UNIVERSITY OF CALICUT
G & A - IV - J

No. 67485/GA - IV - J1/2015/Admn (II) Calicut University.P.O

Dated: 03.12.2025

From

The Registrar

To

The Principal,
Sacred Heart College (Autonomous),
Chalakudy

Sir,

Sub:- Syllabus of M.Sc Computer Science programme w.e.f 2025 admission - Approval by the Board of Studies in Computer Science and Application (PG) - Intimating of - Reg.

Ref:- 1. Your letter dated: 24.10.2025 forwarding the Syllabus of various FYUG & PG Programmes.

2. Minutes of the meeting of Board of Studies in Computer Science and Application (PG) held on 12.11.2025

3. Orders of the Vice chancellor in the file of Even No. dated: 29.11.2025

With reference to the above, I am to inform you that, the Board of Studies in Computer Science and Application (PG) has approved the syllabus of **M.Sc Computer Science** programme w.e.f 2025 admission, submitted by your college.

Yours faithfully

Arsad M
Deputy Registrar
(For The Registrar)

SACRED HEART COLLEGE (AUTONOMOUS), CHALAKUDY

DEGREE OF MASTER OF SCIENCE (M.Sc.) IN COMPUTER SCIENCE

AS PER

CHOICE BASED CREDIT AND SEMESTER SYSTEM - CBCSS (PG)

OBE BASED PROGRAMME CURRICULUM

(EFFECTIVE FROM 2025 ADMISSION ONWARDS)

BOARD OF STUDIES IN COMPUTER SCIENCE AND APPLICATIONS (PG)

CONTENTS

I. REGULATIONS FOR THE DEGREE OF MASTER OF SCIENCE (COMPUTER SCIENCE)

II. PROGRAM STRUCTURE

III. DETAILED SYLLABUS

IV. FORMAT FOR THE PROJECT REPORT (APPENDIX A)

REGULATIONS FOR THE DEGREE OF MASTER OF SCIENCE (COMPUTER SCIENCE) CBCSS – PG

EFFECTIVE FROM 2024 ADMISSION ONWARDS

1. PROGRAMME OBJECTIVES

The course of the MSc (Computer Science) Programme is designed with the following objectives:

1. To equip students to take up challenging research-oriented responsibilities and courses for their higher studies/profession.
2. To train and equip the students to meet the requirements of the Software industry in the country and outside.
3. To motivate and support the students to prepare and qualify challenging competitive examinations such as JRF/NET/JAM/GATE etc.

2. PROGRAMME OUTCOME (PO)

After the successful completion of the Post Graduate Programme, M.Sc Computer Science at Sacred Heart College, a student would have :

PO1:	Advanced Subject Mastery: Graduates will achieve profound and expansive knowledge in their respective fields, empowering them to become thought leaders and innovators.
PO2:	Elevated Critical Thinking: Students will refine their ability to critically evaluate complex theories and problems, enabling them to generate sophisticated and original solutions
PO3:	Exemplary Communication Proficiency: Graduates will master the art of eloquent and precise communication, effectively conveying intricate ideas to both academic and professional audiences.
PO4:	Ethical Stewardship and Leadership: Students will embody ethical principles and demonstrate leadership that is both visionary and principled, fostering trust and integrity in all endeavours.

PO5:	Synergistic Interdisciplinary Collaboration: Graduates will excel in forging interdisciplinary connections, leveraging diverse perspectives to tackle global challenges with holistic approaches.
PO6:	Research Excellence and Innovation: Students will engage in groundbreaking research, contributing novel insights and advancing the frontiers of knowledge in their disciplines.
PO7:	Cultural and Global Awareness: Graduates will develop a deep understanding of cultural and global dynamics, enabling them to operate with sensitivity and effectiveness in an interconnected world.
PO8:	Lifelong Intellectual Curiosity: Students will cultivate an enduring passion for learning, ensuring they remain at the forefront of their fields through continuous personal and professional development.
PO9:	Analytical and Quantitative Expertise: Graduates will acquire sophisticated analytical and quantitative skills, empowering them to interpret data and make informed, impactful decisions.
PO10:	Psychological Insight and Empathy: Students will enhance their psychological acumen and empathy, enabling them to understand and influence human behaviour in personal, academic, and professional contexts.

3. PROGRAMME SPECIFIC OUTCOME (PSO)

PSO1:	Evaluate complex real-world problems by applying principles of theoretical computing, engineering and Mathematical models.
PSO2:	Modern Tool usage: Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.
PSO3:	Understand all dimensions of the concepts of software application development and projects.
PSO4:	Aware the students to publish their work in reputed journals.

PSO5:	Conceive Project Management capabilities to solve real world problems in accordance to the needs of the industry, in a specific time frame.
PSO6:	Design and develop computer programs/computer-based systems in the field of Computer Sciences viz. Computational Intelligence, Machine learning, Web technology, Information Retrieval Systems, Data Analytics, Communication and networking.
PSO7:	To prepare the students to address the challenging requirements coming from the enterprise applications.

4. DMISSION

1. The admission to all M.Sc Computer Science programmes shall be as per the rules and regulations of the University.
2. The eligibility criteria for admission shall be as announced by the University from time to time.
3. Separate rank lists shall be drawn up for reserved seats as per the existing rules.
4. The college shall make available to all the admitted students the information regarding all the courses including electives offered with syllabus and credit for the entire course.
5. There shall be a uniform calendar prepared by the University for the Conduct of the programmes.
6. There shall be provision for inter collegiate and inter University transfer in the 2nd and 3rd semester within a period of two weeks from the date of commencement of the semesters.
7. There shall be provision for credit transfer subject to the conditions specified by the Board of Studies concerned.
8. There shall be a uniform calendar prepared by the University for the registration, conduct/schedule of the courses, examinations and publication of results.

5. READMISSION

1. There shall be provision for readmission of students.
2. For readmission, the vacancy should be within the sanctioned strength in the parent college. If there is no vacancy in the junior batch of the parent college, readmission can be taken in another college with the junior batch, if there is vacancy within the sanctioned strength in the concerned college.
3. This readmission is not to be treated as college transfer.

4. There should be a gap of at least one semester for readmission.
5. The candidate seeking readmission to a particular semester should have registered for the previous semester examination.
6. Readmission shall be taken within two weeks from the date of commencement of the semester concerned.
7. The Principal can grant readmission to the student, subject to the above conditions, and inform the matter of readmission to the Controller of Examinations within one month of such readmission.
8. If change in scheme occurs while readmission, provision for credit transfer will be subject to the common guidelines prepared by the Board of Studies/ Faculty concerned.

6. DURATION OF THE PROGRAMME

1. The minimum duration for completion of a four semester PG Programme is two years. The maximum period for completion is 4 years.
2. The duration of each semester shall be 90 working days, inclusive of examinations, spread over five months.
3. Odd semesters shall be held from June to October and even semesters from November to March subject the academic calendar of the University.

7. PROGRAMME STRUCTURE

1. The Programme includes three types of courses, viz., Core courses (Code C), Elective Courses (Code E) and Audit Courses (Code A).
2. Every student of the MSc Computer Science Programme shall have to work on a project/dissertation of not less than 8 credits under the supervision of a faculty member as per the curriculum. Project/dissertation shall be treated as Core Courses.
   Project Work is mandatory for all regular Programmes and Comprehensive Vivavoce is optional and these shall be done in the end semester. The combined Credit for the Project Work and Comprehensive Viva-voce shall not be more than 8 (eight) credits subject to a minimum of 4 (four) credit for Project Work. All students have to submit a Project Report/Dissertation in the prescribed structure and format as a part of the Project Work undertaken.
3. Total credit for the Programme shall be 80 (eighty), this describes the weightage of the course concerned and the pattern of distribution is as detailed below

   i) Total Credit for Core Courses shall not be less than 60 (sixty) and not more than 68 (sixty-eight). ii) Total Credit for Elective Course shall not be less than 12 (twelve) and not more than 20 (Twenty).

iii) Total Credits for Comprehensive Viva-voce and Project Work combined together shall be 8 (eight) subject to a minimum of 4 (four) credit for Project Work.

iv) Total credit in each semester shall vary between 18 to 22.

v) No course shall have less than 2 credits and more than 5 credits.

1. Elective courses shall be spread over either in the Third & Fourth Semesters combined.
2. Audit Courses: There will be two Audit Courses (Ability Enhancement Course & Professional Competency Course) with 4 credits each. These have to be done one each in the first two semesters. The credits will not be counted for evaluating the overall SGPA & CGPA. The colleges shall conduct examinations for these courses and have to intimate
   /upload the results of the same to the University on the stipulated date during the III Semester. Students have to obtain only minimum pass requirements in the Audit Courses.
3. A student shall accumulate a minimum of 80 credits for the successful completion of the Programmes.

8. REGISTRATION

1. A student shall be permitted to register for a Programme at the time of admission.
2. A student who registers for a Programme shall complete it within 4 years.
3. The college shall send a list of students registered for each Programme in each semester giving the details of courses registered to the university in the prescribed form within 45 days of the commencement of the semester.
4. Students shall be normally permitted to register for the examination if they have required minimum attendance. If the student has a shortage of attendance in a semester, the student shall be permitted to move to the next semester and can write the examination for the entire courses of the semester in which shortage of attendance occurs as supplementary examination only after the completion of the entire Programme. In such cases, a request from the student may be forwarded through the Principal of the college to the Controller of Examinations within two weeks of the commencement of the semester. There will not be any Repeat semester in CBCSS PG 2021.
5. The students who have attendance within the limit, but could not register for the semester examinations, have to apply for token registration, within two weeks of the commencement of the next semester.

9. ATTENDANCE

1. The students admitted in the PG Programmes in affiliated colleges shall be required to attend at least 75 percent of the total number of classes

(theory/practical) held during each semester. The students having less than prescribed percentage of attendance shall not be allowed to appear for the University examination.

2. Condonation of shortage of attendance for a maximum of 9 days (10% of the working days in a semester) in the case of single condonation and 18 days (20% of the working days in a semester) in the case of double condonation in a semester subject to a maximum of two times (for single condonation only) during the whole period of Post Graduate Programme may be granted by the University as per the existing procedures. In the case of double condonation, only one condonation shall be allowed during the entire Programme.

3. Benefit of condonation of attendance will be granted to the students on health grounds, for participating in University Union activities, meeting of the University bodies /Govt. bodies and participation in other extracurricular activities on production of genuine supporting documents, with the recommendation of the Head of the Department concerned.

4. A student who is not eligible for such condonation shall be observed the provisions as per clause 6.4 of this regulation. The principal should intimate the details of these candidates at the commencement of the next semester.

