audio & video system and camcorder etc.

System Requirement (Hardware component):
Computer network with LAN with minimum 60 multimedia systems with the following specifications:

i) P – IV Processor
   a) Speed – 2.8 GHZ
   b) RAM – 512 MB Minimum
   c) Hard Disk – 80 GB

ii) Headphones of High quality

Reference Books:

1. A Textbook of English Phonetics for Indian Students 2nd Ed T. Balasubramanian. (Macmillian), 2012.
2. A Course in Phonetics and Spoken English, Dhamija Sethi, Prentice-Hall of India Pvt.Ltd
   Speaking English Effectively, 2nd Edition Krishna Mohan
4. A Hand book for English Laboratories, E.Suresh kumar, P.Sreehari, Foundation Books,2011
5. English Pronunciation in Use. Intermediate & Advanced, Hancock, M. 2009. CUP
6. Basics of Communication in English, Soundararaj, Francis. 2012.. *New Delhi: Macmillan*
7. Spoken English (CIEFL) in 3 volumes with 6 cassettes, OUP.
   English Pronouncing Dictionary, Daniel Jones Current Edition with CD.Cambridge,7th
8. edition 2011

Web Reference:

1. https://www.vocabulary.com/lists/291470
2. https://www.skillsyouneed.com/writing-skills.html

I.T. WORKSHOP

I - Semester								
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF00103	Core	-	-	3	2	40	60	100
Contact Classes: Nil	Tutorial Classes: Nil			Practical Classes: 45		Total Classes: 45		

Course Objectives:

- Learning about the Computer internal components
- Practice on operating system installation and configuration settings.
- Prepare productivity tools like word processors, spreadsheets, presentations.

Course Outcomes:

- Able to Assemble and dissemble the computer components.
- Able to prepare power point presentations
- Able to construct data charts and graphs.
- Able to write documentation.

Exp-1	Learn about computer internal parts & Peripherals.
Exp-2	Assembling & Disassembling a Computer.
Exp-3	Installation of various Operating Systems.
Exp-4	Networking two or more computers and document the process.
Exp-5	Browsing Internet and creating an email account: Studying various web browsers and their features.
Exp-6	**Word Processor**: Introduction to Word: Importance of word as word processor, overview of toolbars, saving, accessing file, using help and resources; Creating project Certificate; Abstract features to be covered; Formatting Styles: Inserting table, bullets and numbering, changing text direction, cell alignment, footnote, hyperlink, symbols, spell check, images from files and clipart, drawing toolbar and Word Art, formatting images, textboxes and paragraphs.
Exp- 7	**Spreadsheet-I**: Spreadsheet basics, modifying worksheets, formatting cells, formulas and functions.
Exp-8	**Spreadsheet-II**: Sorting and filtering, charts, renaming and inserting worksheets, hyper linking, count function, sorting, and conditional formatting.
Exp- 9	**Presentations**: creating, opening, saving and running the presentations, Selecting the style for slides, formatting the slides with different fonts, colors, creating charts and tables, inserting and deleting text, graphics and animations, bulleting and numbering, hyperlinking, running the slide show, setting the timing for slide show.

Optional Tasks:

Exp- 10	A report on specifications of Laboratory Equipment
Exp-11	A report on different Antivirus softwares and their installation, usage.

References:

1. Introduction to Computers, Peter Norton, Mc Graw Hill
2. “MOS study guide for word, Excel, Power point & Outlook Exams”, Joan Lambert, Joyce Cox, PHI.

COMPUTER PROGRAMMING LAB

I – Semester								
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF00104	Core	-	-	3	2	40	60	100
Contact Classes: Nil	Tutorial Classes: Nil		Practical Classes: 45			Total Classes:45		

Course Objectives:

- To work with the compound data types
- To explore dynamic memory allocation concepts
- Able to design the flowchart and algorithm for real world problems
- Able to write C programs for real world problems using simple and compound data types
- Employee good programming style, standards and practices during program development

Course Outcomes

- Able to have fundamental concept.
- Able to write, compile and debug programs in C language.
- Able to formulate problems and implement algorithms in C.
- Able to effectively choose programming components that efficiently solve computing problems in real-world.
- Able to use different data types in a computer program.
- Able to design programs involving decision structures, loops and functions.

List of **Programs**

Exp-1

1) Write a C program to make the following exchange between the variables a-> b->c->d- > a

2) Write a C program to carry out the arithmetic operations addition, subtraction, multiplication, and division between two variables

3) Write a C program for printing prime numbers between 1 and n.

Exp-2

1) Write a C program to construct a multiplication table for a given number.

2) Write a program to reverse the digit of a given integer.

3) Write a C program to find the sum of individual digits of a positive integer.

4) Write a C program to calculate the factorial of a given number

Exp-3

1. Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and

1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence

2) Write a program to calculate tax, given the following conditions:

a) If income is less than 1,50,000 then no tax.

b) If taxable income is in the range 1,50,001 – 300,000 then charge 10% tax

c) If taxable income is in the range 3,00,001 – 500,000 then charge 20% tax

d) If taxable income is above 5,00,001 then charge 30% tax

Exp-4

1. Write a program to print the calendar for a month given the first Week - day of the month.

Input the first day of the month (Sun=0,Mon=1,Tue=2,Wed=3,.....) :: 3

Total number of days in the month : 31

Expected output

Sun	Mon	Tue	Wed	Thu	Fri	Sat
-	-	-	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	-

2) Write a C program to find the roots of a quadratic equation

Exp-5

1) Write a program to print the Pascal triangle for a given number

2) Write a C program to find the GCD (greatest common divisor) of two given integers

3) Write a C program to construct a pyramid of numbers.

4) Write C code to define a function cash_dispense, which takes an amount as its input, and returns the number of 1000, 500, 100, 50, 20, 10, 5, 2, 1 rupee denomination that make up the given amount.

Exp-6

1).Write C code to reverse the contents of the array. For example, [1,2,3,4,5] should become

[5,4,3,2,1]

2) Write a C program that uses functions to perform the following:

i) Addition of Two Matrices

ii) Multiplication of Two Matrices

3) Write a program that will search and find out the position where the given key element exist in a user chosen array and print it as output

Exp-7

1) Write C code to compute the frequency table of survey responses given by 20 users.

The survey responses range from 1 to 5 and are stored in an array. For example, 10 responses are stored in the array [1,1,5,2,3,3,5,5,2,2]. The frequency table will be as shown below:

a. 1 = 2

b. 2 = 3

c. 3 = 2

d. 4 = 0

e. 5 = 3

2) Write a program to define a function to sort an array of integers in ascending order by using exchange sort.

Exp-8

1) Write a C program to check whether a given string is a palindrome or not, without using any built-in functions.

2) Write a C program to determine if the given string is a palindrome or not by using string functions.

3) Write a function that accepts a string and delete the first character.

4) Write a function that accepts a string and delete all the leading spaces

Exp-9

Write a program to accept a string from user and display number of vowels, consonants, digits and special characters present in each of the words of the given string.

Exp-10

1) Write a C program to define a union and structure both having exactly the same numbers using the sizeof operators print the sizeof structure variables as well as union variable.

2) Declare a structure time that has three fields hr, min, secs. Create two variables, start_time and end_time. Input there values from the user. Then while start_time is not equal to end _time display GOOD DAY on screen

Exp-11

1) Write a program to read in an array of names and to sort them in alphabetical order. Use sort function that receives pointers to the functions strcmp, and swap, sort in turn should call these

functions via the pointers.

2) Write a program to read and display values of an integer array. Allocate space dynamically for the array using the malloc().

3)Write a program to calculate area of a triangle using function that has the input parameters as pointers as sides of the triangle.

Exp-12

1) Two text files are given with the names text1 and text2. These files have several lines of text. Write a program to merge (first line of text1 followed by first line of text2 and so on until both the files reach the end of the file) the lines of text1 and text2 and write the merged text to a new file text3.

2) Write a program to split a given text file into n parts. Name each part as the name of the original file followed by .part<n> where n is the sequence number of the part file.

Reference Books

1. Computer Science, A Structured Programming Approach Using C by Behrouz A. Forouzan& Richard F. Gilberg, Third Edition, Cengage Learning
2. C Programming A Problem-Solving Approach, Behrouz A. Forouzan& E.V. Prasad, F. Gilberg, Third Edition, Cenga ge Learning
3. Programming with C RemaTheraja, Oxford
4. “C Test Your Skills”, Kamthane, Pearson Education
5. Programming in C: A Practical Approach, Ajay Mittal, Pearson
6. Problem solving with C, M.T.Somasekhara, PHI
7. C Programming with problem solving, J.A. Jones & K. Harrow,Dreamtech Press
8. Programming withc, Byron S Gottfried, Jitender Kumar Chhabra, TMH, 2011

Web References

1. https://www.programiz.com/
2. https://www.programmingsimplified.com
3. https://www.techcrashcourse.com
4. https://www.sanfoundary.com/

DATABASE MANAGEMENT SYSTEMS

III – Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF00301	Core	3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		

Objectives:

- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- To understand the relational database design principles.
- To become familiar with the basic issues of transaction processing and concurrency control.
- To become familiar with database storage structures and access techniques.

Course Outcomes:

- Demonstrate the basic elements of a relational database management system,
- Ability to identify the data models for relevant problems.
- Ability to design entity relationship and convert entity relationship diagrams into RDBMS and formulate SQL queries on the respect data.
- Apply normalization for the development of application software

Unit-I		**Classes: 10**

Introduction to file and database systems: Database system structure, data models, introduction to network and hierarchical models, ER model.

Relational Model: Introduction to the Relational Model - Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design, Introduction to Views Destroying/ altering Tables and Views.

Unit-II		**Classes: 10**

Relational algebra and calculus: Relational algebra, selection and projection, set operations, renaming, joins, division, examples of algebra queries, relational calculus, Tuple relational calculus, domain relational calculus, expressive power of algebra and calculus.

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

Unit-III **Classes: 10**

Queries in SQL: SQL data definition and updates, views, integrity and security, relational database design. Functional dependencies and normalization for relational databases up to five normal forms.

Database System Architecture: Database System Architectures, Distributed Databases, Parallel Databases.

Unit-IV **Classes: 08**

Transaction processing: Introduction, need for concurrency control, desirable properties of transaction, schedule and recoverability, Serializability and schedules.

Concurrency control: Types of locks, two phases of locking, deadlock, time stamp based concurrency control, recovery techniques, concepts, immediate update, deferred update, shadow paging.

Unit-V **Classes: 07**

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

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

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

Text Books:

1. Abraham Silberschatz, Abraham. Database system concepts. 5th edition. Boston : McGraw-Hill Higher Education, ©2006.
2. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, McGrawHill

   Education, 3rd Edition, 2003.

Reference Books:

1. Ramez Elmasri, Shamkant B. Navathe, "Fundamental Database Systems", Pearson Education, 4rdEdition, 2003.
2. Raghu Rama krishnan, "Database Management System", Tata McGraw-Hill Publishing Company, 3rd Edition, 2003.
3. Hector Garcia Molina, Jeffrey D. Ullman, Jennifer Widom, "Database System Implementation", Pearson Education, United States, 1st Edition, 2000.
4. Peter Rob, Corlos Coronel, "Data base System, Design, Implementation and Management", Thompson Learning Course Technology, 5th Edition, 2003.

Web References:

1. https://www.youtube.com/results?search_query=DBMS+onluine+classes
2. http://www.w3schools.in/dbms/
3. http://beginnersbook.com/2015/04/dbms-tutorial/

E-Text Books:

1. http://www.e-booksdirectory.com/details.php?ebook=10166
2. http://www.e-booksdirectory.com/details.php?ebook=7400re

COMPUTER NETWORKS

III – Semester

Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
17CF00302	Core	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		

Course Objectives:

- Study the evolution of computer networks and future directions.
- Study the concepts of computer networks from layered perspective.
- Study the issues open for research in computer networks.

Course Outcomes:

- Ability to choose the transmission media depending on the requirements.
- Ability to design new protocols for computer network.
- Ability to configure a computer network logically.

Unit-I		**Classes: 10**

Introduction: Networks, Network Types, Internet History, Standards and Administration, Network Models: Protocol Layering, TCP/IP Protocol Suite, The ISO Model.
Transmission media: Introduction, Guided Media, Unguided Media, Switching: Introduction, Circuit
Switched Networks, Packet switching.

Unit-II		**Classes: 12**

The Data Link Layer: Introduction, Link layer addressing, Error detection and Correction: Cyclic codes, Checksum, Forward error correction, Data link control: DLC Services, Data link layer protocols,
HDLC, Point to Point Protocol, Media Access control: Random Access, Controlled Access, Channelization, Connecting devices and virtual LANs: Connecting Devices.

Unit-III		**Classes: 10**

Network Layer: design issues, routing algorithms, congestion control algorithms, Internetworking: Tunneling, internetworking, fragmentation, network layer in the internet, IP protocols, IP addresses, subnets, internet control protocols, OSPF (Open Shortest Path First), BGP (Border Gateway Protocol), IP (Internet Protocol), ICMP (Internet Control Message Protocol). IGMP(Internet Group Message Protocol)

Unit-IV		**Classes: 08**

Transport Layer: Elements of transport protocols-addressing, establishing a connection, releasing connection, flow control & buffering & crash recovery, End to End Protocol UDP-reliable byte streams (TCP)-end to end format, segment format connection establishment & termination, sliding window revisited, adaptive retransmission.

Unit-V		**Classes: 07**

Application Layer : Network Security - Cryptographic Algorithms, DES, RSA, security mechanisms, Authentication protocols, Message Integrity protocol, Firewalls, Name service (DNS) Domain Hierarchy, Name servers, Name resolutions, SMTP, MIME, World Wide Web-HTTP, SNMP.

Text Books:

1. “Data communications and networking”, Behrouz A. Forouzan, Mc Graw Hill Education, 5th edition, 2012.
2. “Computer Networks”, Andrew S. Tanenbaum, Wetherall, Pearson, 5th edition, 2010.

Reference Books:

1. Data Communication and Networks, Bhushan Trivedi, Oxford
2. “Internetworking with TCP/IP – Principles, protocols, and architecture- Volume 1, Douglas E.
Comer, 5th edition, PHI
3. “Computer Networks”, 5E, Peterson, Davie, Elsevier.
4. “Introduction to Computer Networks and Cyber Security”, Chawan- Hwa Wu, Irwin, CRC Publications.
“Computer Networks and Internets with Internet Applications”, Comer.

Web References:

https://www.vssut.ac.in/lecture_notes/lecture1428550521.pdf

E-Text Books:

http://www.uoitc.edu.iq/images/documents/informatics-institute/exam_materials/Computer%20Networks%20-%20A%20Tanenbaum%20-%205th%20edition.pdf

LINUX PROGRAMMING

III – Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
17CF00303	Core	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		-	4	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		

Course Objectives:

- To understand and make effective use of Linux utilities and Shell scripting language (bash) to solve Problems.
- To implement in C some standard Linux utilities such as ls, mv, cp etc. using system calls.
- To develop the skills necessary for systems programming including file system programming, process and signal management, and interprocess communication.
- To develop the basic skills required to write network programs using Sockets

Course Outcomes:

- Understand all the LINUX utilities, and implement shell scripting.
- Mastery of simple LINUX filters
- Work confidently in Linux environment.
- Work with shell script to automate different tasks as Linux administration
- Mastery of the basic LINUX process structure and the LINUX file system.
- Familiarity of LINUX pipes and redirection, LINUX environment, traps, signals, filter parameters, filter options, LINUX contentions, and Regular Expressions.
- Mastery of at least one Shell scripting language

Unit-I		**Classes: 12**

Linux Utilities - File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking commands, Filters, Text processing utilities and Backup utilities, sed – scripts, operation, addresses, commands, applications, awk – execution, fields and records, scripts, operation, patterns, actions, functions, using system commands in awk.

Unit-II		**Classes: 12**

Working with the Bourne shell - Introduction, shell responsibilities, pipes and input Redirection, output redirection, here documents, running a shell script, the shell as a programming language, shell meta characters, file name substitution, shell variables, command substitution, shell commands, the environment, quoting, test command, control structures, arithmetic in shell, shell script examples.