5. Women students can avail maternity leave as per the existing university rules.

10. EXAMINATION

1. There shall be a University examination at the end of each semester.
2. Practical examinations shall be conducted by the University at the end of each semester. There will be one internal and one external examiner for the conduct of End Semester Practical examination.
3. Project Work / Dissertation shall be evaluated at the end of the Programme only. There shall be both Internal and External evaluation for the Project Work.
4. There shall be one end-semester examination of 3 hours duration for each theory course and practical course.

11. EVALUATION AND GRADING

1. Evaluation: The evaluation scheme for each course shall contain two parts; (a) Internal / Continuous Assessment (CA) and (b) External / End Semester Evaluation (ESE).
2. Of the total, 20% weightage shall be given to Internal evaluation / Continuous assessment and the remaining 80% to External/ESE and the ratio and weightage between Internal and External is 1:4.

3. Primary evaluation for Internal and External shall be based on 6 letter grades (A+, A, B, C, D and E) with numerical values (Grade Points) of 5, 4, 3, 2, 1 & 0 respectively.
4. Grade Point Average: Internal and External components are separately graded and the combined grade point with weightage 1 for Internal and 4 for external shall be applied to calculate the Grade Point Average (GPA) of each course. Letter grade shall be assigned to each course based on the categorization based on Ten-point Scale.
5. Evaluation of Audit Courses: The examination and evaluation shall be conducted by the college itself either in the normal structure or MCQ model from the Question Bank and other guidelines provided by the University/BoS. The Question paper shall be for minimum 20 weightage and a minimum of 2 hour duration for the examination. The result has to be intimated / uploaded to the University during the Third Semester as per the notification of the University.

12. INTERNAL EVALUATION – CONTINUOUS ASSESSMENT

1. This assessment shall be based on a predetermined transparent system involving periodic written tests, assignments, seminars and viva-voce in respect of theory courses and based on tests, lab skill and records/viva in respect of practical courses.
2. The criteria and percentage of weightage assigned to various components for internal evaluation are as follows:

   (a) Theory: The weightage assigned to various components for internal evaluation for theory papers is as shown below.

Sl.No	Component	Percentage	Weightage
1	Examination /Test	40%	2
2	Seminars / Presentation	20%	1
3	Assignment	20%	1
4	Attendance	20%	1

To ensure transparency of the evaluation process, the internal assessment grade awarded to the students in each course in a semester shall be published on the notice board at least one week before the commencement of external examination. There shall not be any chance for improvement for internal grade.

The course teacher shall maintain the academic record of each student registered for the course, which shall be forwarded to the University, through the college Principal.

(b) Practical: The mark distribution to award internal continuous assessment marks for practical course should be as follows:

Sl.No	Component	Percentage	Weightage
1	Lab Skill	40%	4
2	Records/viva	30%	3
3	Practical Test	30%	3

Note:

- All students should have a rough record (observation note book) in which they write all the works to be carried out in the lab prior to his/her entering the lab. (S)he may also note down the i/p and o/p that (s)he gives for program verification in the observation note book (rough record).
- All lab works should be neatly recorded in a Laboratory Record Book (Fair Record) in written form. However, program results can be pasted in the lefthand side of the fare record.
- Chairperson, Board of Examination (PG) has to prepare the modalities of the practical papers (list of experiments to be done, number of minimum experiments required in the practical record etc.) and distributed to all departments concerned, at the beginning of each semester itself. Model lists of experiments are provided with the syllabus for each practical session.
- No candidate will be permitted to attend the end-semester test unless he/she produces a certified record of the laboratory.

3. Grades shall be given for the internal evaluation based on the grades A+, A, B, C, D & E with grade points 5,4,3,2, 1 & 0 respectively. The overall grades shall be as per the Ten Point scale.
4. There shall be no separate minimum Grade Point for internal evaluation.
5. To ensure transparency of the evaluation process, the internal assessment marks awarded to the students in each course in a semester shall be published on the notice board before 5 days of commencement of external examination.
6. There shall not be any chance for improvement of internal marks.
7. The course teacher shall maintain the academic record of each student registered for the course, which shall be forwarded to the University, through the college Principal, after being endorsed by the Head of the Department.
8. For each course there shall be class test/s during a semester. Grades should be displayed on the notice board. Valued answer scripts shall be made available to the students for perusal.
9. Each student shall be required to do assignment/s for each course. Assignments after valuation must be returned to the students. The teacher shall define the expected quality of the above in terms of structure, content, presentation etc. and inform the same to the students. Punctuality in submission is to be considered.
10. Every student shall deliver Seminar / Presentation as an internal component for every course and must be evaluated by the respective course teacher in terms of structure,

content, presentation and interaction. The soft and hard copies of the seminar report are to be submitted to the course teacher.

11. All the records of Continuous Assessment (CA) must be kept in the college and must be made available for verification by university, if asked for. Calculation of overall internal grade for one theory course will be done as shown below:

Components	Weightage (W)	Grade Awarded	Grade Point(GP)	Weighted GP	Overall Grade of the course
Examination /Test	4	A	4	16	Weighted GP/Total Weight 43/10 = 4.30
Seminars / Presentation	3	A+	5	15	
Assignments	3	A	4	12	
Total	10			43	O

Calculation of overall internal grade for one Lab Course will be done as shown below:

Components	Weightage (W)	Grade Awarded	Grade Point(GP)	Weighted GP	Overall Grade of the course
Lab Skill	2	A	4	8	Weighted GP/Total Weight 22/5 = 4.40
Records/viva	1	A+	5	5	
Practical Test	1	A	4	4	
Viva-voce	1	A+	5	5	
Total	5			22	

13. EXTERNAL / END SEMESTER EVALUATION (ESE)

1. The semester-end examinations in theory courses shall be conducted by the University with question papers set by external experts. The evaluation of the answer scripts shall be done by examiners based on a well-defined scheme of valuation.

2. After the external evaluation, only Grades are to be entered in the space provided in the answer script for individual questions and calculations need to be done only up to the Cumulative Grade Point (CGP) and all other calculations including grades are to be done by the University.

3. Students shall have the right to apply for revaluation or scrutiny as per rules within the time permitted for it.
4. Photocopies of the answer scripts of the external examination shall be made available to the students for scrutiny on request by them as per rules.
    4. The external evaluation shall be done immediately after the examination preferably in a Centralized Valuation Camp.

5. The language of writing the examination shall be English.

6. Pattern of questions for external/ESE (theory courses):

   a. Questions shall be set to assess the knowledge acquired, standard, and application of knowledge, application of knowledge in new situations, critical evaluation of knowledge and the ability to synthesize knowledge. Due weightage shall be given to each module based on content/teaching hours allotted to each module.

   b. It has to be ensured that questions covering all skills are set. The setter shall also submit a detailed scheme of evaluation along with the question paper.

   c. A question paper shall be a judicious mix of short answer type, short essay type /problem solving type and long essay type questions.

   d. The question shall be prepared in such a way that the answers can be awarded A+, A, B, C, D, E Grades.

   e. Weightage: Different types of questions shall be given different weightages to quantify their range given in the following model:

Sl. No.	**Type of Questions**	**Individual weightage**	**Total Weightage**	**Number of questions to be answered**
1	Short Answer type questions	2	2×4 = 8	4 out of 7
2	Short essay/ problem solving type	3	3×4 = 12	4 out of 7
3	Long Essay type questions	5	5×2 = 10	2 out of 4
Total			30	18

   f. Questions should be asked as far as possible from all modules following a uniform distribution.

A sample ESE evaluation sheet of a theory course is illustrated below:

Type of Question	Qn. No	Grade Awarded	Grade Point	Weightage	Weighted Grade Point	Calculation
	1	A+	5	2	10	
	2	-	-	-	-	

Short Answer type	3	A	4	2	8	Overall Grade of the theory paper = Sum of Weighted Grade Points / Sum of the weightage 115/30 = 3.83 = Grade A+
	4	C	2	2	4	
	5	-	-	-	-	
	6	A	4	2	8	
	7	-	-	-		
Medium Essay type	8	B	3	3	9	
	9	A+	5	3	15	
	10	-	-	-	-	
	11	-	-	-	-	
	12	-	-	-	-	
	13	A	4	3	12	
	14	B	3	3	9	
Long Essay type	15	A+	5	5	25	
	16	-	-	-	-	
	17	-	-	-	-	
	18	B	3	5	15	
TOTAL				30	115	

7. End Semester Evaluation in Practical Courses shall be conducted and evaluated by both Internal and External Examiners.

Mark distribution for practical courses shall be as follows:

Component	Weightage
Algorithm/Flow diagram/UI diagram/Class	6
Implementation	6
Result/ Output	6
Record	6
Viva	6
Total	**30**

A sample ESE evaluation sheet of a theory course is illustrated below:

Type of Question	Grade Awarded	Grade Point	Weightage	Weighted Grade Point	Calculation
Algorithm/Flow diagram/UI diagram/Class Diagram	A	4	6	24	114/30 = 3.80
Implementation	A	4	6	24	
Result/ Output	B	3	6	18	
Record	A	4	6	24	
Viva	A	4	6	24	
Total			**30**	**114**	

14. EVALUATION OF PROJECT WORK / DISSERTATION

1. There shall be External and Internal evaluation for Project Work done and the grading system shall be followed.
2. One component among the Project Work evaluation criteria shall be Viva-voce (Project Work related) and the respective weightage shall be 40%.
3. Consolidated Grade for Project Work is calculated by combining both the External and Internal in the Ratio of 4:1 (80% & 20%).
4. For a pass in Project Work, a student has to secure a minimum of P Grade in External and Internal examination combined. If the students could not secure minimum P Grade in the Project work, they will be treated as failed in that attempt and the students may be allowed to rework and resubmit the same in accordance with the University exam stipulations. There shall be no improvement chance for Project Work.
5. The External and Internal evaluation of the Project Work shall be done based on the following criteria and weightages as detailed below:

No	Criteria	% of Weightage	Weightage	
			External	Internal
1	Relevance of the topic and Statement of problem, Methodology & Analysis Quality of Report & Presentation	60%	24	6
2	Viva-voce	40%	16	4
Total Weightage		**100%**	**40**	**10**

The first component for 60% weightage can be sub-divided into following project implementation components:

Sl No	Components	Weightage	
		External	Internal
1	Relevance of the Topic, Statement of Objectives, Methodology	2	2
2	Quality of Literature Survey/Product Review	2	
3	Quality of Analysis Phase	2	
4	Quality of Design Phase	2	
5	Quality of Implementation/Simulation	4	2
6	Quality of Testing/Result Analysis	2	
7	Quality of Contributions	2	
8	Identification of Future Work	1	2
9	Quality of Project Report	4	
10	Publications/Presentations out of the Project Work*	1	
11	Quality of Presentation	1	
12	Demonstration of the Project Work	1	
13	General Viva Voce	16	4
	Total	**40**	**10**

15. DIRECT GRADING SYSTEM

1. Direct Grading System based on a 10 – Point scale is used to evaluate the performance (External and Internal Examination of students)

2. For all courses (Theory & Practical)/Semester/Overall Programme, Letter grades and GPA/SGPA/CGPA are given on the following way:

   a. First Stage Evaluation for both Internal and External done by the Teachers concerned in the following Scale:

Grade	Grade Points
A+	5
A	4
B	3
C	2
D	1
E	0

   b. The Grade Range for both Internal & External shall be :

Letter Grade	**Grade Range**	**Range of Percentage (%)**	**Merit Indicator**
O	4.25 - 5.00	85.00 - 100.00	Outstanding
A+	3.75 - 4.24	75.00 - 84.99	Excellent
A	3.25 - 3.74	65.00 - 74.99	Very Good
B+	2.75 - 3.24	55.00 - 64.99	Good
B	2.50 - 2.74	50.00 - 54.99	Above Average
C	2.25 - 2.49	45.00 - 49.99	Average
P	2.00 -2.24	40.00 - 44.99	Pass
F	< 2.00	Below 40	Fail
I	0	-	Incomplete
Ab	0	-	Absent

3. No separate minimum is required for Internal evaluation for a pass, but a minimum P Grade is required for a pass in the external evaluation. However, a minimum P grade is required for pass in a course.

4. A student who fails to secure a minimum grade for a pass in a course will be permitted to write the examination along with the next batch.

5. Improvement of Course- The candidates who wish to improve the grade / grade point of the external examination of a course/s they have passed already can do the same by

appearing in the external examination of the concerned semester along with the immediate junior batch

6. Betterment Programme One time- A candidate will be permitted to improve the CGPA of the Programme within a continuous period of four semesters immediately following the completion of the Programme allowing only once for a particular semester. The CGPA for the betterment appearance will be computed based on the SGPA secured in the original or betterment appearance of each semester whichever is higher.

16. SEMESTER GRADE POINT AVERAGE (SGPA)

The SGPA is the ratio of sum of the product of the number of credits with the grade points scored by a student in all the courses taken by a student and the sum of the number of credits of all the courses taken by a student.

After the successful completion of a semester, Semester Grade Point Average (SGPA) of a student in that semester is calculated using the formula given below. Semester Grade

Point Average - SGPA (Sj) = Σ(Ci x Gi) / Cr

(SGPA= Total Credit Points awarded in a semester / Total credits of the semester)

Where 'Sj' is the j^{th} semester , 'Gi' is the grade point scored by the student in the i^{th} course 'Ci is the credit of the i^{th} course, 'Cr' is the total credits of the semester.

17. CUMULATIVE GRADE POINT AVERAGE (CGPA) - CALCULATION

Cumulative Grade Point Average (CGPA) = Σ (Ci x Si) / Cr

(CGPA= Total Credit points awarded in all semesters/Total credits of the programme)

Where C_1 is the credit of the I^{st} semester S1 is the SGPA of the 1^{st} semester and Cr is the total number of credits in the programme. The CGPA is also calculated in the same manner taking into account all the courses undergone by a student over all the semesters of a programme. The SGPA and CGPA shall be rounded off to 2 decimal points.

For the successful completion of a semester, a student should pass all courses and score a minimum SGPA of 2.0. However, the students are permitted to move to the next semester irrespective of their SGPA.

18. GRADE CARD

1. The University shall issue to the students grade card on completion of each semester, which shall contain the following information:
    - Name of University

- Name of College
- Title of PG Programme
- Semester concerned
- Name and Register Number of the student.
- Code number, Title and Credits of each Course opted in the semester including Audit Courses
- Letter grade in each course in the semester
- The total credits, total credit points and SGPA in the Semester (corrected to three decimal places)

2. The final Grade card issued at the end of the final semester shall contain the details of all courses taken during the entire Programme, including those taken over and above the prescribed minimum credits for obtaining the degree. The final grade card shall show CGPA (corrected to three decimal places), percentage of marks (corrected to two decimal places) and the overall letter grade of a student for the entire Programme. The final Grade card will also contain the list of Audit courses.

19. AWARD OF DEGREE

The successful completion of all the courses with P Grade shall be the minimum requirement for the award of the degree.

20. POSITION CERTIFICATE

The University publishes list of top 10 positions for each Programme after the publication of the Programme results. Position certificates shall be issued to candidates who secure positions from 1st to 10th in the list. The position list shall be finalized after the result of revaluation.

The position list shall be prepared in the order of merit based on the CGPA scored by the students. Grace Grade points awarded to the students shall not be counted for fixing the position.

21. GRIEVANCE REDRESSAL COMMITTEE

Department Level Committee:

The college shall form a Grievance Redressal Committee in each department comprising of course teacher, one senior teacher and elected representative of Students (Association Secretary) as members and the Head of the Department as Chairman. The committee shall have initial jurisdiction over complaints against Continuous Assessment.

College Level Committee:

There shall be a college level grievance redressal committee comprising of student adviser, two senior teachers, two staff council members (one shall be elected member) and elected representative of students (College Union Chairperson) as members and the Principal as Chairman. This committee shall address all grievances relating to the internal assessment grades of the students.

University level:

The University shall form a Grievance Redressal Committee as per the existing norms.

II M.Sc COMPUTER SCIENCE(CBCSS)-PROGRAMME STRUCTURE

LEGEND	
Item	**Description**
C	Credits
E	External Component (%)
I	Internal Component (%)
L	Lecture Hours
P	Practical Hours
T	Total

SEMESTER I

No	Course Code	Course Name	C	Weightage			Hrs/Week		
				I	E	T	L	P	T
1.1	CSS1C01	Discrete Mathematical Structures	4	1	4	5	4	0	4
1.2	CSS1C02	Advanced Data Structures	4	1	4	5	3	2	5
1.3	CSS1C03	Theory of Computation 0	4	1	4	5	4	0	4
1.4	CSS1C04	Object Oriented Programming Concepts	4	1	4	5	2	2	4
1.5	CSS1C05	Computer Organization Architecture	4	1	4	5	4	0	4
1.6	CSS1L01	Practical I	2	1	4	5	0	4	4
1.7	CSS1A01	Introduction to Research (Ability Enhancement Audit Course)	4	5	0	5	0	0	0
Total Credits (Excluding Audit Course): 22							**17**	**8**	**25**

SEMESTER II

No	Course Code	Course Name	C	Weightage			Hrs/Week		
				I	E	T	L	P	T
2.1	CSS2C06	Design and Analysis of Algorithms	4	1	4	5	4	0	4
2.2	CSS2C07	Operating System Concepts	4	1	4	5	3	2	5

2.3	CSS2C08	Computer Networks	4	1	4	5	4	0	4
2.4	CSS2C09	Computational Intelligence	4	1	4	5	2	2	4
2.5	CSS2C10	Principles of Compilers	4	1	4	5	4	0	4
2.6	CSS2L02	Practical II	2	1	4	5	0	4	4
2.7	CSS2A02	Term Paper (Professional Competency Audit Course)	4	5	0	5	0	0	0
Total Credits (Excluding Audit Course): 22							**17**	**8**	**25**

SEMESTER III

No	Course Code	Course Name	C	Weightage			Hrs/Week		
				I	E	T	L	P	T
3.1	CSS3C11	Advanced Database Management System	4	1	4	5	3	1	4
3.2	CSS3C12	Python Programming Concepts	4	1	4	5	2	3	5
3.3	CSS3C13	Principles of Software Engineering	4	1	4	5	4	0	4
3.4	CSS3E01	Elective I	4	1	4	5	4	0	4
3.5	CSS3E02	Elective 2	4	1	4	5	4	0	4
3.6	CSS3L03	Practical III	2	1	4	5	0	4	4
Total Credits (Excluding Audit Course): 22							**17**	**8**	**25**

List of Elective Courses for CSS3E01	
Course Code	Course Name
CSS3E01a	Computer Graphics
CSS3E01b	Introduction to Soft Computing
CSS3E01c	Web Technology
CSS3E01d	Bioinformatics
CSS3E01e	Computer Optimization Techniques
CSS3E01f	Numerical and Statistical Methods

List of Elective Courses for CSS3E02	
Course Code	Course Name
CSS3E02a	Pattern Recognition
CSS3E02b	Wireless and Mobile Networks
CSS3E02c	Cryptography and Network Security
CSS3E02d	Advanced Web Technology
CSS3E02e	Virtualisation and Cloud Computing
CSS3E02f	Data Warehousing and Data Mining

SEMESTER IV

No	Course Code	Course Name	C	Weightage			Hrs/Week		
				I	E	T	L	P	T
4.1	CSS4E03	Elective 3	3	1	4	5	5	0	5
4.2	CSS4E04	Elective 4	3	1	4	5	5	0	5
4.3	CSS4P01	Project Requirements Analysis & Design Related Discussion	8	1	4	5	3	1	4
		Project Coding, Testing & Implementation Related Discussion					2	2	4
		Project Evaluation & Assessment					2	0	2
		Project Lab Work					0	5	5
Total Credits (Excluding Audit Course): 14							**17**	**8**	**25**

List of Elective Courses for CSS4E03	
Course Code	**Course Name**
CSS4E03a	Data Compression
CSS4E03b	Pervasive Computing

CSS4E03c	System Security
CSS4E03d	Molecular Simulation and Modelling
CSS4E03e	Fundamentals of Big Data
CSS4E03f	Web Engineering

List of Elective Courses for CSS4E04	
Course Code	**Course Name**
CSS4E04a	Digital Image Processing
CSS4E04b	Advanced Topics In Database Design
CSS4E04c	Software Development for Portable Devices
CSS4E04d	Storage Area Networks
CSS4E04e	Semantic Web
CSS4E04f	Advanced Java Programming

III. DETAILED SYLLABUS

SEMESTER I

CSS1C01 – DISCRETE MATHEMATICAL STRUCTURES

Objectives: To introduce discrete mathematics concepts necessary to understand the basic foundation of Computer Science.

Course Outcome

CO1: Verify the validity of an argument using propositional and predicate logic.

CO2: Understand allocations of set theory by applying operations on set.