Linux Files - File Concept, File System Structure, Inodes, File types, The standard I/O and formatted I/O, stream errors, kernel support for files, low level file access - usage of open, create, read, write, close, lseek, stat family, umask, dup, dup2, fcntl, file and record locking. file and directory management - chmod, chown, links(soft links & hard links - unlink, link, symlink), mkdir, rmdir, chdir, getcwd.

Unit-III		**Classes: 12**

Linux Process - Process concept, Kernel support for process, process attributes, process control - process creation, waiting for a process, process termination, zombie process, orphan process, system call interface for process management - fork, vfork, exit, wait, waitpid, exec family, Signals.

Inter process Communication - Introduction to IPC, pipes, FIFOs, Introduction to three types of IPC (Linux)-message queues, semaphores and shared memory, message Queues.

Unit-IV		**Classes: 12**

Semaphores - Kernel support for semaphores, Linux APIs for semaphores, file locking with semaphores

Shared Memory - Kernel support for shared memory, Linux APIs for shared memory, semaphore and shared memory example.

Unit-V		**Classes: 12**

Multi programming - Differences between threads and processes, Thread structure and uses. POSIX Thread APIs, Creating Threads, Thread Attributes, Thread Synchronization with semaphores and with Mutexes, Example programs

Sockets - Introduction to Linux Sockets, Socket system calls for connection oriented protocol and connectionless protocol, Example-client/server programs.

Text Books:

1. Unix Concepts and Applications, 4th Edition, Sumitabha Das, TMH, 2006.
2. Beginning Linux Programming, 4th Ed, N. Matthew, R. Stones, Wrox, Wiley India Edition, rp - 2008.
3. Unix and Shell programming, B.A. Forouzan and R.F. Gilberg, Cengage Learning.
4. Unix System Programming using C++, T. Chan, PHI.
5. Unix Network Programming, W.R. Stevens, PHI.

Reference Books:

1.Unix for programmers and users, 3rd Edition, Graham Glass, King Ables, Pearson Education,2003.

2. Advanced P System Programming with C and Unix, A. Hoover, Pearson.

3. Unix Programming, Kumar Saurabh, 1st Edition, Wiley India pvt Ltd

4.Programming in the Unix environment, 2nd Edition, W.R. Stevens, Pearson Education

Web References:

http://as.wiley.com/WileyCDA/WileyTitle/productCd-0764543733.html
http://gvpce.ac.in/syllabi/UNIX%20&%20SHELL%20Programming.pdf

E-Text Books:

https://zimslifeintcs.files.wordpress.com/2011/12/sumitabahdas.pdf

https://doc.lagout.org/operating%20system%20/linux/Beginning%20Linux%20Programming%2

SOFTWARE ENGINEERING

III – Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF00304	Core	3	-	-	3	40	60	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		

Course Objectives:

- Learn how to elicitate requirements and develop software life cycles.
- Understand the design considerations for enterprise integration and deployment.
- Analyze testing methodologies.
- Prepare a project plan for a software project that includes estimates of size and effort, a schedule, resource allocation, configuration control, and project risk.

Course Outcomes:

- An ability to apply knowledge of mathematics, science, and engineering
- An ability to design and conduct experiments, as well as to analyze and interpret data.
- An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
- An ability to function on multi-disciplinary teams.
- An ability to identify, formulates, and solves engineering problems.
- An understanding of professional and ethical responsibility.

Unit-I		**Classes: 10**

Software Process: Introduction to software engineering, software process, perspective and specialized process models; **Software project management**: Estimation: LOC and FP based estimation, COCOMO model; **Project scheduling:** Scheduling, earned value analysis, risk management

Unit-II		**Classes: 10**

Requirements Analysis And Specification: Functional and nonfunctional, user requirements, system requirements, software requirements document; Requirement engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management; Classical analysis: Structured system analysis, petri nets, data dictionary.

Unit-III		**Classes: 09**

Software Design **Design process:** Design concepts, design mode, design heuristic, architectural design architectural styles, architectural design, and architectural mapping using data flow. **User interface design:** Interface analysis, interface design; Component level design: Designing class based components, traditional components.		
Unit-IV		**Classes: 10**
Software testing fundamentals: Internal and external views of testing, white box testing, basis path testing, control structure testing, black box testing, regression testing, unit testing, integration testing, validation testing, system testing and debugging; Software implementation techniques: Coding practices, refactoring.		
Unit-V		**Classes: 07**
Project Management **Estimation:** FP based, LOC based, make/buy decision; COCOMO II: Planning, project plan, planning process, RFP risk management, identification, projection; RMMM: Scheduling and tracking, relationship between people and effort, task set and network, scheduling; EVA: Process and project metrics.		
Text Books:		
1. Roger S. Pressman, “Software Engineering – A Practitioner’s Approach”, Tata Mcgraw-Hill International Edition, 7th Edition, 2010. 2. Ian Somerville, “Software Engineering”, Pearson Education Asia, 9th Edition, 2011		
Reference Books:		
1. Rajib Mall, “Fundamentals of Software Engineering”, PHI Learning Private Limited, 3rd Edition, 2009. 2. Pankaj Jalote, “Software Engineering, A Precise Approach”, Wiley India, 1st Edition, 2010.		
Web References:		
1. http:*//www.softwareengineerinsider.com/articles/what-is-software-engineering.html* 2. *https://www.udacity.com/courses/software-engineering* 3. *http://www.tutorialspoint.com/software_engineering* 4. http://computingcareers.acm.org/?page_id=12 5. http:*//en.wikibooks.org/wiki/Introduction_to_Software_Engineering*		
E-Text Books:		
1. *http://*www.acadmix.com/eBooks_Download 2. http:*//www.freetechbooks.com/software-engineering-f15.html*		

JAVA PROGRAMMING

III – Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
17CF00305	PC	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	1	-	4	30	70	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		

Course Objectives:

- Learn the Java programming language: its syntax, idioms, patterns, and styles.
- Become comfortable with object oriented programming: Learn to think in objects
- Learn the essentials of the Java class library, and learn how to learn about other parts of the library when you need them.
- Introduce event driven Graphical User Interface (GUI) programming

Course Outcomes:

- Create Java programs that solve simple business problems.
- Validate user input.
- Perform a test plan to validate a Java program.
- Document a Java program.

Unit-I		**Classes: 10**

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

Inheritance – Inheritance concept, benefits of inheritance, Super classes and Sub classes, Member access rules, Inheritance hierarchies, super uses, preventing inheritance: final classes and methods, casting, polymorphism- dynamic binding, method overriding, abstract classes and methods, the Object class and its methods.

Unit-II		**Classes: 10**

Interfaces – Interfaces vs. Abstract classes, defining an interface, implementing interfaces, accessing implementations through interface references, extending interface.

Inner classes – Uses of inner classes, local inner classes, anonymous inner classes, static inner classes, examples.

Packages-Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages.

Unit-III **Classes: 10**

Data structures creation and manipulation in java – Introduction to Java Collections, Overview of Java Collection frame work, Commonly used Collection classes – ArrayList, LinkedList, HashSet, HashMap, TreeMap,

Collection Interfaces – Collection, Set, List, Map, Legacy Collection classes – Vector, Hashtable, Stack, Dictionary(abstract), Enumeration interface, Iteration over Collections – Iterator interface, ListIterator interface. Other Utility classes – StringTokenizer, Formatter, Random, Scanner, Observable, Using java.util.

Files – streams- byte streams, character streams, text Input/output, binary input/output, random access file operations, File management using File class, Using java.io.

Unit-IV **Classes: 08**

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

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

Unit-V **Classes: 07**

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

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

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

Text Books:

1. Java: the complete reference, 7th editon, Herbert Schildt, TMH.
2. Java for Programmers, P.J.Deitel and H.M.Deitel, Pearson education / Java: How to Program P.J.Deitel and H.M.Deitel ,8th edition, PHI.

Reference Books:

1. Core Java, Volume 1-Fundamentals, eighth edition, Cay S.Horstmann and Gary Cornell, Pearson eduction.
2. Java Programming, D.S.Malik, Cengage Learning.
3. Object Oriented Programming with Java, B.Eswara Reddy, T.V.Suresh Kumar, P.Raghavan, Pearson-Sanguine.
4. An introduction to Java programming and object oriented application development, R.A. Johnson-Cengage Learning.
5. Advanced Programming in Java2, K.Somasundaram, Jaico Publishing House.
6. Starting out with Java, T.Gaddis, dreamtech India Pvt. Ltd.
7. Object Oriented Programming with Java, R.Buyya, S.T.Selvi, X.Chu, TMH.

Web References:

- java.sun.com
- ibm.com/developerworks/java
- https://www.javatpoint.com/java-tutorial
- https://www.tutorialspoint.com/java/
- https://www.udemy.com/java-tutorial/
- *https://www.w3schools.in/java-tutorial/*

E-Text Books:

- Thinking in Java (3rd edition)
- The Java Language Specification, Java SE 8
- The Java Tutorials
- Think Java (How to Think Like a Computer Scientist)

DATABASE MANAGEMENT SYSTEMS LAB

III - Semester								
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF00306	Core	**_**	**_**	**3**	**2**	**40**	**60**	**100**
Contact Classes: Nil	Tutorial Classes: Nil		Practical Classes: 45			Total Classes:45		

Course Objectives:

- Implement the basic knowledge of SQL queries and relational algebra.
- Construct database models for different database applications.
- Apply normalization techniques for refining of databases.
- Practice various triggers, procedures, and cursors using PL/SQL.

Course Outcomes:

- Gain a good understanding of the architecture and functioning of database management systems as well as associated tools and techniques, principles of data modeling using entity relationship and develop a good database design and normalization techniques to normalize a database.
- Understand the use of structured query language and its syntax, transactions, database recovery and techniques for query optimization.

List of Programs

Exp-1	**Creation of Tables and DML Commands**

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints.

Example:- Select the roll number and name of the student who secured fourth rank in the class.

Exp -2	**Using Aggregate and Conversion Functions**

<table>
<tr><td colspan="2">1. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
2. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)</td></tr>
<tr><td><b>Exp -3</b></td><td><b>Queries Using Aggregate Functions</b></td></tr>
<tr><td colspan="2">1. Creation of simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
2. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
3. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.</td></tr>
<tr><td><b>Exp -4</b></td><td><b>Programs on PL/SQL</b></td></tr>
<tr><td colspan="2">1) Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT–IN Exceptions, USE defined Exceptions, RAISE-APPLICATION ERROR.
2) Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.</td></tr>
<tr><td><b>Exp -5</b></td><td><b>Procedures and Functions</b></td></tr>
<tr><td colspan="2">1. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
2. Program development using creation of package specification, package bodies, private objects, package variables and cursors and calling stored packages.</td></tr>
<tr><td><b>Exp -6</b></td><td><b>Triggers and Cursors</b></td></tr>
<tr><td colspan="2">1. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
2. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers</td></tr>
<tr><td><b>Exp -7</b></td><td><b>CASE STUDY: Book Publishing Company</b></td></tr>
</table>

A publishing company produces scientific books on various subjects. The books are written by authors who specialize in one particular subject. The company employs editors who, not necessarily being specialists in a particular area, each take sole responsibility for editing one or more publications.

A publication covers essentially one of the specialist subjects and is normally written by a single author. When writing a particular book, each author works with on editor, but may submit another work for publication to be supervised by other editors. To improve their competitiveness, the company tries to employ a variety of authors, more than one author being a specialist in a particular subject for the above case study, do the following:

1. Analyze the data required.
2. Normalize the attributes.
3. Create the logical data model using E-R diagrams

Exp -8 CASE STUDY: General Hospital

A General Hospital consists of a number of specialized wards (such as Maternity, Pediatric, Oncology, etc). Each ward hosts a number of patients, who were admitted on the recommendation of their own GP and confirmed by a consultant employed by the Hospital. On admission, the personal details of every patient are recorded. A separate register is to be held to store the information of the tests undertaken and the results of a prescribed treatment. A number of tests may be conducted for each patient. Each patient is assigned to one leading consultant but may be examined by another doctor, if required. Doctors are specialists in some branch of medicine and may be leading consultants for a number of patients, not necessarily from the same ward. For the above case study, do the following.

1. Analyze the data required.
2. Normalize the attributes.
3. Create the logical data model using E-R diagrams.

Exp -9 CASE STUDY: Car Rental Company

A database is to be designed for a car rental company. The information required includes a description of cars, subcontractors (i.e. garages), company expenditures, company revenues and customers. Cars are to be described by such data as: make, model, year of production, engine size, fuel type, number of passengers, registration number, purchase price, purchase date, rent price and insurance details. It is the company policy not to keep any car for a period exceeding one year. All major repairs and maintenance are done by subcontractors (i.e. franchised garages), with whom CRC has long-term agreements. Therefore the data about garages to be kept in the database includes garage names, addresses, range of services and the like. Some garages require payments immediately after a repair has been made; with others CRC has made arrangements for credit facilities. Company expenditures are to be registered for all outgoings connected with purchases, repairs, maintenance, insurance etc. Similarly the cash inflow coming from all sources: Car hire, car sales, insurance claims must be kept of file. CRC maintains a reasonably stable client base. For this privileged category of customers special credit card facilities are provided. These customers may also book in advance a particular car. These reservations can be made for any period of time up to one month. Casual customers must pay a deposit for an estimated time of rental, unless they wish to pay by credit card. All major credit cards are accepted. Personal details such as name, address, telephone number, driving license, number about each customer are kept in the database. For the above case study, do the following:

1. Analyze the data required.

2. Normalize the attributes.
3. Create the logical data model using E-R diagrams.

Exp -10

CASE STUDY: Student Progress Monitoring System

A database is to be designed for a college to monitor students' progress throughout their course of study. The students are reading for a degree (such as BA, BA (Hons) M.Sc., etc) within the framework of the modular system. The college provides a number of modules, each being characterized by its code, title, and credit value, module leader, teaching staff and the department they come from. A module is coordinated by a module leader who shares teaching duties with one or more lecturers. A lecturer may teach (and be a module leader for) more than one module. Students are free to choose any module they wish but the following rules must be observed: Some modules require pre- requisites modules and some degree programmes have compulsory modules. The database is also to contain some information about students including their numbers, names, addresses, degrees they read for, and their past performance i.e. modules taken and examination results. For the above case study, do the following:

1. Analyze the data required.
2. Normalize the attributes.
3. Create the logical data model i.e., ER diagrams.
4. Comprehend the data given in the case study by creating respective tables with primary keys and foreign keys wherever required.
5. Insert values into the tables created (Be vigilant about Master- Slave tables).
6. Display the Students who have taken M.Sc course.
7. Display the Module code and Number of Modules taught by each Lecturer.
8. Retrieve the Lecturer names who are not Module Leaders.
9. Display the Department name which offers ‘English’ module.
10. Retrieve the Prerequisite Courses offered by every Department (with Department names).
11. Present the Lecturer ID and Name who teaches ‘Mathematics’.
12. Discover the number of years a Module is taught.
13. List out all the Faculties who work for ‘Statistics’ Department.
14. List out the number of Modules taught by each Module Leader.
15. List out the number of Modules taught by a particular Lecturer.
16. Create a view which contains the fields of both Department and Module tables. (Hint- The fields like Module code, title, credit, Department code and its name).
17. Update the credits of all the prerequisite courses to 5. Delete the Module ‘History’ from the Module table.

REFERENCE BOOKS

1. Ramez Elmasri, Shamkant, B. Navathe, “Database Systems”, Pearson Education, 6th Edition, 2013. Peter Rob, Carles Coronel, “Database System Concepts”, Cengage Learning, 7th Edition, 2008.

2. Database Management Systems, Peter Rob, A.Ananda Rao and Carlos Coronel, Cengage Learning.

3. ORACLE PL/SQL by example. Benjamin Rosenzweig, Elena Silvestrova, Pearson Education 3rd Edition

4. SQL & PL/SQL for Oracle 10g, Black Book, Dr.P.S. Deshpande

WEB REFERENCES

1. http://www.scoopworld.in

LINUX PROGRAMMING LAB

III - Semester								
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
17CF00307	Core	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		-	**-**	**3**	**2**	**40**	**60**	**100**
Contact Classes: Nil	Tutorial Classes: Nil			Practical Classes: 45		Total Classes: 45		

Course Objectives:

- Learning and practice the Linux utilities
- Practice programs on shell scripts.

Course Outcomes:

- Work confidently in Unix/Linux environment
- Write shell scripts to automate various tasks
- Master the basics of linux administration

1. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
2. Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.
3. Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
4. Write a shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.
5. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.
6. Write a shell script to list all of the directory files in a directory.
8. Write an awk script to count the number of lines in a file that do not contain vowels.
9. Write an awk script to find the number of characters, words and lines in a file.
10. Write a c program that makes a copy of a file using standard I/O and system calls.
11. Implement in C the following Unix commands using System calls
    A . cat B. ls C. mv
12. Write a program that takes one or more file/directory names as command line input and reports the following information on the file.
    A. File type. B. Number of links.
    C. Time of last access. D. Read, Write and Execute permissions.
13. Write a C program to emulate the Unix ls –l command.
14. Write a C program to list for every file in a directory, its inode number and file name.
15. Write a C program that demonstrates redirection of standard output to a file.Ex: ls > f1.
16. Write a C program to create a child process and allow the parent to display “parent” and the child to display “child” on the screen.
17. Write a C program to create a Zombie process.
18. Write a C program that illustrates how an orphan is created.
19. Write a C program that illustrates how to execute two commands concurrently with a command pipe. Ex:- ls –l | sort

20. Write C programs that illustrate communication between two unrelated processes using named pipe.
21. Write a C program (sender.c) to create a message queue with read and write permissions to write 3 messages to it with different priority numbers.
22. Write a C program (receiver.c) that receives the messages (from the above message queue as specified in (21)) and displays them
23 Write a C program that implements a producer-consumer system with two processes. (Using Semaphores).
24.Write a C program that illustrates two processes communicating using shared memory
25. Write client and server programs (using c) for interaction between server and client processes using Unix Domain sockets
26. Write client and server programs (using c) for interaction between server and client processes using Internet Domain sockets.

References:

1. Advanced Unix Programming, N.B.Venkateswarulu, BS Publications.
2. Unix and Shell programming, B.A.Forouzan and R.F.Gilberg, Cengage Learning.
3. Unix and Shell Programming, M.G. Venkatesh Murthy, Pearson Education, 2005.
4. Unix Shells by Example, 4th Edition, Elllie Quigley, Pearson Education.

5. Sed and Awk, O.Dougherty&A.Robbins,2nd edition,SPD.

JAVA PROGRAMING LAB

III Semester

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF00308	Core	-	-	3	2	40	60	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 39				Total Classes: 39		

Course Objectives:

- Practice object-oriented programs and build java applications.
- Implement java programs for establishing interfaces.
- Implement sample programs for developing reusable software components.
- Create database connectivity in java and implement GUI applications.

Course Outcomes:

- Create a software application using the Java programming language.
- Debug a software application written in the Java programming language.
- Test a software application written in the Java programming language

List of Experiments

1.Basic Programs

a. Try debug step by step with small program of about 10 to 15 lines which contains at least one if else condition and a for loop.

b. Write a java program that prints all real solutions to the quadratic equation $ax^2+bx+c=0$. Read in a, b, c and use the quadratic formula.

c. The Fibonacci sequence is defined by the following rule. The first two values in the sequence are 1 and 1. Every subsequent value is the sum of the two values preceding it. Write a java program that uses both recursive and non-recursive functions.

2.Matrices, Overloading, Overriding

a. Write a java program to multiply two given matrices.

b. Write a java program to implement method overloading and constructors overloading.

c. Write a java program to implement method overriding.

3.Palindrome, Abstract Class

a. Write a java program to check whether a given string is palindrome.

b. Write a java program for sorting a given list of names in ascending order.

c. Write a java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.

4.Interface

Write a program that creates a user interface to perform integer division. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 and Num2 were not integers, the program would throw a Number Format Exception. If Num2 were zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.

5.AWT

Write a java program that creates menu which appears similar to the menu of notepad application of the Microsoft windows or any editor of your choice.

6.Files

a. Write a java program that reads a file name from the user, and then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.

b. Write a java program that displays the number of characters, lines and words in a text file.

c. Write a java program that reads a file and displays the file on the screen with line number before each line.

7.Files ctd..

a. Suppose that table named table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using labels in grid layout.

b. Write a java program that connects to a database using JDBC and does add, delete, modify and retrieve operations.

8.Keyboard Events

Write a java program to handle keyboard events

9.Files ctd.

a. Write a java program that takes tab separated data (one record per line) from a text file and insert them into a database.
b. Write a java program that prints the metadata of a given table.

10. Traffic Light

Write a java program that simulates a traffic light. The program lets the user select one of three lights: Red, Yellow or Green with radio buttons. On selecting a button an appropriate message with "STOP" or "READY" or "GO" should appear above the buttons in selected color. Initially, there is no message shown.

11.Mouse Events

a. Write a java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired. Use adapter classes.
b. Write a java program to demonstrate the key event handlers.

12.Calculator

Write a java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +,-,*, % operations. Add a text field to display the result. Handle any possible exception like divided by zero.

13.Applet

a. Develop an applet that displays a simple message.
b. Develop an applet that receives an integer in one text field and computes its factorial value and returns it in another text field, when the button named "compute" is clicked.

Reference Books:

1. P. J. Deitel, H. M. Deitel, "Java for Programmers", Pearson Education, PHI, 4th Edition, 2007.
2. P. Radha Krishna, "Object Oriented Programming through Java", Universities Press, 2nd Edition, 2007
3. Bruce Eckel, "Thinking in Java", Pearson Education, 4th Edition, 2006.
4. Sachin Malhotra, Saurabh Chaudhary, "Programming in Java", Oxford University Press, 5th Edition, 2010.

Web References:

1. www.niecdelhi.ac.in
2. https://www.linkedin.com/in/achin-jain-85061412
3. www.rank1infotech.com

OBJECT ORIENTED ANALYSIS AND DESIGN

IV – Semester								
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
17CF00401	Core	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	**1**	**-**	**4**	**40**	**60**	**100**
Contact Classes:45	Tutorial Classes:15		Practical Classes: Nil			Total Classes:60		

Course Objectives:

The course should enable the students to:

- Develop the skills to analyze and design object-oriented problems.
- Create design patterns to solve problems based on object oriented concepts.
- Understand the various processes and techniques for building object-oriented software systems.
- Prepare unified modeling techniques for case studies.

Course Outcomes:

- Understand the principles of object oriented design.
- Understand and apply the programming language Java in object oriented software development.

- Understand and apply the tool UML in object oriented software modelling.
- Develop a simple software application using the object oriented approach.

Unit – I		Classes:10

Introduction: The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems, Evolution of Object Model, Foundation of Object Model, Elements of Object Model, Applying the Object Model.

Unit – II		Classes:10

Classes and Objects: Nature of object, Relationships among objects, Nature of a Class, Relationship among Classes, Interplay of Classes and Objects, Identifying Classes and Objects, Importance of Proper Classification, Identifying Classes and Objects, Key abstractions and Mechanisms.

Unit – III		Classes:08

Introduction to UML: Why model, Conceptual model of UML, Architecture, Classes, Relationships, Common Mechanisms, Class diagrams, Object diagrams.

Unit – IV		Classes:09

Structural Modeling: Package Diagram, Composite Structure Diagram, Component Diagram, Deployment Diagram, Profile diagram.

Unit –V		Classes:08

Behavioural Modelling: Use Case Diagram, Activity Diagrams, State Machine Diagrams, Sequence Diagram, Communication Diagram, Timing Diagram, Interaction Overview Diagram.

Text Books:

1. Object Oriented Analysis And Design with Applications", Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston, PEARSON, 3rd edition, 2013.
2. "The Unified Modeling Language User Guide", Grady Booch, James Rumbaugh, Ivar Jacobson,

   PEARSON 12th Impression, 2012.

References:

1. Simon Bennett, Steve Mc Robb, Ray Farmer, "Object Oriented Systems Analysis and Design Using UML", Mc Grew Hill Education, 4th Edition, 2010.
2. Pascal Roques, "Modeling Software Systems Using UML2", WILEY- Dreamtech India Pvt. Ltd, 2nd Edition, 2007.

Web References

1. https://www.tutorialspoint.com/uml/uml_overview.html
2. https://www.utdallas.edu/~chung/OOAD/M03_1_StructuralDiagrams.ppt
3. https://onedrive.live.com/download?cid=99CBBF765926367

E-Text Books:

1. https://www.utdallas.edu/UML2.0/Rumbaugh
2. https://www.utdallas.edu/~chung/SP/applying-uml-and-patterns.pdf

WEB TECHNOLOGIES

IV – Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF00402	Core	3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		

Course Objectives:

- To learn the HTML tags and its usage for creating web interfaces.
- To understand various data handling objects in javascript and advanced java technologies.
- To understand installation of web servers.
- To learn about database connectivity and run applications.

Course Objectives:

- Learn the basic concepts& techniques of java.
- Learn the advanced concepts of java.
- Generate an application based upon the concepts of java & advance java.

Unit-I		**Classes: 10**

Review of HTML: Common tags, HTML Tables, lists, form, frames and formatting internal linking, Complex HTML forms.

Scripting Language: Java Scripts, Control structures, functions, arrays & objects, DHTML, CSS, event model.

<table>
<tr><td><b>Unit-II</b></td><td></td><td><b>Classes: 9</b></td></tr>
<tr><td colspan="3"><b>XML</b>: Document type definition, XML Schemas, Document Object model, Presenting XML, Using XML Processors: DOM and SAX
<b>PHP</b>: PHP installation and Introduction, Loops, String Functions in PHP, PHP Email Function, PHP Basics, Variables, Arrays in PHP with Attributes, Date & Time, Image Uploading, File handling in PHP, Functions in PHP, Errors handling in PHP.</td></tr>
<tr><td><b>Unit-III</b></td><td></td><td><b>Classes: 11</b></td></tr>
<tr><td colspan="3"><b>Java Beans</b>: Introduction to Java Beans, Advantages of Java Beans, BDK Introspection, Using Bound properties, Bean Info Interface, Constrained properties, Persistence, Customizes, Java Beans API.
Web Servers and Servlets: Tomcat web server, Introduction to Servelets: Lifecycle of a Serverlet, JSDK, The Servelet API, The javax.servelet Package, Reading Servelet parameters, Reading Initialization parameters. The javax.servelet HTTP package, Handling Http Request & Responses, Using Cookies-Session Tracking, Security Issues.</td></tr>
<tr><td><b>Unit-IV</b></td><td><b>Java Server Pages</b></td><td><b>Classes: 08</b></td></tr>
<tr><td colspan="3"><b>Introduction to JSP</b>: The Problem with Servelet. The Anatomy of a JSP Page, JSP Processing, JSP Application Design with MVC. Setting Up and JSP Environment: Installing the Java Software development Kit, Tomcat Server & Testing Tomcat.
<b>JSP Application Development</b>: Generating Dynamic Content, Using Scripting Elements Implicit JSP Objects, Conditional Processing – Displaying Values Using an Expression to Set an Attribute, Declaring Variables and Methods Error Handling and Debugging Sharing Data Between JSP pages, Requests, and Users Passing Control and Date between Pages – Sharing Session and Application Data – Memory Usage Considerations.</td></tr>
<tr><td><b>Unit-V</b></td><td><b>Database Access</b></td><td><b>Classes: 07</b></td></tr>
<tr><td colspan="3"><b>Database Access</b>: Database Programming using JDBC, Studying Javax.sql.* package, Accessing a Database from a JSP Page, Application – Specific Database Actions, Deploying JAVA Beans in a JSP Page.</td></tr>
<tr><td colspan="3"><b>Text Books:</b></td></tr>
<tr><td colspan="3">1. Web Programming, building internet applications, Chris Bates 2nd edition, WILEY Dream tech.
2. The complete Reference Java Seventh Edition by Herbert Schildt. TMH.
3. Java Server Pages –Hans Bergsten, SPD O‟ Reilly
4. PHP for Absolute Beginners 2nd ed. Edition by Jason Lengstorf (Author), Thomas Blom Hansen (Author)</td></tr>
<tr><td colspan="3"><b>Reference Books: Reference Books</b></td></tr>
<tr><td colspan="3">4. Programming world wide web-Sebesta, Pearson
5.Core Servlets and Java Server Pages Volume 1: Core Technologies By Marty Hall and Larry Brown Pearson
6. Internet and World Wide Web – How to program by Dietel and Nieto PHI/Pearson Education Asia.
7. Jakarta Struts Cookbook , Bill Siggelkow, S P D O’Reilly for chap
8. An Introduction to web Design and Programming –Wang-Thomson
9. Web Applications Technologies Concepts-Knuckles, John Wiley</td></tr>
<tr><td colspan="3"><b>Web References:</b></td></tr>
</table>

http://www.jkmaterials.yolasite.com/resources/materials/webtechnology/JWT/Web-Technologies-Notes.pdf
E-Text Books:
http://iiti.ac.in/people/~tanimad/JavaTheCompleteReference.pdf http://www.sebizfinishingschool.com/ebook/java/Java%202%20%20The%20Complete%20Reference%20(5th%20Edition).pdf https://rungringjung.files.wordpress.com/2010/10/javaserver-pages-jsp.pdf

DATA WAREHOUSING AND DATA MINING

IV- Semester								
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
17CF00403	Core	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	**1**	**-**	**4**	**40**	**60**	**100**
Contact Classes:45	Tutorial Classes:15		Practical Classes: Nil			Total Classes:60		

Course Objectives:

The course should enable the students to:

- Understand data warehouse and online analytical processing technology for data mining.
- Make mining association with rules in large databases, do classification and prediction with different techniques.
- Conceptualize the architecture of a data warehouse and the need for pre-processing.
- Develop and understand data mining applications and trends of data mining.
- Analyze the major techniques of preprocessing for different types of data.

Course Outcomes:

- To understand the basic principles, concepts and applications of data warehousing and data mining,
- Ability to do Conceptual, Logical, and Physical design of Data Warehouses OLAP applications and OLAP deployment
- Have a good knowledge of the fundamental concepts that provide the foundation of data mining.

Unit - I	**Data Warehousing**	**Classes:10**

Introduction to data mining: Motivation, importance, definition of data mining, kinds of data mining, kinds of patterns, data mining technologies, kinds of applications targeted, major issues in data mining; Preprocessing: data objects and attribute types, basic statistical descriptions of data, data visualization, data quality, data cleaning, data integration, data reduction, data transformation and data discretization.

Unit - II	**Business Analysis**	**Classes:08**

Data warehouse and OLAP technology for data mining, what is a data warehouse, multi-dimensional data model, data warehouse architecture, data warehouse implementation, development of data cube technology, data warehousing to data mining; Data preprocessing: Data summarization, data cleaning, data integration and transformation data reduction, discretization and concept hierarchy generation.

Unit - III	**Data Mining**	**Classes:10**

Data mining primitives: Define a data mining, data mining query language, designing graphical user interfaces based on a data mining query language.

Concept description: Characterization and comparison, analytical characterization, mining class comparison, mining, descriptive statistical measures in large database

Unit - IV	**Association Rule Mining And Classification**	**Classes:09**

Mining frequent patterns, associations and correlations, mining methods, mining various kinds of association rules, correlation analysis, constraint based association mining, classification and prediction, basic concepts, decision tree induction, Bayesian classification, rule based classification, classification by back propagation.

Unit –V	**Clustering And Trends In Data Mining**	**Classes:08**

Cluster analysis: Types of data, categorization of major clustering methods, K-means partitioning methods, hierarchical methods, density based methods, grid based methods, model based clustering methods, clustering, high dimensional data, constraint based cluster analysis, outlier analysis; Trends in data mining: Data mining applications, data mining system products and research prototypes, social impacts of data mining.