CO3: Apply operations of relations and functions in discrete structures.

CO4: Understand applications of Lattices and Boolean algebra in computer science domain.

CO5: Identify Group, Ring and Field in Group Theory.

CO6: Understand applications of Graph Theory and Tree.

CO7: Apply the concepts of graph theory and trees to formulate problem solving.

Detailed Syllabus

Module	Unit	Content	Hrs	Weigtage
I		**Sets and Mathematical Logic**	**20**	**6**
	1	Set Theory - Types of sets, Set operations, Principles of Inclusion and Exclusion.	**4**	
	2	Mathematical Logic - Propositional Calculus - Statement, Connectives, Conditional and Bi-conditional, Equivalence of Formula, Well Formed Formula, Tautologies, Duality Law	**5**	
	3	Functionally Complete Sets of Connectives, Normal Forms, Theory of Inference for the Statement Calculus	**6**	
	4	Predicate Calculus - Statement Functions, Variables and Quantifiers, Free and Bound Variables, Theory of Inference for the Predicate Calculus.	**5**	
II		**Functions and Relations**	**10**	**6**
	5	Functions and Relations: Functions – Types of Functions, Composition of Functions and Inverse Functions	**4**	

	6	Relations - Relations and Their Properties, Functions as relations, Closure of Relations, Composition of relations, Equivalence Relations and Partitions	**4**	
	7	Partial Ordering, Hasse Diagram. The Pigeon Hole Principle.	**2**	
III		**Lattices and Boolean Algebra**	**15**	**6**
	8	Lattices and Boolean Algebra - Lattices and Algebraic Systems.	**5**	
	9	Principles of Duality, Basic Properties of Algebraic Systems Defined by Lattices.	**5**	
	10	Distributive Lattices and Complemented Lattices. Boolean Lattices and Boolean Algebras. Boolean Functions and Boolean Expressions.	**5**	
IV		**Group Theory**	**15**	**6**
	11	Group Theory – Definition and Elementary Properties - Permutation Groups.	**5**	
	12	Cyclic Groups – Subgroups - Cosets and Lagrange's Theorem.	**5**	
	13	Semi group and Monoid. Homeomorphism and Isomorphism.	**4**	
	14	Rings, Integral Domains and Fields.	**1**	
V		**Graph Theory**	**15**	**6**
	15	Introduction, Directed Graph, Undirected Graph, Connected and Disconnected Graphs, Bipartite Graph, Complete Bipartite Graph, Isomorphic Graphs, Sub graph.	**5**	
	16	Paths and Circuits. Shortest Paths in Weighted Graphs - Dijkstra's Algorithm.	**4**	
	17	Eulerian Paths and Circuits, Hamiltonian Paths and Circuits.	**4**	
	18	Trees - Spanning Trees and Cut- Sets, Minimum Spanning Trees - Kruskal's Algorithm, Prim's Algorithm.	**2**	

References:

1. C Liu and D. Mohapatra, *Elements of Discrete Mathematics - A Computer Oriented Approach*, TMH, ISBN: 1259006395.
2. Alan Doerr and Kenneth Levassur, *Applied Discrete Structure for Computer Science*, Galgotia Publications Pvt. Ltd, ISBN: 9780574217554.

3. J. K. Sharma, *Discrete Mathematics*, Macmillan Publishers India Limited, ISBN: 1403924759.

4. J. P. Tremblay and R. Manohar, *Discrete Mathematical Structures with Application to Computer Science*, McGraw-Hill Companies, ASIN: B001FPXR5Y.

Mapping of COs with POs and PSOs

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7
CO1	**3**	**3**	**2**	**1**	**0**	**2**	**0**	**1**	**3**	**1**	**3**	**2**	**1**	**1**	**0**	**2**	**1**
CO2	**3**	**2**	**1**	**0**	**1**	**2**	**0**	**1**	**3**	**0**	**3**	**2**	**1**	**0**	**1**	**2**	**1**
CO3	**3**	**3**	**1**	**0**	**2**	**2**	**0**	**1**	**3**	**0**	**3**	**2**	**2**	**1**	**1**	**3**	**2**
CO4	**3**	**2**	**1**	**1**	**1**	**2**	**0**	**1**	**3**	**0**	**3**	**2**	**2**	**1**	**1**	**3**	**2**
CO5	**3**	**3**	**1**	**1**	**1**	**2**	**0**	**1**	**3**	**0**	**3**	**2**	**2**	**1**	**1**	**2**	**2**
CO6	**3**	**3**	**2**	**1**	**2**	**3**	**1**	**1**	**3**	**0**	**3**	**2**	**2**	**1**	**2**	**3**	**3**
CO7	**3**	**3**	**2**	**1**	**2**	**3**	**1**	**1**	**3**	**1**	**3**	**2**	**2**	**1**	**2**	**3**	**3**

CSS1C02 – ADVANCED DATA STRUCTURES

Objective: To introduce basic and advanced data structures dealing with algorithm development and problem solving.

Course Outcome

CO1: Summarize different categories of data structures and C programming RGDFconcepts.

CO2: Design algorithms to perform operations with linear and non – linear data DGDGstructures.

CO3: Describe how arrays, linked lists, stacks, queues, trees and graphs are represented in memory and used by algorithms.

CO4: Describe common applications for arrays, linked lists, stack, queue, tree and CVBgraphs.

CO5: Demonstrate different methods for traversing trees.

CO6: Design and implement an appropriate hashing function for an application.

CO7: Discuss the computational efficiency of the principal algorithms for sorting, BNNNsearching and hashing.

CO8: Describes various types of trees and heap structures.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I		**Introduction to Data Structures and C Programming**	**10**	**6**
	1	Data structure – definition – types & operations, characteristics of data structures.	3	
	2	Abstract Data Type (ADT) – algorithms – concepts – definition – objectives of algorithms – quality of an algorithm.	3	
	3	Space complexity and time complexity of an algorithm.	1	
	4	Introduction to C Programming – Decision making (if, if else, nesting of if else, else if ladder, switch, break, continue, goto, return). Loops (while, do-while, for, nesting, skipping & breaking). Arrays, Pointers, Structures & Union.	3	
II		**Linear Data Structures and Recursion**	**20**	**6**

	5	Counting Techniques – permutations and combinations, asymptotic behavior of functions.	5	
	6	Linear data structures – Arrays, records, representation, operations (traversing, inserting, deleting, sorting, searching), Sorting algorithms – linear & binary search – complexity.	5	
	7	Linked lists – operations & implementation. Stack – operations & implementations (array and linked list). Applications – parsing arithmetic expressions, conversion & evaluating expressions. Recursion – characteristics, types, applications, comparison of recursive & non-recursive algorithms.	5	
	8	Queue – operations & implementations (array & linked list). Circular queue – dequeue – priority queues. Recursive lists, heterogeneous lists, deterministic skip lists, doubly linked lists & circular lists, sparse matrix representation.	5	
III	**Non-linear Data Structures**		**20**	**6**
	9	Trees – terminology, tree traversals algorithms. Binary trees, threaded binary trees.	5	
	10	Binary Search Trees (BST) – traversals & operations. Heap Tree, balanced trees, M-way trees	5	
	11	B and B+ trees, Red-Black Tree, Digital Search Tree, Tries, Treaps	5	
	12	Huffman algorithm for extended binary tree – operations & implementation. Graphs – representation, operations, traversals & implementation.	5	
IV	**Hashing and Heaps**		**15**	**6**
	13	Hashing – overview, hash tables, hash functions & computations.	3	
	14	Open addressing – linear probing, quadratic probing, double hashing – algorithms & implementations.	4	
	15	Rehashing, extendable hashing, separate chaining.	4	

	16	Hashing efficiency – Heaps – overview, implementation & operations.	4	
V	**Advanced Heaps and Trees**		**10**	**6**
	17	Heap structures – Min-Max heaps, Deaps.	2	
	18	Leftist heaps, Binomial heaps.	4	
	19	Fibonacci heaps, Binary heaps, Skew heaps,Pairing heaps. Splay trees.	4	

References:

1. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, *Data Structures and Algorithms*, Addison-Wesley, ISBN: 978-0201000238.
2. Horowitz E and Sahni S, *Fundamentals of Data Structures*, Computer Science Press, ISBN: 9780716780427.
3. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, *Fundamentals of Data Structures in C*, Silicon Press, ISBN: 0929306406.
4. Richard F. Gilberg and Behrouz A. Forouzan, *Data Structures: A Pseudocode Approach With C*, Thomson Brooks/Cole Publications, Course Technology, ISBN: 9780534390808.
5. Aaron M. Tenenbaum, Yedidyah Langsam and Moshe J. Augenstein, *Data Structure using C*, Prentice- Hall, ISBN: 9780131997462.
6. Robert Kruse, Tondo C L and Bruce Leung, *Data Structures & Program Design in C*, Pearson India, 2nd Edition, ISBN: 9788177584233.
7. U. A. Deshpande and O. G. Kakde, *Data Structures & Algorithms*, ISTE Learning Materials Centre, New Delhi, ISBN: 9788188057054.
8. Thomas H Cormen, Charles E Leiserson, and Ronald L Rivest, *Introduction to Algorithms*, 3rd Edition, Prentice Hall of India Private Limited, New Delhi, ISBN: 978- 0262033848.
9. Seymour Lipschutz, *Data Structures With C*, 1st Edition, Tata Mcgraw Hill Education Private Limited, ISBN: 0070701989.
10. Jean-Paul Tremblay, Paul G. Sorenson, P. G. Sorenson, *Introduction to Data Structures with Applications*, 2nd Edition, Mcgraw-Hill College, ISBN: 0070651574.
11. E Balagruswamy, *Programming in ANSI C*, TMH, 5th Edition, ISBN: 0070681821.

Mapping of COs with POs and PSOs

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7
CO1	**3**	**3**	**2**	**1**	**0**	**2**	**0**	**1**	**3**	**1**	**3**	**2**	**1**	**1**	**0**	**2**	**1**
CO2	**3**	**2**	**1**	**0**	**1**	**2**	**0**	**1**	**3**	**0**	**3**	**2**	**1**	**0**	**1**	**2**	**1**
CO3	**3**	**3**	**1**	**0**	**2**	**2**	**0**	**1**	**3**	**0**	**3**	**2**	**2**	**1**	**1**	**3**	**2**
CO4	**3**	**2**	**1**	**1**	**1**	**2**	**0**	**1**	**3**	**0**	**3**	**2**	**2**	**1**	**1**	**3**	**2**
CO5	**3**	**3**	**1**	**1**	**1**	**2**	**0**	**1**	**3**	**0**	**3**	**2**	**2**	**1**	**1**	**2**	**2**
CO6	**3**	**3**	**2**	**1**	**2**	**3**	**1**	**1**	**3**	**0**	**3**	**2**	**2**	**1**	**2**	**3**	**3**
CO7	**3**	**3**	**2**	**1**	**2**	**3**	**1**	**1**	**3**	**1**	**3**	**2**	**2**	**1**	**2**	**3**	**3**
CO8	**3**	**3**	**2**	**1**	**2**	**3**	**1**	**1**	**3**	**1**	**3**	**2**	**2**	**1**	**2**	**3**	**3**

CSS1C03 – THEORY OF COMPUTATION

Objectives: To provide the students with an understanding of basic concepts in the theory of computation.