Text Books:

1. Jiawei Han, Michelin Kamber, "Data Mining-Concepts and techniques", Morgan Kaufmann Publishers, Elsevier, 2nd Edition, 2006.
2. Alex Berson, Stephen J. Smith, "Data Warehousing Data Mining and OLAP", Tata McGraw-Hill, 2nd Edition, 2007.

References:

1. Arun K Pujari, "Data Mining techniques", Universities Press, 3rd Edition, 2005
2. Pualraj Ponnaiah, "Data Warehousing Fundamentals", Wiley, Student Edition.2004.
3. E. Balagurusamy, "Programming in ANSI C", Mc Graw Hill Education, 6th Edition, 2012.
4. Ralph Kimball, "The Data Warehouse Life Cycle Toolkit", Wiley, Student Edition, 2006.
5. Vikram Pudi, P Radha Krishna, "Data Mining", Oxford University, 1st Edition, 2007.

Web References

1.http://www.anderson.ucla.edu
2.https://www.smartzworld.com
3.http://iiscs.wssu.edu

E-Text Books:

1.https://www.cisco.com/application/pdf/en/us/guest/products/ps2011/c2001/ccmigration_09186a00802342cf.pdf
2.https://www.jntubook.com
3. http://ftp.utcluj.ro/pub/users/cemil/dwdm/dwdm_Intro/0_5311707.pdf.

.NET TECHNOLOGIES

IV – Semester

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
17CF00404	Elective -1	3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		

Course Objectives:

- Understand the basic .net technologies
- Understand the syntax of basic C# programming constructs.
- Create and use new types (enumerations, classes, and structures), and understand the differences between reference types and value types.
- Implement custom collection classes that support enumeration.
- Explore on dynamic languages for creating web applications.

Course Outcomes:

- write programming code that makes use of: structured programming constructs of sequence, selection and repetition; variables, symbolic constants, structures, arrays, simple text files and built-in functions and methods for conversion between various data types; selected built-in string handling functions and methods; simple classes and objects
- Design and write code that uses principles of modular design with use of sub procedures and function
- Given user requirements, design and implement a well structured and documented coded solution to solve simple business problems develop a test plan and create useful test data to test written code

Unit-I		**Classes: 10**

Introducing C# and the .NET platform: The philosophy of .NET, the .NET solution, building blocks of the .NET platform(the CLR, CTS, and CLS), an overview of .NET assemblies, understanding the CTS, CLS, and CLR, the assembly / namespace / type distinction, exploring an assembly using ildasm.exe, exploring an assembly using reflector, the platform independent nature of .NET; **Building C# application:** The role of the .NET framework 4.0 SDK, building C# applications using csc.exe, building NET applications using notepad++, building .NET applications using C# development, building .NET applications using visual C# 2010 express, building .NET applications using visual studio 2010.

Unit-II		**Classes: 10**

Core C# programming constructs part - I: The anatomy of simple C# program, environment class, the system, Console class, system data types and C# shorthand notation, working with string data, narrowing and widening data type local variables, C# iteration constructs, decision constructs and the relational / equality operators; **Core programming constructs part-II:** Methods and parameter modifiers, understanding C# arrays, understanding the enum type, understanding the structure type, understanding value types and reference types, understanding C# nullable type.

Unit-III		**Classes: 10**

Inheritance: The basic mechanics of inheritance, revising visual studio class diagrams, defining the pillars of OOP, the first pillar, the second pillar of OOP, the third pillar of OOP, understanding base class / derived class casting rules, the master parent class.

Understanding structured exception handling: ODE to errors, bugs, and exceptions, the role of .NET exception handling, the simplest possible example, configuring the state of an exception, types of exceptions, processing multiple exceptions.

Unit-IV		**Classes: 08**

Delegates and events: Understanding the .NET delegate type, defining a delegate type in C#, the system multicast delegate and system, delegate base classes, the simple possible delegate example, sending object state notification using delegates;

Programming with .NET assemblies: Configuring .NET assemblies, defining custom namespaces, the role of .NET assemblies, understanding the format of a .NET assembly, building and consuming a single-file assembly, building and consuming a multi file assembly, understanding private assembly, understanding shared assembly, consuming a shared assembly, configuring shared assemblies, understanding publisher policy assemblies, understanding the<codebase> element, the system, configuration namespace.

Unit-V		**Classes: 07**

ADO.NET part - I: The connected layer, a high-level definition of ADO.NET, understanding ADO.NET data provider, additional ADO.NET namespaces, the types of the system, data, namespace, abstracting data providers using interfaces, creating the auto lot database, the ADO.NET data provider factory model, understanding the connected layer of ADO.NET, working with data readers, building a reusable data access library, creating a console ui-based front end, understanding database transactions;

ADO.NET part - II: Disconnected layer understanding the disconnected layer of ADO.NET, understanding the role of the dataset, working with data columns, working with data rows, working with data tables, binding with data adapters, adding disconnected functionality to autolotdal.dll, multi tabled dataset objects and data relationships, the windows forms database code into a class library, programming with LINQ to dataset.

Text Books:

1. Andrew Troelsen, "Pro C# and the .NET 4 Platform", Springer (India) Private Limited, New Delhi, India, 5th Edition, 2010.
2. S. Thamarai Selvi, R. Murugesan, “A Textbook on C#”, Pearson Education, 1st Edition, 2003.

Reference Books:

1. E. Balagurusamy, "Programming in C#", Tata Mcgraw-Hill, New Delhi, India, 5th Edition, 2004.
2. Herbert Schildt, "The Complete Reference: C#", Tata Mcgraw-Hill, New Delhi, India, 7th Edition, 2004.
3. Simon Robinson, Christian Nagel, Karli Watson, Jay Gl, "Professional C#", Wiley& Sons, India, 3rd Edition, 2006.

Web References:

1. https://www.cs.colorado.edu/~kena/classes/5448/
2. https://www.c-sharpcorner.com/
3. https://www.tutorialspoint.com/csharp/
4. http://www.completecsharptutorial.com/

E-Text Books:

1. http://www.c-sharpcorner.com/ebooks/
2. http://www.freebookcentre.net/MicroSoftTech/Microsoft-Dotnet-Books-Download.html

INFORMATION SECURITY

IV- Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
17CF00405	Elective -1	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**

		3	-	-	3	40	60	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		

Course Objectives:

The course should enable the students to:

- Learn the basic categories of threats to computers and networks.
- Understand various cryptographic algorithms and be familiar with public-key cryptography.
- Apply authentication functions for providing effective security.
- Analyze the application protocols to provide web security.
- Discuss the place of ethics in the Information Security Area.

Course Outcomes:

- Identify a range of security and privacy issues and threats that drive the need for security
- Understand the three security principles Confidentiality, Integrity and Availability (C,I,A) and how they relate to security threats and technologies
- Identify a range of security paradigms and models and understand how they can be deployed in a security strategy to protect information and preserve privacy
- Understand cryptographic technologies and how they can be deployed to protect information and preserve privacy

Unit-I	**Attacks on Computers and Computer Security**	**Classes: 08**

Attacks on computers and computer security: Introduction, the need for security, security approaches, principles of security, types of security attacks, security services, security mechanism, a model for network security; Cryptography concepts and techniques: Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, steganography, key range and key size, possible types of attacks.

Unit-II	**Symmetric Key Ciphers**	**Classes: 10**

Symmetric key ciphers: Block cipher principles and algorithms (DES, AES, Blowfish), differential and linear cryptanalysis, block cipher modes of operation, stream ciphers,RC4 location, and placement of encryption function, key distribution; Asymmetric key ciphers: Principles of public key cryptosystems, algorithms (RSA Diffie- Helman, ECC) key distribution.

Unit-III	**Message Authentication Algorithm and Hash Functions**	**Classes: 08**

Message authentication algorithm and hash functions: Authentication requirements, functions, message, authentication codes, hash functions, secure hash algorithm, whirlpool, HMAC, CMAC, digital signatures, knapsack algorithm.

Authentication application: Kerberos, X.509 authentication service, public – key infrastructure, biometric authentication.

Unit-IV	**E-Mail Security**	**Classes: 10**

E-mail security: Pretty good privacy; S/MIMI IP Security: IP security overview, IP security architecture, authentication header, encapsulating security payload, combines security associations, key management.

Unit-V	**Web Security**	**Classes: 09**

Web security: Web security considerations, secure socket layer and transport layer security, secure electronic transaction intruders; Virus and firewalls: Intruders, intrusion detection password management, virus and related threats, countermeasures, firewall design principles; Types of firewalls case studies on cryptography and security: Secure inter-branch payment transactions, cross site scripting vulnerability, virtual electronics.

Text Books:

1. William Stallings, “Cryptography and Network Security”, Pearson Education, 4th Edition, 2005.
2. AtulKahate, “Cryptography and Network Security”, McGraw Hill, 2nd Edition, 2009.

Reference Books:

1. C K Shymala, N Harini, Dr. T R Padmanabhan, “Cryptography and Network Security”, Wiley India, 1st Edition, 2016.
2. Behrouz A. Forouzan, Debdeep Mukhopadhyay , “Cryptography and Network Security”, McGraw Hill, 2nd Edition, 2010.

Web References:

1. http://bookboon.com/en/search?q=INFORMATION+SECURITY
2. https://books.google.co.in/books/about/Cryptography_Network_Security_Sie_2E.html?id=Kokjwdf0E7QC
3. https://books.google.co.in/books/about/Information_Security.html?id=Bh45pU0_E_4C

E-Text Books:

1. https://books.google.co.in/books/about/Information_Security.html
2. http://www.amazon.in/Cryptography-Network-Security-Behrouz-Forouzan/dp/007070208X

COMPUTER GRAPHICS

IV-Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
17CF00406	Elective -1	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**

		3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		

Course Objectives:

- The course should enable the students to:
- Learn the basic representation of graphics.
- Understand various graphic algorithms.
- Understand various 3D-graphic algorithms

Course Outcomes:

- Critical understanding of the theory of 2D and 3D transformations, projection and viewing
- Ability to find & combine relevant sources and synthesize designs
- Detailed knowledge of the graphics pipeline
- Detailed knowledge of shading and texture mapping algorithms
- Broad knowledge of 3D modeling and rendering techniques
- Ability to understand, design and implement scene graphs

Unit-I		**Classes: 10**

Introduction: Application areas of Computer Graphics, overview of graphics systems, video display devices, raster-scan systems, random scan systems, graphics monitors and work stations and input devices. **Output primitives:** Points and lines, line drawing algorithms, mid-point circle and ellipse algorithms. Filled area primitives: Scan line polygon fill algorithm, boundary-fill and flood-fill algorithms.

Unit-II		**Classes: 10**

2-D Geometrical transforms: Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms transformations between coordinate systems.

Unit-III		**Classes: 10**

2-D Viewing: The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, Cohen-Sutherland and Liang - Barsky line clipping algorithms, Sutherland –Hodgeman polygon clipping algorithm

Unit-IV		**Classes: 08**

3-D Object representation: Polygon surfaces, quadric surfaces, spline representation, Hermite curve, Bezier curve and B-spline curves, Bezier and B-spline surfaces. Basic illumination models, polygon rendering methods.

Unit-V		**Classes: 07**

Visible surface detection methods: Translation, rotation, scaling, reflection and shear transformations, composite transformations, 3-D viewing: Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping.
Computer animation: Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems, motion specifications.

Text Books:

1. “Computer Graphics C version”, Donald Hearn and M. Pauline Baker, Pearson education.
2. “Computer Graphics Principles & practice”, second edition in C, Foley, VanDam, Feiner and Hughes, Pearson Education.

References:

1. “Computer Graphics Second edition”, Zhigand xiang, Roy Plastock, Schaum‟s outlines, Tata Mc Graw hill edition.
2. “Procedural elements for Computer Graphics”, David F Rogers, Tata Mc Graw hill, 2nd edition.
5. “Principles of Interactive Computer Graphics”, Neuman and Sproul, TMH.
6. “Principles of Computer Graphics”, Shalini, Govil-Pai, Springer.
7. “Computer Graphics”, Steven Harrington, TMH
8. Computer Graphics, F.S.Hill, S.M.Kelley ,PHI.
9. Computer Graphics, P.Shirley, Steve Marschner & Others, Cengage Learning.
10. Computer Graphics & Animation, M.C.Trivedi, Jaico Publishing House.

Web References:

http://www.alljntuworld.in/download/computer-graphics-cg-materials-notes/

E-Text Books:

https://www.tutorialspoint.com/computer_graphics/

ARTIFICIAL INTELLIGENCE

IV- Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
17CF00407	Elective -2	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**

		-	4	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		

Course Objectives:

- Select a search algorithm for a problem and estimate its time and space complexities.
- Possess the skill for representing knowledge using the appropriate technique for a given problem
- Possess the ability to apply AI techniques to solve problems of game playing, expert systems, machine learning and natural language processing.

Course Outcomes:

- Describe the key components of the artificial intelligence (AI) field and its relation and role in Computer Science;
- Identify and describe artificial intelligence techniques, including search heuristics, knowledge representation, automated planning and agent systems, machine learning, and probabilistic reasoning;
- Identify and apply AI techniques to a wide range of problems, including complex problem solving via search, knowledge-base systems, machine learning, probabilistic models, agent decision making, etc.;
- Design and implement appropriate AI solution techniques for such problems;
- Analyze and understand the computational trade-offs involved in applying different AI techniques and models.
- Communicate clearly and effectively using the technical language of the field correctly.

Unit-I	**Introduction and Local Search Algorithms and Optimization Problems**	**Classes: 12**
AI-Acting and Thinking humanly, rationally, Searching: Searching for solutions, Uniformed Search Strategies, Informed Search Strategies, Heuristic Functions. Hill-climbing, Simulated annealing, Local beam, Genetic algorithms, Constraint Satisfaction Problems, Backtracking Search for CSPs.		
Unit-II	**Adversial Search, Knowledge and Reasoning**	**Classes: 10**
Games, Optimal Decision in Games, Alpha-Beta Pruning, Evaluation Functions, Cutting off search, Games that include an Element of chance, Game programs. Knowledge and reasoning-I: Logical Agents Syntax and Semantics, Using First Order Logic, Knowledge Engineering, Inference in First-Order Logic: Propositional vs. First-Order Inference, Unification and Lifting, Resolution, Forward and Backward Chaining.		
Unit-III	**Planning and Learning**	**Classes: 11**
Classical planning problem, Language of planning problems, Expressiveness and extension, planning with state-space search, Partial-Order planning, Planning Graphs, Planning with Propositional Logic Forms of learning, Introduction learning, Learning Decision Tree, Statistical learning methods, learning with complete data, learning with hidden variables-EM Algorithms, Instance based learning, Neural networks.		
Unit-IV	**Expert Systems**	**Classes: 10**
Introduction, Advantages, Characteristics, General concepts, Applications and Domains, Languages, Shells and Tools, Elements, Production Systems, Procedural and Nonprocedural Paradigms, Artificial Neural Systems, Connectionist Expert Systems and Inductive Learning.		
Unit-V	**Design of Expert Systems**	**Classes: 08**
Selecting the Appropriate Problem, Stages in the Development of an Expert System, The Expert System Life Cycle. Detailed life cycle model, Expert system design examples-Certainty factors, Decision trees, backward chaining.		

Text Books:

1. Russell, Norvig-“Artificical Intelligence-A Modern Approach”,2e, 2004, PEA
2. Giarratano, Riley-“Expert Systems-Principles and Programming”, 3e,2003, Thomson

Reference Books:

1. George F Luger – “Artificial Intelligence-Structures and strategies for Complex problem Solving”, 4e, 2004, PEA.
2. Rich, Knight, Nair – “Artificial Intelligence”, 3e, TMH.

Web References:

1. https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_overview.htm
2. http://www.ggu.ac.in/download/ClassNote13/Artificial%20Intelligence%20and%20xpert%20 System 24.10.13.pdf
3. https://sumytsaju.files.wordpress.com/2016/05/cource-outline.pdf
4. nptel.ac.in/syllabus/syllabus_pdf/106106126.pdf

E-Text Books:

1. https://dcs.abu.edu.ng/staff/abdulrahimabdulrazaq/courses/cosc208/Artificial%20 Intelligence%20A% 20Modern%20Approach%20(3rd%20Edition).pdf

DISTRIBUTED SYSTEMS

IV- Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
17CF00408	Elective – 2	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		-	4	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		

Course Objectives:

- To learn the different Principles of Networking
- To learn about the Distributed Systems
- To understand concurrency control and transactions

Course Outcomes:

- Demonstrate knowledge of the basic elements and concepts related to distributed system technologies
- Demonstrate knowledge of the core architectural aspects of distributed systems
- Design and implement distributed applications
- Demonstrate knowledge of details the main underlying components of distributed systems (such as RPC, file systems)
- Uuse and apply important methods in distributed systems to support scalability and fault tolerance
- Demonstrate experience in building large-scale distributed applications.

Unit-I **Classes: 12**

Introduction, Networking and Internetworking – Types of network, Network principles, Internet protocols, Case studies.
Inter Process Communication- The API for Internet protocols, External data representation and marshalling, Client-Server Communication, Group Communication, IPC in UNIX.

Unit-II **Classes: 12**

Distributed Objects and Remote Invocation -Communication between distributed objects, Remote Procedure Call, Events and notifications, Case study – Java RMI
Operating System Support- The operating system layer, Protection, Process and threads, Communication and invocation, Operating system architecture.

Unit-III **Classes: 12**

Distributed File Systems – File service architecture, Sun Network File System, The Andrew File System.
Name Services - Name services and the Domain Name System, Directory services
Time and Global States and Coordination and Agreement -Introduction, Clocks Events and Process States, Synchronizing Physical Clocks, Logical Time and Logical Clocks, Global States, Distributed Debugging.

Unit-IV **Classes: 12**

Transactions and Concurrency Control: Introduction, Transactions, Nested Transactions, Locks, Optimistic Concurrency Control, Timestamp Ordering, Comparison of Methods for Concurrency Control.

Unit-V **Classes: 12**

Distributed Transactions - Flat and nested distributed transactions, Atomic commit protocols, Concurrency control in distributed transactions, distributed Deadlocks, Transactions with replicated data, Transaction recovery, Fault-tolerant services, Hierarchical and group masking of faults.
Distributed shared memory -Design and Implementation issues, Sequential consistency and ivy, Release consistency and Munin, Overview of Distributed Operating systems Mach, Chorus.

Text Books:

1. Distributed Systems Concepts and Design, G Coulouris, J Dollimore and T Kindberg, Fourth Edition, Pearson Education.

2. Distributed Operating Systems, Pradeep K.Sinha, PHI

Reference Books:

1. Advanced Concepts in Operating Systems, M Singhal, N G Shivarathri, Tata McGraw-Hill Edition.
2. Distributed Systems, S.Ghosh, Chapman &Hall/CRC, Taylor & Francis Group, 2010.
3. Distributed Systems – Principles and Paradigms, A.S. Tanenbaum and M.V. Steen, PearsonEducation.
4. Distributed Algorithms, N.A.Lynch, Elsevier.

Web References:

https://www.abebooks.com/first-edition/Advanced-Concepts-Operating-Systems-Mukesh-Singhal/5056007082/bd

E-Text Books:

https://www.scribd.com/doc/166936614/Advanced-Concepts-in-Operating-Systems

CLOUD COMPUTING

IV- Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF00409	Elective – 2	3	1	-	4	40	60	100

Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil	Total Classes: 60

Course Objectives:

- To explain the evolving computer model caned cloud computing.
- To introduce the various levels of services that can be achieved by cloud.
- To describe the security aspects in cloud.

Course Outcomes:

- Analyze the Cloud computing setup with it's vulnerabilities and applications using different architectures.
- Design different workflows according to requirements and apply map reduce programming model.
- Apply and design suitable Virtualization concept, Cloud Resource Management and design scheduling algorithms.
- Create combinatorial auctions for cloud resources and design scheduling algorithms for computing clouds

Unit-I	**Systems Modeling, Clustering and Virtualization**	**Classes: 10**

Distributed System Models and Enabling Technologies. Computer Clusters for Scalable Parallel Computing. Virtual Machines and Virtualization of Clusters and Data centers.

Unit-II	**Foundations**	**Classes: 10**

Introduction to Cloud Computing, Migrating into a Cloud, Enriching the ‘Integration as a Service’ Paradigm for the Cloud Era. The Enterprise Cloud Computing Paradigm

Unit-III	**Infrastructure as a Service (IAAS) & Platform and Software as a Service (PAAS / SAAS)**	**Classes: 10**

Virtual machines provisioning and Migration services, On the Management of Virtual machines for Cloud Infrastructures, Enhancing Cloud Computing Environments using a cluster as a Service. Secure Distributed Data Storage in Cloud Computing. Aneka, Comet Cloud, T-Systems’, Workflow Engine for Clouds. Understanding Scientific Applications for Cloud Environments.

Unit-IV	**Monitoring, Management and Applications**	**Classes: 08**

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

Unit-V	**Governance and Case Studies**	**Classes: 07**

Organisational Readiness and Change management in the Cloud age. Data Security in the Cloud, Legal issues in Cloud computing. Achieving Production Readiness for Cloud Services

Text Books:

- Cloud Computing: Principles and Paradigms by Rajkumar Bi.
- Distributed and Cloud Computing. Kal Hwang. Geoffeiy C.Fox. Jack J.Dongarra. E)sevier. 2012.

Reference Books:

- Cloud Computing: A Practical Approach. Anthony T.Velte. Toby J.VeFte, Robert Elsenpeter. Tata McGraw Hill. rp2Oll.
- Cloud Computing: Implementation, Management and Security, John W. Rittinouse, James F Ransome. CRC Press, rp2012.
- Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Ktriaraswamy, Shahed Latif, O'Redç SPD, rp2Oll.

Web References:

https://canvas.harvard.edu/courses/4077/assignments/syllabus
http://www.facweb.iitkgp.ernet.in/~shamik/spring2013/cc/cc2013_dtls.html

E-Text Books:

1. Mastering Cloud Computing, By Raj Kumar Buyya

OBJECT ORIENTED ANALYSIS AND DESIGN LAB

IV- Semester								
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF00410	Core	**_**	**_**	**3**	**2**	**40**	**60**	**100**

Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 42	Total Classes:42

Course Objectives:

- Design use cases and develop the use case model.
- Capture a business process model.
- Practice the object oriented analysis and design through UML on a particular application.
- Explore tools that support UML and object oriented software development.

Course Outcomes:

After completing this course the student must demonstrate the knowledgeand ability to:

- Show the importance of systems analysis and design in solving complex problems.
- Show how the object-oriented approach differs from the traditional approach to systems analysis and design.
- Construct various UML models (including use case diagrams, class diagrams, interaction diagrams, statechart diagrams, activity diagrams, and implementation diagrams) using the appropriate notation.
- Recognize the difference between various object relationships: inheritance, association, whole-part, and dependency relationships.
- Show the role and function of each UML model in developing object-oriented software.

List of Programs

Exp-1	**Software Requirements Specification**

Introduction to UML Diagrams. Create SRS for Recruitment System.

Exp-2	**Use Case Diagram**

a. Passport Automation System
b. Book bank management system
c. Online course reservation system
d. Foreign trading system
e. Conference Management System
f. BPO Management System

Exp-3	**Activity Diagram**

a. Passport Automation System
b. Book bank management system
c. Online course reservation system
d. Foreign trading system
e. Conference Management System
f. BPO Management System

Exp-4	**Domain Model**

Identity the conceptual classes and Develop a domain model with UML Class diagram for passport automation system

Exp-5	**Scenarios**

Using the identified scenarios find the interaction between objects and represent them using UML Interaction diagrams.

Exp-6	**State Chart Diagram**

Draw a state chart diagram for

a. Passport Automation System
b. Book bank management system
c. Online course reservation system

Exp-7 **State Chart Diagram**

a. Foreign trading system
b. Conference Management System
c. BPO Management System

Exp-8 **Architecture Diagram**

Identify the User Interface, Domain objects, and Technical services.

Exp-9 **Architecture Diagram**

Draw the partial layered, logical architecture diagram with UML package diagram notation

Exp-10 **Component Diagram**

Draw a Component diagram for

a. Passport Automation System
b. Book bank management system
c. Online course reservation system

Exp-11 **Component Diagram**

Draw a Component diagram for

a. Foreign trading system
b. Conference Management System
c. BPO Management System

Exp-12 **Deployment Diagrams**

Draw a Component diagram for

a. Passport Automation System
b. Book bank management system
c. Online course reservation system

Exp-13 **Deployment Diagrams**

Draw a Deployment diagram for

a. Foreign trading system
b. Conference Management System
c. BPO Management System

REFERENCE BOOKS

1. Simon Bennett, Steve Mc Robb and Ray Farmer, “Object Oriented Systems Analysis and Design Using UML”, Mc Graw Hill Education, 4th Edition, 2010
2. Pascal Roques, “Modeling Software Systems Using UML 2”, WILEY- Dreamtech India Pvt. Ltd, 2nd Edition, 2007.

WEB REFERENCES

1. https://www.tutorialspoint.com/uml/uml_overview.html
2. https://www.utdallas.edu/~chung/OOAD/M03_1_StructuralDiagrams.ppt
3. https://onedrive.live.com/download?cid=99CBBF765926367

WEB TECHNOLOGIES LAB

IV -Semester				
Course Code	Category	Hours/Week	Credits	Maximum Marks

17CF00411	Core	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		-	**-**	**3**	**2**	**40**	**60**	**100**
Contact Classes: Nil	Tutorial Classes: Nil		Practical Classes: 45			Total Classes:45		

Course Objectives:

- To create a fully functional website with mvc architecture.
- To develop an online Book store using we can sell books (Ex amazon .com).

Course Outcomes:

- define modern protocols and systems used on the Web (such as HTML, HTTP, URLs, CSS, XML)
- explain the functions of clients and servers on the Web, and describe the strengths and weaknesses of the client-server internet approaches to web design and implementation
- program, access, and manipulate data through the adoption of accepted standards, mark-up languages, client-side programming, and server-side programming
- design and implement an interactive web site(s) with regard to issues of usability, accessibility and internationalisation
- design and implement a client-server internet application that accommodates specific requirements and constraints, based on analysis, modelling or requirements specification
- justify and explain particular internet application concepts, relevant alternatives and decision recommendations, including design considerations for internet security

List of **Programs**

Exp -1	**Online Book Store Web Site**

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

1) HOME PAGE:

The static home page must contain three frames.

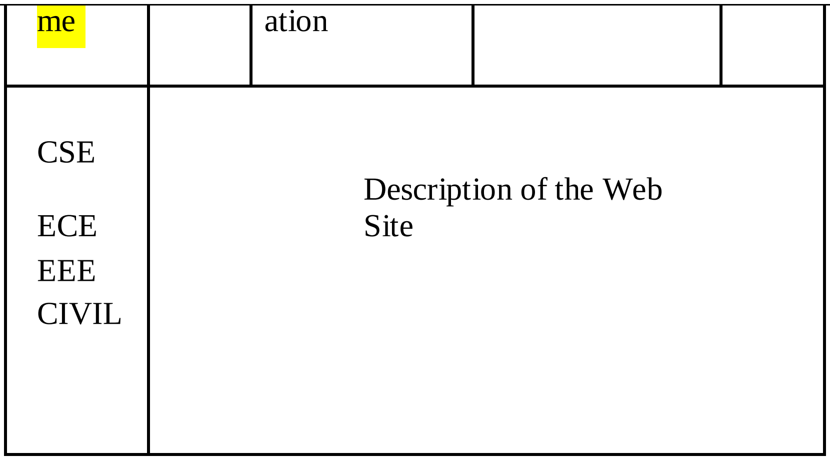
Top frame : Logo and the college name and links to Home page, Login page, Registration page, Catalogue page and Cart page (the description of these pages will be given below).

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

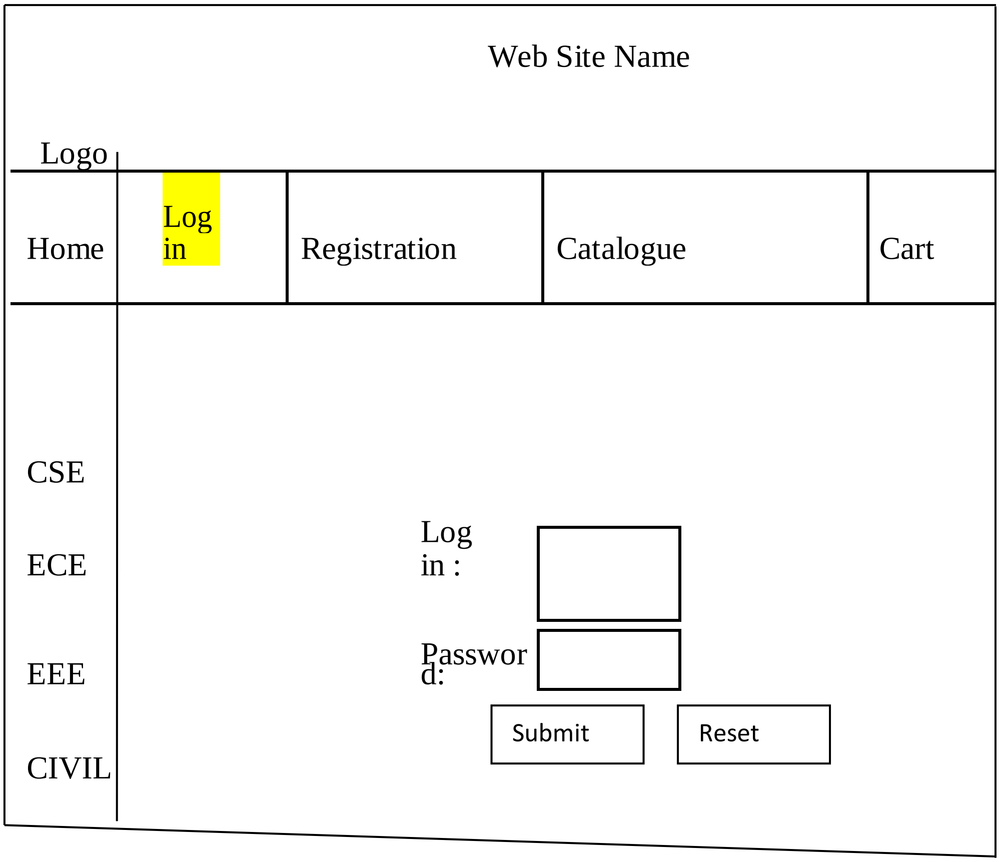
For e.g.: When you click the link “MCA” the catalogue for MCA Books should be displayed in the Right frame.

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

Logo	Web Site Name			
Ho	Login	Registr	Catalogue	Cart



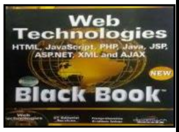
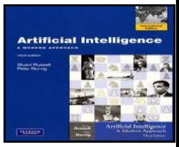
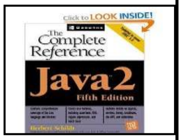
2) LOGIN PAGE: This page looks like below



3) CATOLOGUE PAGE:

The catalogue page should contain the details of all the books available in the web site in a table. The details should contain the following:

3. Snap shot of Cover Page.
4. Author Name.
5. Publisher.
6. Price.
7. Add to cart button.

Logo				
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL		Book : XML Bible Author : Winston Publication : Wiely	$ 40.5	Add to cart
		Book : AI Author : S.Russel Publication : Princeton Hall	$ 63	Add to cart
		Book : Java 2 Author : Watson Publication : BPB Publications	$ 35.5	Add to cart
		Book : HTML in 24 Hours Author : Sam Peter Publication : Sam Publication	$ 50	Add to cart

Exp - 2

4) CART PAGE:

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

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

5) REGISTRATION PAGE:

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

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

Exp - 3	

VALIDATION:
Write JavaScript to validate the following fields of the above registration page.
1. Name (Name should contains alphabets and the length should not be less than 6 characters).
2. Password (Password should not be less than 6 characters length).
3. E-mail id (should not contain any invalid and must follow the standard pattern name@domain.com)
4. Phone number (Phone number should contain 10 digits only).