Course Outcome

CO1: Describe broad overview of the theoretical foundations of computer science.

CO2: Understand regular languages and finite automata.

CO3: Apply the concept of context free languages in problem solving.

CO4: Solve various problems of applying normal form techniques, push down automata and Turing Machines.

CO5: Propose solutions for the problems based on computability and decidability.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I		**Preliminaries**	**15**	**6**
	1	Introduction to formal proof and inductive proofs.	3	
	2	Central concepts of Automata Theory –Alphabets, Strings, Languages.	3	
	3	Introduction to Automata and Grammar – DFA and NFA.	4	
	4	Epsilon transitions in NFA – Equivalence of DFA and NFA with/without ε-moves.	5	
II		**Regular Languages**	**15**	**6**
	5	Regular Expressions – connection with Finite Automata.	3	
	6	Properties of Regular Languages – Pumping Lemma, non-regular languages.	3	
	7	Closure properties, homomorphism, substitution, Decision Properties.	4	
	8	Myhill Nerode theorem, DFA minimization, Regular Grammar.	5	
III		**Context-Free and Context-Sensitive Languages**	**15**	**6**
	9	Context Free Languages – Equivalence of CFG and PDA.	3	
	10	Normal Forms (CNF, GNF), Closure properties of CFL.	3	
	11	DCFLs – properties, Decision procedures, CYK algorithm.	4	

	12	Pumping lemma for CFLs, Context-sensitive languages – Equivalence of LBA and CSG.	5	
IV		**Turing Machines**	**15**	**6**
	13	Turing Machines – definition, computations.	3	
	14	Equivalence of TM variants – multi-tape, nondeterministic TM.	3	
	15	Turing acceptable, decidable, enumerable classes. Equivalence of Type-0 grammar and TM.	4	
	16	Church's Thesis, Chomsky hierarchy, closure properties of recursive & recursively enumerable languages.	5	
V		**Computability and Complexity**	**15**	**6**
	17	Halting problem and reductions.	3	
	18	Post Correspondence Problem.	3	
	19	Computational complexity – time and space bounded simulations, Classes P and NP.	4	
	20	NP-Completeness, Cook's theorem.	5	

References:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, *Introduction to Automata Theory*, Languages of Computation, 3rd Edition, Prentice Hall, ISBN: 0321455363.
2. Linz P, *An Introduction to Formal Languages and Automata*, Narosa Publishing House Pvt. Ltd., New Delhi, ISBN: 9788173197819.
3. Michael Sipser, *Introduction to Theory of Computation*, Cengage Learning India Private Limited, Indian Edition, ISBN: 8131505138.
4. H.R. Lewis and C.H. Papadimitriou, *Elements of Theory of Computation*, 2nd Edition, Prentice Hall, ISBN: 0132624788.
5. J. E. Savage, *Models of Computation*, *Exploring the Power of Computing*, Addison Wesley, 1998, Available at http://cs.brown.edu/~jes/book/.
6. Martin J.C, *Introduction to Languages and Theory of Computation*, Tata McGraw Hill, 3rd Edition, ISBN: 9780070660489.

Mapping of COs with POs and PSOs

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**	**PSO7**
CO1	3	3	2	1	0	2	0	1	3	1	3	2	1	1	0	2	1
CO2	3	3	2	1	1	2	0	1	3	0	3	2	2	1	1	3	2
CO3	3	3	2	1	2	2	0	1	3	0	3	2	2	1	1	3	2
CO4	3	3	2	1	2	3	1	1	3	0	3	2	2	1	2	3	3
CO5	3	3	2	1	2	3	1	1	3	1	3	2	2	1	2	3	3

CSS1C04 – OBJECT ORIENTED PROGRAMMING CONCEPTS

Objectives: To learn object-oriented concepts and programming concepts and methodologies and to learn its implementation using Java.

Course Outcome:

CO1: Recall the object-oriented programming concepts and basics of Java.

CO2: Design and implement object oriented programs including packages &interfaces.

CO3: Explain and handle exceptions and threads.

CO4: Develop interactive programs using applets, AWT and swings.

CO5: An introduction to Unified Modelling Language (UML).

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I		**Introduction to OOPS and Java Basics**	**15**	**6**
	1	Basic principles of Object Orientation – objects, attributes, methods, encapsulation, information hiding, state retention, object identity, messages, class hierarchy, inheritance, polymorphism, genericity.	3	
	2	Introduction to Java – history, versioning, JVM, byte code, features of Java.	3	
	3	Language components – primitive data types, comments, keywords, literals, variables, scope & declarations.	4	
	4	Control structures (FOR, IF, WHILE, DO WHILE, SWITCH, BREAK, CONTINUE), operators, casts & conversions, arrays.	5	
II		**Classes and Objects**	**15**	**6**
	5	Object-oriented programming concepts – classes and class fundamentals.	3	
	6	Declaring objects, new operator, methods, parameter passing.	3	
	7	Constructors, parameterized constructors, this keyword.	4	
	8	Finalize method and object lifecycle.	5	
III		**Advanced OOP Concepts**	**15**	**6**
	9	Method and constructor overloading, access controls, static and final.	3	
	10	Nested and inner classes.	3	
	11	Inheritance – extends, member access, super keyword.	4	

	12	Polymorphism – method overriding, dynamic method dispatch, abstract classes, packages, and interfaces.	5	
IV		**Exceptions, Threads and I/O**	**15**	**6**
	13	Java Exception Handling – basics, exception hierarchy, user-defined exception classes.	3	
	14	Streams – file and standard streams, stream classes & interfaces, byte and character streams.	3	
	15	Threads – processes vs. threads, creating threads, runnable interface, thread class.	4	
	16	Thread communication & synchronization.	5	
V		**Applets, AWT & Swing**	**15**	**6**
	17	Applets – applet class, types, skeleton, applet tag, passing parameters.	3	
	18	Event handling – delegation event model, event classes, listeners.	3	
	19	AWT – classes and window fundamentals, frames, fonts, graphics, colors, controls, layouts, menus, dialogue boxes.	4	
	20	Swing – applets, icons, labels, buttons, textbox, combo box, tables, panes.	5	

References:

1. Herbert Scheldt, *Java Complete Reference*, 8th Edition, Tata Mcgraw Hill Education Private Limited, ISBN: 1259002462.
2. E Balaguruswamy, *Programming in Java: A Primer*, 4th Edition, Tata Mcgraw Hill Education Private Limited, ISBN: 007014169X.
3. Kathy Sierra, *Head First Java*, 2nd Edition, Shroff Publishers and Distributors Pvt Ltd, ISBN: 8173666024.
4. David Flanagan, Jim Farley, William Crawford and Kris Magnusson, *Java Enterprise in a Nutshell: A Desktop Quick Reference*, 3rd Edition, O'Reilly Media, ISBN: 0596101422.
5. Grady Booch, James Rumbaugh and Ivar Jacobson, *The Unified Modeling Language User Guide*, 2nd Edition, Pearson, ISBN: 8131715825.

Mapping of COs with POs and PSOs

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**	**PSO7**
CO1	3	2	1	0	1	1	0	1	2	0	3	2	2	0	1	2	1
CO2	3	3	2	0	1	2	0	1	2	0	3	3	3	1	2	3	2
CO3	3	3	2	0	1	2	0	1	2	0	3	3	3	1	2	3	2
CO4	3	3	2	1	2	3	1	1	3	0	3	3	3	1	2	3	3
CO5	2	2	2	0	1	2	0	1	2	0	2	2	3	2	2	2	2

CSS1C05 – COMPUTER ORGANIZATION & ARCHITECTURE

Objectives: To familiarize with the digital fundamentals, computer organization, computer architecture and assembly language programming.

Course Outcome

CO1: Identify, understand and apply different number systems and codes.
CO2: Understand the digital representation of data in a computer system.
CO3: Understand the general concepts in digital logic design and their use in sequential and combinational circuit design.
CO4: Describe fundamental organization of a computer system.
CO5: Explain addressing modes, instruction formats and program control statements.
CO6: Understand computer arithmetic formulae and solve problems.
CO7: Distinguish the organization of various parts of a system memory hierarchy.
CO8: Identify and compare different methods for computer I/O.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I	**Number Systems, Logic & Digital Circuits**		**15**	**6**
	1	Number systems and conversions, Boolean Algebra, Truth Tables, Logic gates, Map simplification.	3	
	2	Flip-flops, design of combinational and sequential circuits, examples of digital circuits.	3	
	3	Adders, multiplexers, decoders, counters, shift registers.	4	
	4	Register transfer language & micro-operations, Data representation – data types, sign & magnitude, complements, fixed-point, floating-point, other binary codes, error detection codes.	5	
II	**Basic Computer Organization & Control Unit**		**15**	**6**
	5	Machine instructions – classification, function, addresses, size, addressing modes.	3	
	6	Instruction cycle, instruction sequencing, registers, register transfers, arithmetic/logic operations.	3	
	7	Memory read & write, execution of complete instruction, branch instruction, bus organization (single, two, three bus).	4	

	8	Control Unit – hardwired control, micro programmed control, microinstructions – types.	5	
III		**Arithmetic & Logic Unit**	**15**	**6**
	9	Addition of positive numbers, fast adders, signed addition & subtraction, addition/subtraction logic unit.	3	
	10	Multiplication of positive numbers – array multiplier, sequential multiplier.	3	
	11	Signed number multiplication, Booth's algorithm, fast multiplication, bit pair recoding.	4	
	12	Division – restoring & non-restoring algorithms, floating point numbers & operations.	5	
IV		**Memory and I/O Organization**	**15**	**6**
	13	Memory hierarchy, main memory (RAM, ROM) – organization, performance considerations.	3	
	14	Cache memory, virtual memory, memory management requirements. Memory hierarchy, main memory (RAM, ROM) – organization, performance considerations.	3	
	15	Secondary storage, memory interleaving, Input/output Organization – programmed I/O, interrupt I/O, interrupts (hardware, controllers, vectored interrupts.	4	
	16	Daisy chaining, nesting), Direct Memory Access (DMA) – operations & controller, I/O interfaces, I/O channels, I/O processors.	5	
V		**Microprocessor Architecture & Instruction Set**	**15**	**6**
	17	General 8-bit microprocessor architecture – 8085, functional block diagram, functions of sections.	3	
	18	Architecture of 8086 CPU.	3	
	19	Instruction sets – instruction format, addressing modes, instruction set of 8085 CPU.	4	
	20	Instruction cycle, timing diagrams, machine cycles, fetch & execute operations, estimation of execution time.	5	