Note : validation of the login page can also be done with these parameters

Exp - 4	**CSS layering**

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

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

```
<HTML>
<HEAD>
<style type="text/css">
B.headline {color:red; font-size:22px; font-family:arial; textdecoration:underline}
</style>
</HEAD>
<BODY>
<b>This is normal bold</b><br>
```

Selector {cursor:value}
For example:

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

```
</b>
</body>
</html>
<b class="headline">This is headline style bold</b>
</BODY>
</HTML>
```

2) Set a background image for both the page and single elements on the page. You can define the background image for the page like this:

```
                         BODY {background-image:url(myimage.gif);}
```

3) Control the repetition of the image with the background-repeat property. As background-repeat: repeat Tiles the image until the entire page is filled, just like an ordinary background image in plain HTML.

4) Define styles for links as

```
A:link
A:visited
A:active
A:hover
```

Example:

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

5) Work with layers:

For example:

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

6) Add a customized cursor:

```
                 Selector {cursor:value}
```

For example:

```
           <html>

           <head>

                <style type="text/css">

               .xlink {cursor:crosshair}

                .hlink{cursor:help}
```

```
        </style>

     </head>

     <body>

        <b>

          <a href="mypage.htm" class="xlink">CROSS LINK</a>

           <br>

          <a href="mypage.htm" class="hlink">HELP LINK</a>

        </b>

     </body>

     </html>
```

Exp - 5	**XML file to display the Book Information**

Write an XML file which will display the Book information which includes the following:
1) Title of the book
2) Author Name
3) ISBN number
4) Publisher name
5) Edition
6) Price
Write a Document Type Definition (DTD) to validate the above XML file.
Display the XML file as follows.
The contents should be displayed in a table. The header of the table should be in color GREY. And the Author names column should be displayed in one color and should be capitalized and in bold.
Use your own colors for remaining columns.
Use XML schemas XSL and CSS for the above purpose.
Note: Give at least for 4 books. It should be valid syntactically.
Hint: You can use some xml editors like XML-spy

Exp- 6	**Install TOMCAT web server**

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

Exp - 7	**User Authentication**

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

1. Create a Cookie and add these four user id‟s and passwords to this Cookie.

2. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies.
If he is a valid user(i.e., user-name and password match) you should welcome him by name(user-name) else you should display " You are not an authenticated user ". Use init-parameters to do this. Store the user-names and passwords in the webinf.xml and access them in the servlet by using the getInitParameters() method.

Exp - 8 **Install a Database**

Install a database (Mysql or Oracle).
Create a table which should contain at least the following fields: name, password, emailid, phone number(these should hold the data from the registration form). Practice 'JDBC' connectivity.
Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them. Experiment with various SQL queries. Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page (week2).

Exp - 9 **Authenticate the User**

Write a JSP which does the following job:
Insert the details of the 3 or 4 users who register with the web site (week9) by using registration form. uthenticate the user when he submits the login form using the user name and password from the database (similar to week8 instead of cookies).

Exp - 10 **Catalogue Page to Display Details of Items**

Create tables in the database which contain the details of items (books in our case like Book name, Price, Quantity, Amount)) of each category. Modify your catalogue page (week 2)in such a way that you should connect to the database and extract data from the tables and display them in the catalogue page using JDBC.

Exp -11 **Session Tracking**

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

Reference Books

1. Java Server Programming for Professionals, 2nd Edition, Bayross and others, O‟reilly,SPD, 2007.
2. JDBC, Servlets, and JSP ,Black Book, K. Santosh Kumar, dreamtech.
3. Core Web Programming, 2nd Edition, Volume 1, M.Hall and L.Brown, PHPTR.
4. Core Web Programming, 2nd Edition, Volume 2, M.Hall and L.Brown, PHPTR.
5. Core Java, Volume 1, Horstman and Cornell, 8th Edition, Pearson Education, 2008.
6. Core Java, Volume 2, Horstman and Cornell, 8th Edition, Pearson Education, 2008.

7. Java Programming: Advanced Topics, 3rd Edition, J.Wiggles worth and P.McMillan,Thomson, 2007.
8. Struts 2 for Beginners,S.Shah & V.Shah,The X Team,SPD,2nd edition.

Web References

http://cse.gecgudlavalleru.ac.in/pdf/manuals/Advanced-Java-and-Web-Technologies-LabManual.pdf

file:///C:/Users/admin/Downloads/32359074.pdf

DATA WAREHOUSE AND DATA MINING LAB

IV- Semester

Course Code	Catagory	Hours/Week			Credits	Maximum Marks		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF00412	Core	**-**	**-**	**3**	**2**	**40**	**60**	**100**

Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 42	Total Classes:42

Course Objectives:

- Understand the need of Data Warehouses over Databases, and the difference between usage of operational and historical data repositories.
- Able to differentiate between RDBMS schemas & Data Warehouse Schemas.
- Get a clear idea of various classes of Data Mining techniques, their need, scenarios (situations) and scope of their applicability.
- Implement association rule for mining and also implement the clustering technique.

Course Outcomes:

The candidate will get knowledge of:

- Data preprocessing and data quality.
- Modeling and design of data warehouses.
- Algorithms for data mining.

List of Programs

Exp-1

List all the categorical (or nominal) attributes and the real-valued attributes separately

Exp-2

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

Exp-3

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

Exp-4

Credit risk assessment

Exp-5

Is testing on the training set as you did above a good idea? Why or Why not

Exp-6

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

Exp-7

Check to see if the data shows a bias against "foreign workers" (attribute 20),or "personal-status" (attribute 9). One way to do this (perhaps rather simple minded) is to remove these attributes from the dataset

Exp-8

Another question might be, do you really need to input so many attributes to get good results? Maybe only a few would do

Exp-9

Sometimes, the cost of rejecting an applicant who actually has a good credit (case 1) might be higher than accepting an applicant who has bad credit (case 2). Instead of counting the misclassifications equally in both cases, give a higher cost to the first case (say cost 5) and lower cost to the second case. You can do this by using a cost matrix in Weka

Exp-10

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

Exp-11

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

Text Books: An Introduction to the WEKA Data Mining System By Zdravko Markov

REFERENCE BOOKS

1. J. Han, M. Kamber, “Data Mining: Concept and Techniques”, Academic Press, Morgan Kanfman Publishers, 3rd Edition, 2008.
2. Alex Berson, Stephen J. Smith, “Data Warehousing, Data Mining & OLAP”, Tata McGraw Hill, 10th Edition, 2007.
3. Pieter Adrians, DolfZantinge, “Data Mining”, Addison Wesley, Peter V, 2000.

WEB REFERENCES

1) http://weka.wikispaces.com/Troubleshooting

2) http://www.cs.waikato.ac.nz/ml/weka/index_documentation.html

FUNDAMENTALS OF DATA SCIENCE

V – Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIA	SEE	Total
17CF00501	Core	3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total classes:60		

Course Objectives:

- To understand the business data analysis through the powerful tools of data application.
- Learn how to apply Table and get introduced in to R .
- Understand the methods of data mining and creation of decision tree.

Course Outcomes:

- Understand business intelligence and business and data analytics.
- To understand the business data analysis through the powerful tools of data application.
- Understand the methods of data mining.
- Apply basic tools (plots, graphs, summary statistics) to carry out EDA.
- Understand the key elements of a data science project
- Identify the appropriate data science technique and/or algorithm to use for the major data science tasks.

Unit-I **Classes: 12**

Introduction, What Is Statistical Learning?, Why Estimate f?, How Do We Estimate f?, The Trade-Off Between Prediction Accuracy and Model Interpretability, Supervised Versus Unsupervised Learning, Regression Versus Classification Problems, Assessing Model Accuracy, Measuring the Quality of Fit, The Bias -Variance Trade-off, The Classification Setting, Introduction to R, Basic Commands, Graphics, Indexing Data, Loading Data, Additional Graphical and Numerical Summaries.

Unit-II **Classes: 12**

Linear Regression, Simple Linear Regression, Multiple Linear Regression, Other Considerations in the Regression Model, Comparison of Linear Regression with K-Nearest Neighbours.

Unit-III **Classes: 12**

Classification, Logistic Regression, Linear Discriminant Analysis, A Comparison of Classification Methods, Logistic Regression, LDA, QDA, and KNN.

Unit-IV **Classes: 12**

Programming for basic computational methods such as Eigen values and Eigen vectors, sparse matrices, QR and SVD, Interpolation by divided differences.

Data Wrangling: Data Acquisition, Data Formats, Imputation, The split-apply-combine paradigm

Unit-V **Classes: 12**

Data Objects and Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity.

Data Warehouse: Basic Concepts, Data Warehouse Modeling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Data Generalization by Attribute-Oriented Induction.

Text Books:

- Gareth James Daniela Witten Trevor Hastie, Robert Tibshirani, An Introduction to Statistical Learning with Applications in R, February 11, 2013, web link: www.statlearning.com.
- Mark Gardener, Beginning R The statistical Programming Language, Wiley, 2015.
- Han , Kamber, and J Pei, Data Mining Concepts and Techniques, 3rd edition, Morgan Kaufman, 2012.

Reference Books:

- Sinan Ozdemir, Principles of Data Science, Packt Publishing Ltd Dec 2016.
- Joel Grus, Data Science from Scratch, Oreilly media, 2015.
- Trevor Hastie, Robert Tibshirani and Jerome Friedman. Elements of Statistical Learning, Second Edition. ISBN 0387952845. 2009. (free online)
- Mohammed J. Zaki and Wagner Miera Jr. Data Mining and Analysis: Fundamental Concepts and Algorithms. Cambridge University Press. 2014.

Web References:

- http://www.cs.tau.ac.il/~apartzin/ds2015/DS_CourseIntro.pdf
- https://storage.ischool.syr.edu/ischool.syr.edu/oldmedia/documents/2012/3/DataScienceBook1_1.pdf

E-Text Books:

- https://www.cs.upc.edu/~robert/teaching/estadistica/rprogramming.pdf
- https://www.analyticsvidhya.com/blog/2016/02/complete-tutorial-learn-data-science-scratch/
- http://www.stat.wmich.edu/s160/hcopy/book.pdf

MOBILE APPLICATION DEVELOPMENT

V – Semester								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
17CF00502	Core	3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total classes:60		

Course Objectives:

- To understand fundamentals of android operating systems.
- Illustrate the various components, layouts and views in creating android applications
- To understand fundamentals of android programming.

Course Outcomes:

- Design components, systems and/or processes to meet required specifications
- Synthesize alternative/innovative solutions, concepts and procedures
- Apply decision-making methodologies to evaluate solutions for efficiency, effectiveness and sustainability
- Implement and test solutions
- Demonstrate research skills

Unit-I	Basics of Mobile Applications Development:	Classes: 12

Tools: Eclipse ADT, Android Studio. Understanding the Role of Android Application Components, Understanding the Utility of Android API, Overview of the Android Project Files, Understanding Activities, Role of the Android Manifest File, Creating the User Interface Commonly Used Layouts and Controls, Event Handling, Displaying Messages Through Toast, Creating and Starting an Activity, Using the Edit Text Control, Choosing Options with Checkbox, Choosing Mutually Exclusive Items Using Radio Buttons

Unit-II	Building Blocks for Android Application Design:	Classes: 12

Introduction to Layouts, Linear Layout, Relative Layout, Absolute Layout, Using Image View, Frame Layout, Table Layout, Grid Layout, Adapting to Screen orientation.

Utilizing Resources and Media Resources, Creating Values Resources, Using Drawable Resources, Switching States with Toggle Buttons, Creating an Images Switcher Application, Scrolling Through Scroll View, playing Audio, Playing Video, Displaying Progress with Progress Bar, Using Assets

Unit-III	Using Selection widgets and Debugging:	Classes: 12

Using List View, Using the Spinner control, Using the GridView Control, Creating an Image Gallery Using the View Pager Control, Using the Debugging Tool: Dalvik Debug Monitor Service(DDMS), Debugging Application, Using the Debug Perspective.

Unit-IV	Building Menus:	Classes: 12

Creating Interface Menus and Action Bars, Menus and Their Types, Creating Menus Through XML, Creating Menus Through Coding, Applying a Context Menu to a List View, Using the Action Bar, Replacing a Menu with the Action Bar, Creating a Tabbed Action Bar, Creating a Drop-Down List

Action Bar

Unit-V	**Storing Data & Communicating with SMS and Emails:**	**Classes: 12**

Using the SQLiteOpenHelperclasss, Accessing Databases with the ADB, Creating a Data Entry Form. Understanding Broadcast Receivers, Using the Notification System, Sending SMS Messages with Java Code, Receiving SMS Messages, Sending Email, Working With Telephony Manager.

Text Books:

- Android Programming by B.M Harwani, Pearson Education, 2013.

Reference Books:

- Android application Development for Java Programmers, James C Sheusi, Cengage Learning
- Android In Action by w.Frank Ableson, Robi Sen, Chris King, C. Enrique Ortiz., Dreamtech.
- Professional Android 4 applications development, Reto Meier, Wiley India, 2012.
- Beginning Android 4 applications development, Wei- Meng Lee, Wiley India,2013
- PawPrints Learning Technologies, Beginning Android Development: Create Your Own Android Apps Today, 2014.
- Erik Hellman, Android Programming: Pushing the Limits, John Wiley and sons ltd, 2014.
- Neil Smyth, Android Studio Development Essentials.
- Joseph Annuzzi,Jr, Lauren Darcey, Introduction to Android Application Development, Addison-Wesley, Fourth Edition.

Web References:

- https://www.tutorialspoint.com/mobile_development_tutorials.htm
- https://www.cs.cmu.edu/~bam/uicourse/830spring09/BFeiginMobileApplicationDevelopment.pdf
- https://www.theserverside.com/tutorial/Mobile-application-development-tutorial

E-Text Books:

- http://read.pudn.com/downloads107/ebook/443059/[J2ME%E5%BC%80%E5%8F%91%E5%A4% A7%E5%85%A8].Java_J2ME_TheCompleteReference_McGrawHill_Osborne.pdf
- http://www.freebookcentre.net/mobile-technology/Mobile-Application-Development-Books.html
- https://www.amazon.com/slp/mobile-application-development-books/uvjx2gdxt3hv4cj

SCRIPTING LANGUAGES

V – Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIA	SEE	Total
17CF00503	Elective	3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total classes:60		

Course Objectives:

The course should enable the students to:

- The goal of the course is the study of scripting languages such as Python, PHP.
- To learn how to use lists, tuples, and dictionaries in Python programs.
- To learn how to design and program Python applications.
- To utilize high-performance programming constructs available in Python to develop solutions in real life scenarios.

Course Outcomes:

Upon successful completion of this subject, students should:

- Master an understanding of scripting and the contributions of scripting languages.
- Know the basic fundamentals of AngularJS.
- Understand the PHP concepts.
- Master an understanding of the built-in objects of Python.
- Apply the best features of Python to program real life problems
- Master an understanding of Python especially the object-oriented concepts
- Implement database applications.

Unit-I	**Introduction to Scripting Languages**	**Classes: 12**

Introduction, Differentiate between scripting languages and programming languages, Types of Scripting languages, characteristics of Scripting Languages, Advantages and Disadvantages of scripting languages.

Basic fundamentals of AngularJS: Overview of AngularJS, Features of AngularJS, Advantages and Disadvantages of AngularJS, MVC architecture of AngularJS. AngularJS expressions, AngularJS Modules, AngularJS Directives.

Unit-II	**PHP Concepts**	**Classes: 12**

Basics of regular expressions**,** Pattern matching, Replacing text, Splitting strings using Regular Expressions,System Variable(GET, POST, Cookies and session),Working with forms.