References:

1. V Carl Hamacher, Zvonko Vranesic and Safwat Zaky, *Computer Organization*, Mc- Graw Hill International Edition, 5th Edition, ISBN: 9780071122184.
2. Morris Mano, *Digital Logic and Computer Design*, Prentice Hall of India, ISBN: 0876924178.
3. Morris Mano, *Computer System Architecture*, Prentice Hall, 3rd Edition. ISBN: 0131755633.
4. William Stallings, *Computer Organization and Architecture*, 9th Edition, Prentice Hall, ISBN: 013293633X.
5. Andrew S Tanenbaum, *Structured Computer Organization*, Prentice Hall, 6th Edition, ISBN: 0132916525.
6. Floyd Thomas L, *Digital Fundamentals*, Pearson Education, 10th Edition, Prentice Hall, ISBN: 0132359235.
7. Albert Paul Malvino, Donald P Leach, *Digital Principles and Applications*, McGraw Hill, 4th Edition, ISBN: 0070398836.
8. Thomas C Bartee, *Digital Computer Fundamentals*, McGraw Hill, 6th Edition, ASIN: B004H0SL5K.
9. Ramesh. S. Gaonkar, *Microprocessor Architecture*, Programming, and Applications with the 8085, 6th Edition, Wiley Eastern Ltd, New Delhi, ISBN: 9788187972884.
10. Mohamed Rafiquzzaman, *Introduction to Microprocessors and Microcomputer Based System Design*, 2nd Edition, CRC Press, ISBN: 9780849344756.
11. Muhammad Ali Mazidi, Janice Mazidi, Rolin Mckinlay, Janice M. Mazidi, Janice Gillispie Mazidi and Rolin D., *The 8051 Microcontroller and Embedded Systems*, Pearson Education Asia, 5th Indian Reprint, ISBN: 013119402X.

Mapping of COs with POs and PSOs

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7
CO1	3	2	1	0	0	1	0	1	3	0	3	2	1	0	1	2	1
CO2	3	2	1	0	0	1	0	1	3	0	3	2	1	0	1	2	1
CO3	3	3	1	0	1	2	0	1	3	0	3	2	2	1	1	3	2
CO4	3	3	2	0	1	2	0	1	3	0	3	2	2	1	2	3	2
CO5	3	3	2	0	1	2	0	1	3	0	3	2	2	1	2	3	2
CO6	3	3	2	0	1	3	0	1	3	0	3	2	2	1	2	3	2
CO7	3	3	1	0	1	2	0	1	3	0	3	2	2	1	2	3	2
CO8	3	3	1	0	1	2	0	1	3	0	3	2	2	1	2	3	2

CSS1L01 – PRACTICAL I

Objectives: To practically implement the theory portions covered in The Object Oriented Programming Concepts (CSS1C04) and Advanced Data Structures (CSS1C02).

Course Outcome

CO1: Use an integrated development environment to write, compile, run, and test simple object- oriented Java programs.

CO2: Understand Java programming concepts and utilize Java Graphical User Interface in program writing.

CO3: Design and develop Java programs that solve real-world problems.

CO4: Demonstrate the concepts of stack, queue and linked list and apply various operations on them.

CO5: Demonstrate the concept of tree traversal and its operations.

CO6: Design program based on the concepts of sorting and searching techniques.

Course Outline

Unit I: Object Oriented Programming Concepts

1. Simple Java programs like computing formulas expressions.
2. Programs involving loops and decisions like generating Fibonacci, prime, strange series.
3. Programs involving arrays.
4. Programs involving class and objects.
5. Illustrate method overloading.
6. Illustrate single level inheritance.
7. Illustrate multiple inheritances using interface.
8. String sorting, pattern matching etc.
9. Illustrate threads and thread priorities.
10. Exception handling (user-defined).
11. Illustrate usage of Applets like moving ball, face etc.

Unit 2: Data Structures and Algorithms

1. Implementation of stacks using arrays.
2. Implementation of queues, circular queue using arrays.
3. Implementation of sequential search and binary search techniques.

4. Implementation of linked lists and operations (add, insert, delete, search) on linked lists.
5. Implementation of stacks using linked list.
6. Implementation of queues using linked list.
7. Implementation of doubly linked list.
8. Implementation of circular linked list.
9. Implementation of binary tree and traversals.
10. Implementation of Binary search trees and perform the operations on BST.
11. Implementation of various sorting algorithms.
12. Conversion of an infix expression to the postfix form using stacks.
13. Evaluation of a postfix expression.
14. Implementation of graphs and graph traversals.
15. Implementation of heap tree and operations.

CSS1A01 – INTRODUCTION TO RESEARCH (ABILITY ENHANCEMENT AUDIT COURSE)

Objectives:

Large numbers of students are actively considering and taking up research and associated higher studies. An introductory course on research aims to introduce students to the important aspects of research. The intent of such a course is to make students aware of the details associated with formal research. By going through this introductory course on research, students are likely to be able to take up research activities in a more systematic and formal manner right from the beginning.

Course Outcome

CO1: Understand research terminology.

CO2: Apply the ethical principles of research.

CO3: Identify the components of a literature review process.

CO4: Critically analyze published research works.

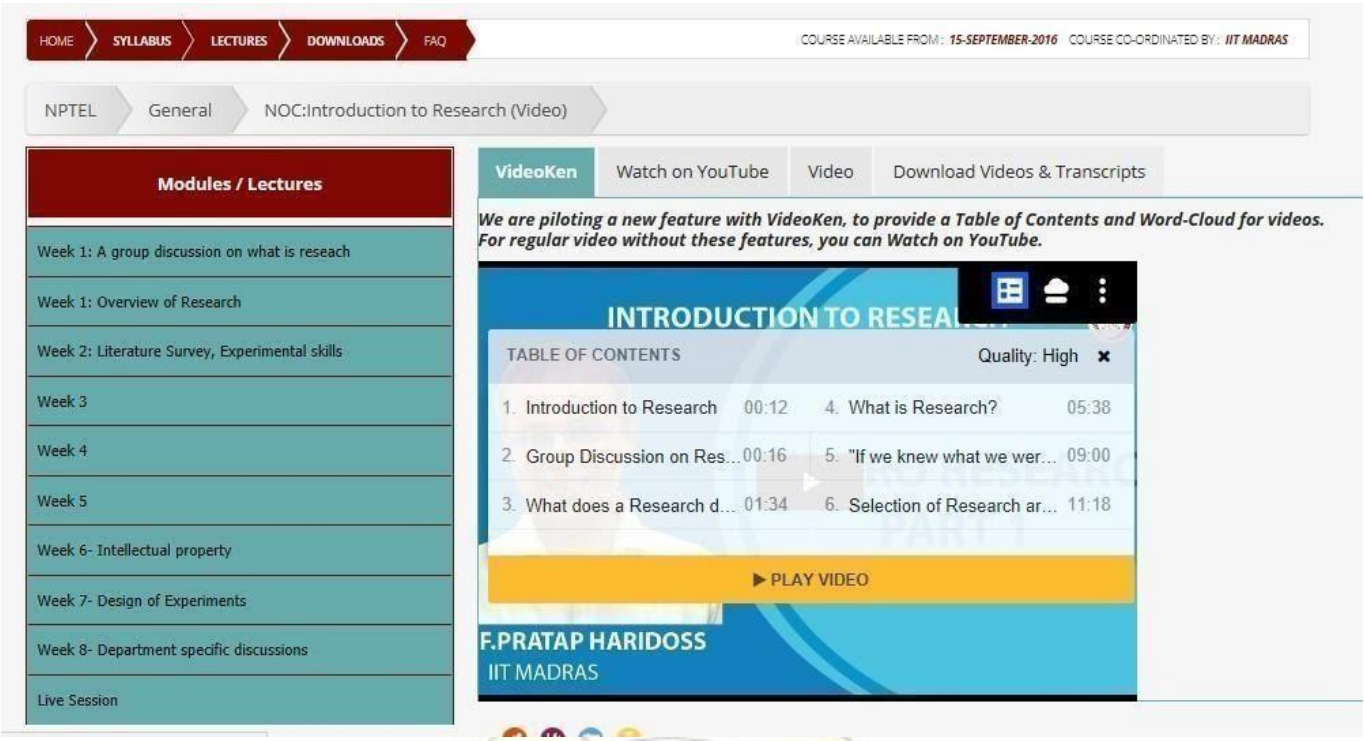
CO5: Innovate and apply research methods in the discipline of computing.

Course Evaluation & Course Credit

The Ability Enhancement Audit Course has 4 credits which will not be counted for evaluating the overall SGPA & CGPA. The College/Department shall conduct examination of 2 Hrs duration with a minimum of 20 weightage before the conclusion of first semester classes and have to intimate /upload the results of the same to the University on the stipulated date during the III Semester. Students have to obtain only minimum pass requirements in this Audit Course.

Course Delivery Mode

This course is an Ability Enhancement Audit Course. The course content is not delivered in the classrooms. Instead, the students have enrol themselves for the online course offered at NPTEL. The online course is available at https://nptel.ac.in/courses/121106007/. Students can either view the video module online or can download the video lessons and transcripts to view or read them offline.



Course Outline

The students are encouraged to cover the following modules of the course *Introduction to Research* from NPTEL:

- Week1: Overview of Research
- Week2: Overview of Literature Survey: Literature Survey using Web of Science, Literature Survey using Scopus, Writing Up, Tutorial on using BibTeX with LaTeX to add references to a document, Tutorial on using Microsoft Word with Bibliographic Sources, Tutorial on using Microsoft Word with endnote entries
- Week3: Data Analysis
- Week4: How to make Technical presentation – Technical Writing
- Week 6: Intellectual property
- Week8: Research in Computer Science & Engineering

References:

1. Video Lessons and Transcripts available (including in the regional language) at https://nptel.ac.in/courses/nptel_download.php?subjectid=121106007.