Introduction to OOPS: Class, Objects, Declaring a class, The new keyword and constructor, Destructor, Access method and properties, using $this variable, Public ,private, protected properties and methods, Static properties and method, Class constant, Inheritance & code reusability, Polymorphism, Parent:: & self:: keyword ,Instance of operator, Abstract method and class ,Interface, Final.

Unit-III	**Python Programming Concepts**	**Classes: 12**

Python Scripts: Introduction to Python language, Control statements, String operations.
Lists-Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences. Comprehensions.
Python Modules and Packages, Different ways to import Packages, File handling mechanisms.

Unit-IV	**Advanced Python Programming_1**	**Classes: 12**

OOPs concepts -Class, Objects, Types of variables, Types of methods, Constructor, Destructor, Garbage collection, Inner classes, Duck type philosophy of python, Overloading, Overriding, Inheritance.

Exception: Difference between an error and Exception, Exception, Handling Exception, example programs, try-finally, try except block, Raising an Exception, User-defined exceptions.

Unit-V	**Advanced Python Programming_2**	**Classes: 12**

Regular Expressions in python, Threads, Threads creation, multi threading, Daemon threads, Thread synchronization, Proper use of synchronization primitives (locks, semaphores, events, condition variables, etc.) Python database connectivity.

Text Books:

- Python Programming: A Modern Approach, Vamsi Kurama, Pearson
- Learning Python, Mark Lutz, Orielly
- Developing Web Applications in PHP and AJAX, Harwani, McGraw Hill

Reference Books:

- Think Python, Allen Downey, Green Tea Press
- Core Python Programming, W.Chun, Pearson.
- Introduction to Python, Kenneth A. Lambert, Cengage

Web References:

- www.tutorialspoint.com
- www.geeksforgeeks.com
- www.w3school.com

E-book Reference:

- https://www.digitalocean.com/community/tutorials/digitalocean-ebook-how-to-code-in-python
- https://tutorialzine.com/2018/03/8-awesome-and-free-php-books
- http://www.angularjsbook.com/

SOFTWARE PROJECT MANAGEMENT

V – Semester								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
17CF00504	Elective	3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total classes:60		

Course Objectives:

- The course should enable the students to:
- Provide an understanding of the purpose, methods and benefits of process management by exposing the student to the concepts, practices, processes, tools and techniques used in process management for software development.
- Perform critical analyses and write feasibility studies of system implementations
- Use oral and electronic communication effectively
- Work cooperatively in teams and with others
- Recognize the need for continuing professional development

Course Outcomes:

- Apply key PM concepts
- Explain the project life cycle (concept, definition, execution, finish).
- Apply estimating and risk management techniques to IS projects.

Unit-I	Conventional Software Project Management and Improving Software Economics	Classes: 12

The Waterfall Model, Conventional software Management Performance. Evolution of Software Economics: Software Economics, Pragmatic Software Cost Estimation, Reducing Software Product Size, Improving software Processes, Improving Team Effectiveness, Improving Automation, Achieving Required Quality, Peer Inspections

Unit-II	Conventional and Modern Software Management	Classes: 12

Principles of Conventional Software Engineering, Principles of Modern Software Management, Transitioning to an Iterative Process. Life Cycle Phases: Engineering and Production Stages, Inception. Elaboration, Construction, Transition Phases.

Unit-III	Artifacts of the Process and Flows of the Process	Classes: 12

The Artifact Sets. Management Artifacts, Engineering Artifacts, Programmatic Artifacts. Model BasedSoftware Architectures: A Management Perspective and Technical Perspective.

Software Process Workflows. Inter Trans Workflows. Checkpoints of the Process: Major Mile Stones,Minor Milestones, Periodic Status Assessments. Interactive Process Planning: Work Breakdown Structures, Planning Guidelines, Cost and Schedule Estimating. Interaction Planning Process. Pragmatic Planning.

Unit-IV	Project Organizations and Responsibilities and Project Control and Process Instrumentation	Classes: 12

Line-of-Business Organizations, Project Organizations and Evolution of Organizations. Process Automation: Automation Building Blocks, The Project Environment, Server Care Metrics, Management Indicators, Quality Indicators, Life Cycle Expectations, Pragmatic Software Metrics, Metrics Automation. Tailoring the process: Process Discriminates, Example

Unit-V	**Future Software Project Management**	**Classes: 12**

Modern Project Profiles Next Generation Software economics, Modern Process Transitions. Case Study: The Command Center Processing and Display System –Replacement (CCPDSR)

Text Books:

- Walker Royce, “Software Project Management”, 1998, PEA.

Reference Books:

- Henrey, “Software Project Management” Pearson.
- Richard H. Thayer: “Software Engineering‟ Project Management”, 1997, IEEE Computer Society.
- Shere K. D. “ Software Engineering and Management”, 1998, PHI.
- S. A. Kelkar, “ Software Project Management: A Concise Study”, PHI.
- Hughes Cotterell, “ Software Project Management”, 2e, TMH. Kaeron Conway, “ Software Project Management from Concept to Development”, Dream Tech.

Web References:

- https://www.slideshare.net/sheetal_singh/software-project-management-by-walker-royce
- https://www.slideshare.net/jhonrehmat/introduction-of-software-project-management

E-Text Books:

- http://www.kvimis.co.in/sites/kvimis.co.in/files/ebook_attachments/Walker%20Royce%20Software %20Project%20Management.pdf

MACHINE LEARNING

V – Semester

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
17CF00505	Elective	3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total classes:60		

Course Objectives:

The course should enable the students to:

- Understanding nature of problems solved with MachineLearning.
- Understand the Shared address space platforms.

Course Outcomes:

- Ability to select and implement machine learning techniques and computing environment that are suitable for the applications under consideration.
- Ability to understand and apply scaling up machine learning techniques and associated computing techniques and technologies
- Ability to recognize and implement various ways of selecting suitable model parameters for different machine learning techniques.
- Ability to integrate machine learning libraries and mathematical and statistical tools with modern technologies like hadoop and mapreduce.

Unit-I	**INTRODUCTION AND MACHINE LEARNING**	**Classes: 12**

Introduction: Why Machine learning, Examples of Machine Learning Problems, Structure of Learning, Learning versus Designing, Training versus Testing, Characteristics of Machine learning tasks, Predictive and descriptive tasks, Machine learning Models: Geometric Models, Logical Models, Probabilistic Models. Features: Feature types, Feature Construction and Transformation, Feature Selection.

Unit-II	**CLASSIFICATION & REGRESSION**	**Classes: 12**

Classification: Binary Classification- Assessing Classification performance, Class probability Estimation-Assessing class probability Estimates, Multiclass Classification.

Regression: Assessing performance of Regression- Error measures, Overfitting- Catalysts for Overfitting, Case study of PolynomialRegression.

Unit-III	**LINEAR MODELS**	**Classes: 12**

Least Squares method, Multivariate Linear Regression, Regularized Regression, Using Least Square regression for Classification. Perceptron, Support Vector Machines, Soft Margin SVM, Obtaining probabilities from Linear classifiers, Kernel methods for non-Linearity.

Unit-IV	**LOGIC BASED AND ALGEBRIC MODELS**	**Classes: 12**

Distance Based Models: Neighbours and Examples, Nearest Neighbours Classification, Distance based clustering-K means Algorithm, Hierarchical clustering,

Rule Based Models: Rule learning for subgroup discovery, Association rule mining.

Tree Based Models: Decision Trees, Ranking and Probability estimation Trees, Regression trees, Clustering Trees.

Unit-V	**PROBABLISITIC MODELS**	**Classes: 12**

Normal Distribution and Its Geometric Interpretations, Naïve Bayes Classifier, Discriminative learning with Maximum likelihood, Probabilistic Models with Hidden variables: Estimation-Maximization Methods, Gaussian Mixtures, and Compression based Models.

Text Books:

1.PeterFlach:MachineLearning:TheArtandScienceofAlgorithmsthatMakeSenseofData, Cambridge University Press, Edition 2012.

2.Hastie, Tibshirani, Friedman: Introduction to Statistical Machine Learning with Applications in R, Springer, 2ndEdition-2012.

Reference Books:

- C.M. Bishop:Pattern Recogintion and Machine Learning, Springer 1st Edition-2013
- Ehem Alpaydin: Introduction to Machine Learning,PHI 2nd Edition-2013
- Parag Kulkami: Reinforcement and systematic Machine Learning for Decision Making,Wiley-IEEE Press,Edition July 2012.

Web References:

https:// medium.com/machine-learning

E-Text Books:

- https://www.kdnuggets.com/2016/10/5-free-ebooks-machine-learning

INTERNET OF THINGS

V – Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIA	SEE	Total
17CF00506	Elective	3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total classes:60		

Course Objectives:

The course should enable the students to:

- Understand the architecture of Internet of Things and connected world.
- Explore on use of various hardware and sensing technologies to build IoT applications.
- Illustrate the real time IoT applications to make smart world.
- Understand the available cloud services and communication API's for developing smart cities

Course Outcomes:

Upon successful completion of this subject, students should:

- Able explain and demonstrate various components of Internet of Things (IoT);
- Able to analyze the role and importance of IoT in the modern world;
- Able to investigate and propose various requirements of IoT for real world applications;
- evaluate a variety of existing and developing architecture technologies for IoT;
- Describe and evaluate different applications of the IoT.

Unit-I	**Introduction to Internet of Things (IoT)**	**Classes: 12**

Definition and characteristics of IoT, physical design of IoT, logical design of IoT, IoT enabling technologies, IoT levels and deployment, domain specific IoTs.

Unit-II	**IoT and M2M**	**Classes: 12**

Introduction, M2M, difference between IoT and M2M, software defined networking (SDN) and network function virtualization (NFV) for IoT, basics of IoT system management with NETCONF-YANG.

Unit-III	**IOT Architecture and Python**	**Classes: 12**

IoT Architecture: State of the art introduction, state of the art; Architecture reference model: Introduction, reference model and architecture, IoT reference model.

Logical design using Python: Installing Python, Python data types and data structures, control flow, functions, modules, packages, file handling.

Unit-IV	**IoT Physical Devices and Endpoints**	**Classes: 12**

Introduction to Raspberry Pi interfaces (Serial, SPI, I2C), programming Raspberry PI with Python, other IoT devices.

Unit-V	**IoT Physical Servers and Cloud Offerings**	**Classes: 12**

Introduction to cloud storage models and communication APIs; WAMP: Auto Bahn for IoT, Xively cloud for IoT; case studies illustrating IoT design: Home Automation, smart environment

Text Books:

- Arshdeep Bahga, Vijay Madisetti, “Internet of Things: A Hands-on-Approach”, VPT, 1st Edition, 2014.
- Matt Richardson, Shawn Wallace, “Getting Started with Raspberry Pi”, O’Reilly (SPD), 3rd Edition,2014.

Reference Books:

- Adrian McEwen, Hakim Cassimally, “Designing the Internet of Things”, John Wiley and Sons, 1st Edition, 2014.
- Francis Da Costa, **“Rethinking the Internet of Things: A Scalable Approach to ConnectingEverything”,** Apress Publications, 1stEdition, 2013.

Web References:

- https://www.upf.edu/pra/en/3376/22580.
- https://www.coursera.org/learn/iot.
- https://bcourses.berkeley.edu.
- www.innovianstechnologies.com

E-Text Books:

- https://mitpress.mit.edu/books/internet-things
- http://www.apress.com

WEB SERVICES

V – Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIA	SEE	Total
17CF00507	Elective	3	1	-	4	40	60	100
Contact classes:45	Tutorial classes:15	Practical Classes: Nil				Total classes:60		

Course Objectives:

The course should enable the students to:

- Apply tools to retrieve the information from the database.
- Understand a well formed SOAP schemas for developing web services using SOAP
- Design and implement web services from the server and client side

Course Outcomes:

- Understand the use of web services in B2C and B2B applications.
- Understand the design principles and application of SOAP and REST based web services.
- Design collaborating web services according to a specification.
- Implement an application that uses multiple web services in a realistic business scenario.
- Use industry standard open source tools such as Apache Axis2, Tomcat, Derby and

Unit-I		**Classes: 12**

Evolution of distributed computing, Core distributed computing technologies, client/server, CORBA, JAVA RMI, Micro Soft DCOM, MOM, Challenges in Distributed Computing, role of J2EE and XML in distributed computing, emergence of Web Services and Service Oriented

Unit-II		**Classes: 12**

Introduction to Web Services- The definition of web services, basic operational model of web services,tools and technologies enabling web services, benefits and challenges of using web services.

Web Services Architecture- Web services Architecture and its characteristics, core building blocks of webservices, standards and technologies available for implementing web services, web

Unit-III		**Classes: 12**

SOAP Message Structure, SOAP Encoding , SOAP message exchange models, SOAP communication and messaging, SOAP security.

Developing Web Services using SOAP- Building SOAP Web Services, developing SOAP Web Servicesusing Java, limitations of SOAP.

Unit-IV		**Classes: 12**

Describing Web Services -WSDL, WSDL in the world of Web Services, Web Services life cycle,anatomy of WSDL definition document, WSDL bindings, WSDL Tools, limitations of WSDL. **Discovering Web Services-** Service discovery, role of service discovery in a SOA, service discoverymechanisms, UDDI: UDDI Registries, uses of UDDI Registry, Programming with UDDI, UDDI data structures, support for categorization in UDDI Registries, Publishing API, Publishing information to a UDDI Registry, searching information in a UDDI Registry, deleting information in a UDDI Registry, limitations of UDDI

Unit-V		Classes: 12

Web Services Interoperability: Means of ensuring Interoperability, Overview of .NET and J2EE. Web Services Security: XML security frame work, XML encryption, XML digital signature, XKMS structure, guidelines for signing XML documents

Text Books:

- Developing Java Web Services, R. Nagappan, R. Skoczylas, R.P.Sriganesh, Wiley India, rp –
  2008.
- Developing Enterprise Web Services, S. Chatterjee, J. Webber, Pearson Education, 2008.
- XML, Web Services, and the Data Revolution, F.P.Coyle, Pearson Education.

Reference Books:

- Building Web Services with Java, Second Edition, S. Graham and others, Pearson Edn., 2008.
- Java Web Services, D.A. Chappell and T. Jewell, O‘Reilly, SPD.
- Java Web Services Architecture, McGovern, et al., Morgan Kaufmann Publishers, 2005.
- J2EE Web Services, Richard Monson-Haefel, Pearson Education.

Web References:

- https://en.wikipedia.org/wiki/Web_service
- http://www.service-architecture.com/articles/web-services/web_services_explained.html

E-Text Books:

- http://bookboon.com/en/introduction-to-web-services-with-java-ebook
- https://ebooks-it.org/category-web-services-1.htm

CYBER SECURITY

V – Semester

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
17CF00507	Elective	3	1	-	4	40	60	100
Contact classes:45	Tutorial classes:15	Practical Classes: Nil				Total classes:60		

Objectives:

The course should enable the students to:

- Understand the core information assurance principles in n-tier web applications.
- Identify the key components of cyber security network architecture.
- Study on digital certificates, signatures and digital forensics for cyber crime investigation.

Unit-I **INTRODUCTION** **Classes: 12**

A web security forensic lesson, web languages, introduction to different web attacks, overview of n-tier web applications; Web servers: Apache, IIS, database servers.

Unit-II **REVIEW OF COMPUTER SECURITY AND CYBER CRIMES ISSUES** **Classes: 12**

Public key cryptography, RSA, online shopping, payment gateways, unauthorized access to computers, computer intrusions, white collar crimes, viruses and malicious code, internet hacking and cracking, virus attacks, pornography, software piracy, intellectual property, mail bombs, exploitation, stalking and obscenity in internet, digital laws and legislation, law enforcement roles

Unit-III **WEB HACKING BASICS AND INVESTIGATION** **Classes: 12**

Web hacking basics HTTP and HTTPS URL, web under the cover overview of java security reading the HTML source, applet security, servlets security, symmetric and asymmetric encryptions, network security basics, firewalls and IDS.Investigation: Introduction to cyber-crime investigation, investigation tools, e-discovery, digital evidence collection, evidence preservation.