SEMESTER II

CSS2C06 – DESIGN AND ANALYSIS OF ALGORITHMS

Objectives:

- To introduce the concept of an algorithmic approach for solving real-life problems.
- To teach basic principles and techniques of computational complexity.
- To familiarize with parallel algorithms and related techniques.

Course Outcome

CO1: Design algorithms in context of space and time complexity and apply asymptotic notation.

CO2: Analyse the problem and develop the algorithms related to these problems.

CO3: Classify the problems and apply the appropriate design strategy to develop algorithms.

CO4: Analyse the problem and develop the algorithms related to these problems.

CO5: Demonstrate the use of parallel algorithms.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I		**Algorithm Design and Problem Types**	**15**	**6**
	1	Algorithm design – Introduction, steps in developing algorithm, methods of specifying an algorithm.	3	
	2	Decisions prior to designing – based on device capabilities, nature of solutions, suitable data structures.	3	
	3	Model of Computation – RAM model and PRAM model.	4	
	4	Important Problem Types – Sorting, Searching, String processing, Graph problems, Combinatorial problems, Geometric problems, Numerical problems.	5	
II		**Algorithm Design Techniques**	**15**	**6**
	5	Brute Force approach – String matching.	3	
	6	Divide-and-Conquer approach – Merge sort.	3	
	7	Branch-and-Bound technique – Knapsack problem, Greedy approach – Kruskal's & Prim's Algorithms.	4	
	8	Dynamic Programming – Longest Common Subsequence, Backtracking – Sum of subsets problem.	5	

III	**Algorithm Analysis**		**15**	**6**
	9	Importance of algorithm analysis, Time and Space complexity.	3	
	10	Growth of Functions – Asymptotic notations, Cost estimation, Big-O, Big-Ω, Little-o, Little-Ω, Big-Θ notations.	3	
	11	Ratio theorems – Big O, Big Ω, Big Θ Ratio Theorems.	4	
	12	Analyzing Algorithm Control Structures, Solving Recurrences – Iteration, Substitution, Recursion Tree, Master's Theorem (Cases 1–3), Analysis of Strassen's algorithm & Merge sort.	5	
IV	**Complexity Theory**		**15**	**6**
	13	Complexity classes – P, NP, NP-Hard, NP-Complete problems, NP-Completeness reductions – Travelling Salesman Problem.	8	
	14	NP-Completeness reductions – Hamiltonian Cycle, P vs NP problem.	7	
V	**Parallel Algorithms**		**15**	**6**
	15	Analyzing parallel algorithms – Time complexity, cost, number of processors.	3	
	16	Space complexity, speedup, efficiency, scalability, Amdahl's Law.	3	
	17	Parallel merging and sorting.	4	
	18	Euler tour technique, Parallel prefix computation, Deterministic symmetry breaking.	5	

References:

1. Thomas H Cormen, Charles E Leiserson, and Ronald L Rivest, *Introduction to Algorithms*, 3rd Edition, Prentice Hall of India Private Limited, New Delhi, ISBN: 9780262033848 (Unit I, II, III and IV).

2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, *The Design and Analysis of Computer Algorithms*, 1st Edition. Addison Wesley, ISBN: 0534915728 (Unit I, II, III and IV).

3. Pallaw, V K, *Design and Analysis of Algorithms*, Asian Books Private Ltd, 2012, ISBN: 8184121687 (Unit I, II, III and IV).

4. Sanjay Razdan, *Fundamentals of Parallel Computing*, Narosa Publishing House, 2014, ISBN: 9788184873481 (Unit V).

5. Pandey H M, *Design and Analysis of Algorithms*, University Science Press, 2013, ISBN: 9788131803349 (Unit I, II, III and IV).

6. Upadhyay N, *Design and Analysis of Algorithms*, SK Kataria & Sons, 2008 (Unit I, II, III and IV).

7. U. Manber, *Introduction to Algorithms: A Creative Approach*, Addison Wesley, ISBN: 9780201003277 (Unit I, II, III and IV).

8. Gilles Brassard and Paul Bratley, *Fundamentals of Algorithmics*, Prentice-Hall of India, ISBN: 0133350681 (Unit I, II, III and IV).

9. Goodman S E and Hedetniemi, *Introduction to the Design and Analysis of Algorithms*, Mcgraw Hill, ISBN: 0070237530 (Unit I, II, III and IV).

10. Horowitz E and Sahni S, *Fundamentals of Computer Algorithms*, Galgotia Publications Pvt. Ltd, ISBN: 8175152575 (Unit I, II, III and IV).

11. Oded Goldreich, P, *NP and NP - Completeness*, Cambridge University Press, 2011. ISBN: 0521122546 (Unit V).

12. Donald Knuth, *The Art of Computer Programming, Fundamental Algorithms*, Volume 1, Addison Wesley, 1997, ISBN: 8177587544 (Unit I).

13. Sanjeev Arora and Boaz Borak, *Computational Complexity - A Modern Approach*, Cambridge University Press; 2009, ISBN: 0521424267 (Unit III).

Mapping of COs with POs and PSOs

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**	**PSO7**
CO1	3	3	1	0	0	2	0	1	3	0	3	2	1	0	1	2	1
CO2	3	3	2	0	1	2	0	1	3	0	3	2	2	1	1	3	2
CO3	3	3	2	0	2	2	0	1	3	0	3	2	2	1	2	3	2
CO4	3	3	2	0	1	3	0	1	3	0	3	2	2	1	2	3	2
CO5	3	3	2	0	2	3	0	1	3	0	3	2	2	1	2	3	3

CSS2C07 – OPERATING SYSTEM CONCEPTS COURSE

Objectives:

- Understand the role and purpose of an operating system as a resource manager.
- Learn the structure and functions of various operating system components.
- Explore concepts such as processes, threads, CPU scheduling, synchronization, memory management, file systems, and I/O systems.
- Analyze different strategies for process coordination, deadlock handling, and system security.

Course Outcome

CO1: Understand the basic components of a computer operating system.
CO2: Compare and interpret the applications of Process and threads.
CO3: Describe the policies for scheduling, deadlocks, synchronization, system calls, and file systems.
CO4: Illustrate the functioning of process management, memory management and file management Modules present in an OS.
CO5: Differentiate various types of scheduling algorithms.
CO6: Understand the concepts of Three-Tier Client/Server Architecture, Middleware and the characteristics of mobile operating systems.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I		**Operating System Overview & Processes**	**15**	**6**
	1	Operating System overview – Objectives, functions, evolution, major achievements.	3	
	2	Process description and control – Processes, creation & termination, Five-state model, suspended process.	3	
	3	Process control – Modes of execution, process creation, process and mode switching.	4	
	4	Threads – Processes vs Threads, multithreading, thread states, types of threads, multicore and multithreading.	5	
II		**Concurrency & Deadlock**	**15**	**6**
	5	Concurrency principles – Race conditions, OS concerns, process interaction, competition & cooperation.	3	
	6	Mutual exclusion – Requirements, hardware support, semaphores, producer-consumer problem.	3	

	7	Monitors, message passing, readers/writers problem.	4	
	8	Deadlock – Principles, prevention, avoidance, detection, recovery, Dining Philosophers problem.	5	
III	**Memory Management**		**15**	**6**
	9	Memory management concepts – Address binding, logical vs physical address space, dynamic loading, dynamic linking, shared libraries.	5	
	10	Overlays and swapping, Contiguous memory allocation.	5	
	11	paging, segmentation, virtual memory, Demand paging, page replacement algorithms, thrashing.	5	
IV	**Scheduling**		**15**	**6**
	12	Uniprocessor scheduling – Types, criteria, non-preemptive vs preemptive scheduling.	3	
	13	Scheduling algorithms – FCFS, SJF.	3	
	14	Scheduling algorithms – Priority, RR, Multilevel, Feedback Queue.	4	
	15	Multiprocessor scheduling – Classification, granularity, design issues; Thread scheduling; Real-time scheduling – Background, characteristics, deadline scheduling, rate monotonic scheduling, priority inversion.	5	
V	**Distributed & Mobile Operating Systems**		**15**	**6**
	16	Client/Server computing – Definition, applications, classes.	3	
	17	Three-tier client/server architecture, middleware.	3	
	18	Service-Oriented Architecture – Distributed message passing, remote procedure calls, clusters.	4	
	19	Mobile operating systems – Characteristics, comparative study of IOS and Android features.	5	

References

1. William Stallings, *Operating System- Internals and Design Principles*, 7th Edition, Pearson, ISBN: 9780273751502.
2. Abraham Silberschatz, Peter B. Galvin and, Greg Gagne, *Operating System Concepts*, 9th Edition, John Wiley & Sons TISBN: 9781118063330.
3. Ann McIver McHone's and Ida M. Flynn, *Understanding Operating Systems*, 6th Edition, Cengage Learning, 2010, ISBN: 9781439079201.
4. Mukesh Singhal and Niranjan G. Shivaratri, *Advanced Concepts in Operating Systems - Distributed, Database, and Multiprocessor Operating Systems*, Tata McGraw- Hill Education Private Limited, ISBN: 9780070575721.
5. Current Literature (for Mobile Operating Systems).

Mapping of COs with POs and PSOs

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**	**PSO7**
CO1	3	2	1	0	0	2	0	1	3	0	3	2	1	0	1	2	1
CO2	3	3	2	0	1	2	0	1	3	0	3	2	2	1	1	3	2
CO3	3	3	2	0	1	3	0	1	3	0	3	2	2	1	2	3	2
CO4	3	3	2	0	2	3	0	1	3	0	3	2	2	1	2	3	2
CO5	3	3	2	0	1	2	0	1	3	0	3	2	2	1	2	2	2
CO6	3	2	2	0	2	2	0	1	3	0	3	2	2	1	2	2	3

CSS2C08 – COMPUTER NETWORKS

Objectives:

- To provide the student with a top-down approach of networking starting from the application layer.
- To introduce computer networking in the backdrop of Internet protocol stack.