Unit-IV **DIGITAL CERTIFICATES AND DIGITAL FORENSICS** **Classes: 12**

Digital certificates, hashing, message digest, and digital signatures; Digital forensics: Introduction to digital forensics, forensic software and hardware, analysis and advanced tools, forensic technology and practices.

Unit-V **SECURING DATABASES, LAWS AND ACTS** **Classes: 12**

Basics, secure JDBC, securing large applications, cyber graffiti; Laws and acts: Laws and ethics, digital evidence controls, evidence handling procedures; Basics of Indian Evidence Act IPC and CRPC: Electronic communication privacy act, legal policies.

Text Books:

- Bill Nelson, Amelia Phillips, Frank Enfinger, Christopher Steuart, “Guide to Computer Forensics and Investigations”, Information Security Professionals, 4th Edition, 2009.
- Stuart McClure, Saumil Shah, Shreeraj Shah, “Web Hacking: Attacks and Defense”, Addison-Wesley Professional, 1st Edition, 2002.

Reference Books:

- Kevin Mandia, Chris Prosise, Matt Pepe, “Incident Response and Computer Forensics “, Tata Mc Graw Hill, 1st Edition, 2006.
- Garms, Jess, Daniel Somerfield, “Professional Java Security”, Wrox Press, Illustrated Edition, 2001.
- Robert M Slade, “Software Forensics”, Tata Mc Graw Hill, New Delhi, 1st Edition, 2005

Web References:

- http://www.mail.nih.gov/user/faq/tlsssl.htm
- http://www.openssl.org/
- http://www.ntsecurity.net/

E-Text Books:

- https://www.mitre.org/sites/.../pr-13-1028-mitre-10-strategies-cyber-ops-center.pdf
- https://www.coursera.org/specializations/cyber-security
- https://www.ccdcoe.org/publications/books/NationalCyberSecurityFrameworkManual.pdf

SOFTWARE TESTING

V – Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	T	P	C	CIA	SEE	Total
17CF00502	Elective	3	1	-	4	40	60	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total classes:60		

Course Objectives:

- Fundamentals for various testing methodologies.
- Describe the principles and procedures for designing test cases.
- Provide supports to debugging methods.
- Acts as the reference for software testing techniques and strategies.

Course Outcomes:

- Understand the basic testing procedures.
- Able to support in generating test cases and test suites.
- Able to test the applications manually by applying different testing methods and
- Automation tools.
- Apply tools to resolve the problems in Real time environment.

Unit-I		**Classes: 12**

Introduction: Purpose of Testing, Dichotomies, Model for Testing, Consequences of Bugs, Taxonomy of Bugs. Flow graphs and Path testing: Basics Concepts of Path Testing, Predicates, Path Predicates and Achievable Paths, Path Sensitizing, Path Instrumentation, Application of Path Testing.

Unit-II		**Classes: 12**

Transaction Flow Testing: Transaction Flows, Transaction Flow Testing Techniques. Dataflow testing:
Basics of Dataflow Testing, Strategies in Dataflow Testing, Application of Dataflow Testing

Unit-III		**Classes: 12**

Domain Testing: Domains and Paths, Nice & Ugly Domains, Domain testing, Domains and Interfaces Testing, Domain and Interface Testing, Domains and Testability.

Unit-IV		**Classes: 12**

Paths, Path products and Regular expressions: Path Products & Path Expression, Reduction Procedure Applications, Regular Expressions & Flow Anomaly Detection.
Logic Based Testing: Overview, Decision Tables, Path Expressions, KV Charts, Specifications.

Unit-V		**Classes: 12**

State, State Graphs and Transition Testing: State Graphs, Good & Bad State Graphs, State Testing, Testability Tips. Graph Matrices and Application: Motivational Overview, Matrix of Graph, Relations, Power of a Matrix, Node Reduction Algorithm, Building Tools.

Text Books:

- Software testing techniques – Boris Beizer, Dreamtech, second edition

Reference Books:

- The craft of software testing - Brian Marick, Pearson Education.
- Software Testing- Yogesh Singh, Camebridge
- Software Testing, 3rd edition, P.C. Jorgensen, Aurbach Publications (Dist.by SPD).
- Software Testing, N.Chauhan, Oxford University Press.
- Introduction to Software Testing, P.Ammann & J.Offutt, Cambridge Univ. Press.

Web References:

- https://www.guru99.com/software-testing.html
- https://www.tutorialspoint.com/software_testing/
- https://www.toolsqa.com/software-testing-tutorial/
- https://www.w3schools.in/category/software-testing/

E-Text Books:

- http://read.pudn.com/downloads107/ebook/443059/[J2ME%E5%BC%80%E5%8F%91%E5%A4%A7%E5%85%A8].Java_J2ME_TheCompleteReference_McGrawHill_Osborne.pdf
- https://www.google.com/search?q=TEXT+BOOK+FOR+SOTWARE+TESTING&rlz=1C1GCEU_enIN823IN827&oq=TEXT+BOOK+FOR+SOTWARE+TESTING&aqs=chrome..69i57j0l3.10273j0j8&sourceid=chrome&ie=UTF-8

R & ANALYTICS LAB

V – Semester								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CI A**	**SEE**	**Total**
17CF00510	core	-	-	3	2	40	60	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 45				Total classes:45		

Course Objectives:

- Design use cases and develop the use case model.
- Capture a business process model.
- Practice the object oriented analysis and design through UML on a particular application.
- Explore tools that support UML and object oriented software development.

Course Outcomes:

- Can define what a typical data mining is and what it can be applied for.
- Can determine the different steps followed in Data mining and pre-processing for Data mining.
- Able to apply Association Rule mining.
- Apply at least one of the Preceiction methods for data mining.

Week-1 **Installation Of R**

Installing R in windows, R Console (R window to edit and execute R Commands), Commands and Syntax (R commands and R syntax), Packages and Libraries (Install and load a package in

Week -2 **Implement The Data Structures Using R Programming**

Introduction to Data Types (Why Data Structures?, Types of Data Structures in R), Vectors, Matrices, Arrays, Lists, Factors, Data Frames, Importing and Exporting Data.

Week - 3,4 **Implement The Graphical Analysis Using R**

Creating a simple graph (Using plot() command), Modifying the points and lines of a graph (Using type, pch, font, cex, lty, lwd, col arguments in plot() command), Modifying Title and Subtitle of graph (Using main, sub, col.main, col.sub, cex.main, cex.sub, font.main, font.sub arguments in plot() command), Modifying Axes of a Graph (Using xlab, ylab, col.lab, cex.lab, font.lab, xlim, ylim, col.axis, cex.axis, font.axis arguments and axis() command), Adding Additional Elements to a Graph (Using points(), text(), abline(), curve() commands), Adding Legend on a Graph (Using legend() command), Special Graphs (Using pie(), barplot(), hist() commands), Multiple Plots (Using mfrow or mfcol arguments in par() command and layoutcommand).

Week - 5,6 **Implement The Descriptive Statistics Using R.**

Measure of Central Tendency (Mean, Median and Mode), Measure of Positions (Quartiles, Deciles, Percentiles and Quantiles), Measure of Dispersion (Range, Median, Absolute deviation about median, Variance and Standard deviation), Measure of Distribution (Skewness and Kurtosis), Box and Whisker Plot (Box Plot and its parts, Using Box Plots to compare

Week -7,8 **In Memory Data Analytics**

Window and text functions in SQL; Advanced SQL functions

Reference books:

- Pascal Roques, “Modeling Software Systems Using UML 2”, WILEY- Dreamtech India Pvt.Ltd 2^{ND} Edition, 2007
- Pieter Adrians, DolfZantinge, “Data Mining”, Addison Wesley, Peter V, 2000.

MOBILE APPLICATION DEVELOPMENT LAB

V – Semester

Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
17CF00511	core			3	2	40	60	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 45				Total classes:45		

Objectives:

- To understand fundamentals of android operating systems.
- Illustrate the various components, layouts and views in creating android applications
- To understand fundamentals of android programming.

Outcomes:

- Create data sharing with different applications and sending and intercepting SMS.
- Develop applications using services and publishing android applications.
- To demonstrate their skills of using Android software development tools

Week 1	**Setting Up the Development Environment :** 1.1 Download/Install the SDK For in-depth instructions, visit Android Installation Documentation. Otherwise perform the following steps. • Go to http://developer.android.com/sdk/index.html. • Unpack to a convenient location - Remember the full path to this location, we will refer to it as <android_sdk_dir>for the rest of the lab. o *<android_sdk_dir> would then be/home/<username>/android_dir.* • Add the path to the <android_sdk_dir>/tools directory to your system PATH o Windows: 1. Right-click My Computer. 2. Click Properties. 3. Click Advanced tab. 4. Click Environment Variables button. 5. Double Click Path under System Variables. 6. Add ; <android_sdk_dir>/tools;<android_sdk_dir>/platform-tools to the end of the Variable Values text field. • Navigate to your <android_sdk_dir>/tools directory and type android. Add the appropriate components. See step 4 in http://developer.android.com/sdk/installing.html. • Test your installation by running adb from the command line. If you did everything right, you should get a long list of help instructions. 1. Click the menu Help -> Software Updates.

	2. Click Available Software tab -> Add Site button. 3. Enter https://dl-ssl.google.com/andriod/eclipse into the "Location" field. 4. Click OK button. 5. Click the checkbox next to Developer Tools. 6. Click the Install button. 7. Click the Next button. 8. Accept the terms, click Finish. 9. Restart Eclipse.
Week 2	1.2 Download/Install the Eclipse Plugin • It is recommended that you use Eclipse 3.4 or later o **Lab Machines**- Fedora Eclipse based on 3.4.2 The version of Eclipse used by the lab machines is missing a vital component and requires adding an additional Eclipse plugin in order to use the Android plugin: 1. Click the menu Help -> Software Updates. 2. Click the tab Available Software -> Add Site button. 3. Enter http:// download.eclipse.org/releases/ganymede into the Location field. 4. Click OK button. 5. Enter WST Common UI into the search/text box at the top of the window (give it a second, it tries to search as you type and its kind of slow). 6. Click the checkbox next to WST Common UI. 7. Click the Install button. 8. Click the Next button. 9. Accept the terms, click Finish. 10. Restart Eclipse. 11. Follow the steps in the next bullet 3.4 Ganymede. o **Eclipse 3.4 Ganymede:** 1. Click the menu Help -> Software Updates. 2. Click Available Software tab -> Add Site button. 3. Enter https://dl-ssl.google.com/andriod/eclipse into the "Location" field. 4. Click OK button. 5. Click the checkbox next to Developer Tools. 6. Click the Install button. 7. Click the Next button. 8. Accept the terms, click Finish. 9. Restart Eclipse. o **Eclipse 3.5 Galileo:** 1. Click Help -> Install New Software. 2. Click Add... button. 3. Enter a name for the site into the Name field. 4. Enter htpps://dl-ssl/google.com/android/eclipse/ into the Location field. 5. Click OK button. 6. Click the checkbox next to Developer Tools. 7. Click the Next button. 8. Accept the terms, click Finish. 9. Restart Eclipse. • Point Eclipse to <android_sdk_dir>: 1. Click the menu Window -> Preferences. 2. Click Android from the Hierarchy view on the left hand side. 3. Enter <android_sdk_dir> into the SDK Location field. 4. Click the Apply button. 5. Click the OK button.

Week 3	1.3 Download/Install the SDK Platform Components At the time of writing this lab there are eight different versions of the Android Platform available, ranging from 1.1 to 2.2. It is best practice to develop for the oldest platform available that still provides the functionality you need. This way you can be assured that your application will be supported by as many devices as possible. However, you will still want to download newer versions of the platforms so that you can test your applications against these as well. Due to the size of each platform component you will only be required to download and develop on one platform for the whole class. We will target the highest platform that the G1 phones support, Android 1.6 (API 4). Before we can begin developing we must download and install this platform: • Select the menu Window ->"Android SDK and AVD Manager", or click on the black phone shaped icon in the toolbar. • Select Available Packages on the left hand side. • Expand the Google Android site in the "Site, Packages, and Archives" Tree. • Check the following items: o SDK Plaform Android 1.6, API 4 Revision 3 o Google APIs by Google Inc., Android API 4, Revision 2 o *NOTE: Those of you developing on Lab Machines should follow these instructions:* http://sites.google.com/site/androidhowto/how-to-1/set-up-the-sdk-on-lab-machines-linux. • Click Install Selected. • Accept the Terms for all packages and click Install Accepted. We're now ready to develop our application.
Week 4	2. Create "Hello World" Application 2.1 Create a new Android Project 2.2 Run "Hello World" on the Emulator 2.3 On a Physical Device 2.4 Greeting the User
Week 5	3. Create Application by Using Widgets 3.1 Creating the Application by using the Activity class (i) onCreate() (ii) onStart() (iii) onResume() (iv) onPause() (v) onStop() (vi) onDestroy() (vii) onRestart()
Week 6	3.2 Creating the Application by using Text Edit control. 3.3 Creating the Application Choosing Options (i) CheckBox (ii) RadioButton (iii) RadioGroup (iv) Spinner

Week-7	4. Create Application by Using Building Blocks for Android Application Design 4.1 Design the Application by using (i) Linear Layout (ii) Relative Layout (iii) Absolute Layout.
Week-8	5. Create Aplication by Using Building Menus and Storing Data 5.1 Design the Application for Menus and Action Bar 5.2 Design the application to display the Drop-Down List Action Bar
References:	
https://www.alljntuworld.in/download/mobile-application-development-lab-manual/	

SCRIPTING LANGUAGES LAB

V – Semester

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
17CF00512	core			3	2	40	60	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 45				Total classes:45		

Course Objectives:

- Learn to apply fundamental problem solving techniques
- To develop hands on experience using open source technologies such as HTML, CSS, JavaScript, PHP
- Teach an example of scripting and interpretative language and compare it with classical compiled programming languages
- Expose students to application development and prototyping using Python

Course Outcomes:

- Design real life situational problems and think creatively about solutions of them.
- Apply a solution clearly and accurately in a program using Python.
- Apply the best features of Python to program real life problems

Week-1	Write a hello world program in angularJS Write a program to add sum of two numbers in angularJS
Week-2	Write a program in AngularJS features to make a shopping list a. Adding items into cart. b. Removing items from cart. c. Identifying the repeated elements in the cart. Create a student information form with submit and reset functionality
Week-3	Write programs to display the date and time of last visited page using cookie Write a program to display the session count using PHP programming
Week-4	Write a PHP program if a searched string is located at the beginning of a sentence or not. Create a student registration form in php.
Week-5	Write a function dups to find all duplicates in the list. Write a function cumulative product to compute cumulative product of a list of
Week-6	Write a program to convert a list of tuples in a dictionary. Write a Python program to check whether an element exists within a tuple.

Week-7	Write a Python script to concatenate following dictionaries to create a new one.
	Sample Dictionary :
	dic1={1:10, 2:20}
	dic2={3:30, 4:40}
	dic3={5:50,6:60}
	Write a Python program to replace dictionary values with their average.

Week-8	Write a program to read character data from a text file by using the following read methods. read(),read(n),readline(),readlines(). Write a program to print number of lines, words& characters present in the given file.

Week-9	Write a program to describe about Instance variable using ATM Machine Class Write a program to describe about Class variable using Robot Class
Week-10	Write a program to access static & instance variable in a class. Write a program to demonstrate the user defined & predefined exceptions.
Week-11	Write a Program to illustrate how to overload a binary + operator. Write a program to create a thread in python by using 3 ways A) Creating a thread without using any class. B) Create a thread by extending thread class. C) Create a thread without extending thread class.
Week-12	Write a program to create user daemon thread. Write a program for synchronization by using rlock. Write a program to connect with oracle database and prints its version
Week-13	Write a program to create employees table in the oracle database. Write a program to insert multiple rows in the employees table with dynamic input from the keyboard.
References:	
• ***<u>A Practitioner's Guide to Software Test Design</u>***, Lee Copeland, 2003	