Course Outcome

CO1: Understand the basics concepts of computer network organization and implementation.
CO2: Describe theoretical understanding of layered network models - OSI and TCP/IP Models.
CO3: Illustrate the functionalities of different network layers.
CO4: Analyse the network application such as data transmission between client and server, file transfer, real-time and multimedia transmission.
CO5: Explain the security aspects in networks and principles of cryptography.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I		**Introduction to Computer Networks**	**15**	**6**
	1	Basics of computer networks – Introduction, topologies, categories of networks.	3	
	2	Internetwork and Internet – network modes, layered models – OSI & TCP/IP models.	3	
	3	Transmission media – wired and wireless media.	4	
	4	Computer networks and Internet – network edge, core, network access, delay & loss, protocol layers & services, history of networking and Internet.	5	
II		**Application Layer Protocols**	**15**	**6**
	5	Principles of application layer protocols.	5	
	6	Web and HTTP, FTP, Email in Internet – DNS.	5	
	7	Socket programming – building a web server, content distribution.	5	
III		**Transport & Network Layer Services**	**15**	**6**
	8	Transport layer – introduction, relationship between transport & network layers, UDP.	3	
	9	Reliable data transfer, TCP – features and working.	3	

	10	Congestion control, Network layer services – routing, IP, routing in Internet, routers.	4	
	11	IPv6, multicast routing, mobility.	5	
IV		**Link Layer Services**	**15**	**6**
	12	Link layer introduction – error detection & correction.	3	
	13	Multiple access protocols, LAN addressing, ARP.	3	
	14	Ethernet – hubs, bridges, switches.	4	
	15	Wireless links, PPP, ATM.	5	
V		**Network Security**	**15**	**6**
	16	Security in networks – introduction, principles of cryptography.	5	
	17	Authentication and integrity, Key distribution and certification	5	
	18	Firewalls – attacks and countermeasures.	5	

References:

1. J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach Featuring Internet*, 6th Edition, Perason Education, ISBN: 0132856204.
2. Behrouz Forouzan, *Data Communications and Networking*, 4th Edition, McGraw-Hill Reprint, ISBN: 0073250325.
3. Peterson L.L. and Davie B .S., *Computer Networks, A Systems Approach*, 5th Edition, Morgan Kaufmann, ISBN: 9780123850591.
4. Keshav, *An Engineering Approach to Computer Networking*, Pearson Education Asia, ISBN: 97898123598652000.
5. Andrew S. Tanenbaum, *Computer Networks*, 5th Edition, PHI, ISBN: 9788131787571.
6. Herbert Scheldt, *Java Complete Reference*, 7th Edition, McGraw-Hill Osborne Media, ISBN: 9780072263855.

Mapping of COs with POs and PSOs

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7
CO1	3	2	1	0	1	2	0	1	3	0	3	2	1	0	1	2	2
CO2	3	3	1	0	1	2	0	1	3	0	3	2	1	1	1	2	2
CO3	3	3	2	0	1	3	0	1	3	0	3	2	2	1	2	3	2
CO4	3	3	2	0	2	3	0	1	3	0	3	3	2	1	2	3	3
CO5	3	2	1	0	1	3	0	1	3	0	3	2	1	1	1	3	2

CSS2C09 – COMPUTATIONAL INTELLIGENCE

Objectives: To introduce concepts of Artificial Intelligence and Machine Learning.

Course Outcome:

CO1: Apply the basic principles, models, and algorithms of AI to recognize, model, and solve problems in the analysis and design of information systems.

CO2: Conceptualize various knowledge representation techniques.

CO3: Analyze the problem-solving methods and algorithms related to searching, reasoning, game playing and machine learning.

CO4: Understand the functioning of expert systems and its importance.

CO5: Demonstrate the implementation various AI algorithms to solve real life problems.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I	**Introduction to AI & Problem Solving**		**15**	**6**
	1	Introduction – Definition of AI, problems, scope, and applications.	3	
	2	Problem space and search – production system & characteristics.	3	
	3	Predicate calculus – inference rules, structures.	4	
	4	Strategies for state space search – using state space to represent reasoning with predicate calculus.	5	
II	**Heuristic Search**		**15**	**6**
	5	Control and implementation of state space search – generate and test.	3	
	6	Heuristic methods – hill climbing, best-first search.	3	
	7	Advanced search – problem reduction, constraint satisfaction, means-ends analysis.	4	
	8	Heuristic in games – complexity issues.	5	
III	**Knowledge Representation**		**15**	**6**

	9	Knowledge representation issues – representation and mappings, representing simple facts in logic.	3	
	10	Representing instances and ISA relationships, computable functions, predicates, resolution, natural deduction.	3	
	11	Knowledge representation using rules – logic programming, forward vs backward reasoning.	4	
	12	Symbolic reasoning under uncertainty – non-monotonic reasoning, DFS & BFS in reasoning.	5	
IV	**Game Playing & Expert Systems**		**15**	**6**
	13	Game playing – minimax search procedure, alpha-beta cut-offs, refinements, iterative deepening.	3	
	14	Planning systems – components of planning, understanding & constrained satisfaction.	3	
	15	Slot and filler structures – semantic nets, frames, conceptual dependency, scripts.	4	
	16	Expert systems – definition, characteristics, knowledge representation & domain knowledge, expert system shells, lifecycle, tools, MYCIN & DENDRAL examples.	5	
V	**Machine Learning**		**15**	**6**
	17	Learning paradigms – rote learning, learning by advice, learning in problem solving.	3	
	18	Learning from examples – explanation-based learning, analogy, formal learning theory.	3	
	19	Connectionist models – Hopfield networks, neural networks, backpropagation.	4	
	20	Evolutionary & society-based learning – genetic algorithm, classifier systems, genetic programming, artificial life, society-based learning.	5	

References:

1. Elaine Rich, Kevin Knight and Shivshankar B. Nair, *Artificial Intelligence*, 3rd Edition, Tata - McGraw Hill, New Delhi, ISBN: 0070087709.
2. V S Janakiraman, K Sarukesi and P Gopalakrishnan, *Foundations of Artificial Intelligence and Expert System*, Macmillan India Limited, ISBN: 0333926250.
3. Stuart Russell and Peter Norvg, *Artificial Intelligence: A Modern Approach*, 3rd Edition, Prentice Hall, ISBN: 0136042597.'
4. G. F. Luger and W.A Stubblefield, *Artificial Intelligence - Structures and Strategies for Complex Problem Solving*, Addison-Wesley, 6th Edition, ISBN: 9780321545893.
5. P. H. Winston, *Artificial Intelligence*, Addison-Wesley, 3rd Edition, ISBN: 0201533774.
6. Nils J. Nilsson, Artificial Intelligence, A New Synthesis, 1st Edition, Morgan Kaufmann Publishers, Inc, ISBN: 1558604677.

Mapping of COs with POs and PSOs

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**	**PSO7**
CO1	3	3	2	0	2	3	0	1	3	0	3	3	2	1	2	3	2
CO2	3	3	2	0	1	2	0	1	3	0	3	2	2	1	1	2	2
CO3	3	3	2	0	2	3	0	1	3	0	3	3	2	1	2	3	3
CO4	3	2	1	0	1	2	0	1	2	0	2	2	1	1	1	2	2
CO5	3	3	2	0	2	3	0	1	3	0	3	3	2	1	2	3	3

CSS2C10 – PRINCIPLES OF COMPILERS

Objectives: To introduce the fundamental concepts and various phases of compiler design.

Course Outcome

CO1: Understand the major phases of compilation, identify tokens of a typical high -level programming language, define regular expressions for tokens, design and implement a lexical analyzer.

CO2: Develop the parsers and experiment the knowledge of different parsers design without automated tools.

CO3: Construct the intermediate code representations and generation.

CO4: Explain the role of different types of runtime environments and memory organization for implementation of typical programming languages.

CO5: Apply the optimization techniques to have a better code for code generation.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I		**Introduction & Lexical Analysis**	**15**	**6**
	1	Introduction to compiling – compiler, translator, interpreter, analysis of source program, phases of compiler, compiler construction tools.	3	
	2	Applications of compiler technology – programming language basics.	3	
	3	Lexical analysis – role of lexical analyser, input buffering, specification of tokens, recognition of tokens.	4	
	4	Regular expressions & finite automata – NFA to DFA, Regular Expression to NFA, design of lexical analyzer generator	5	
II		**Syntax Analysis**	**15**	**6**
	5	Syntax analysis – role of parser, error handling & recovery, parsing definitions, top-down & bottom-up parsing.	3	
	6	Context free grammars – derivations, parse tree, ambiguity, associativity & precedence of operators, grammar writing.	3	
	7	Top-down parsing – recursive descent parsing, FIRST & FOLLOW, LL(1) Grammars, recursive predictive parsing.	4	

	8	Bottom-up parsing – reductions, handle pruning, shift reduce parsing, operator precedence parsing, simple LR parsing.	5	
III	**Intermediate Code Generation & Type Checking**		**15**	**6**
	9	Intermediate code – DAG, three address code, addresses & instructions, quadruples, triples, SSA form.	3	
	10	Types & declarations – type expressions, type equivalences, declarations, type checking rules.	3	
	11	Advanced type handling – type conversion, function/operator overloading, type inference, polymorphic functions.	4	
	12	Control flow translation – boolean expressions, short circuit code, flow control statements (if, while, break, continue, goto).	5	
IV	**Run-Time Environments**		**15**	**6**
	13	Storage organization – static vs dynamic allocation, stack allocation of space.	3	
	14	Activation records – activation trees, calling sequences, access to non-local data.	3	
	15	Heap management – memory manager, memory hierarchy, locality in programs.	4	
	16	Reducing fragmentation – manual deallocation requests, issues with nested procedures.	5	
V	**Code Generation & Optimization**		**15**	**6**
	17	Code generation basics – design issues, target language, target machine model, program & instruction costs.	3	
	18	Addressing in target code – static & stack allocation, run-time addresses for names.	3	
	19	Basic blocks & flow graphs – representation, introduction to code optimization.	4	
	20	Data flow analysis & region-based optimization – reaching definitions, live variable analysis, available expressions, regions & hierarchies.	5	

References:

1. V Aho A, Ravi Sethi, D Ullman J, *Compilers Principles*, Techniques and Tools, 2nd Edition, Pearson Education Singapore Pte Ltd, ISBN: 8131721019.
2. K. V. N. Sunitha, *Compiler Construction*, Pearson, ISBN:9789332500297.
3. W Appel and Andrew, *Modern Compiler Implementation in C*, 1st Edition,Cambridge University Press, ISBN: 817596071X.
4. Allen I Holub, *Compiler Design in C*, 1st Edition, PHI Learning Pvt Ltd, ISBN: 812030778X.
5. Tremblay and Sorenson, *The Theory and Practice of Compiler Writing*, 1st Edition, BSP Books Pvt Ltd, ISBN: 8178000776.
6. Torben Ægidius Mogensen, *Basics of Compiler Design*, Department of Computer Science, University of Copenhagen (Online Edition).

Mapping of COs with POs and PSOs

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7
CO1	3	2	1	0	0	2	0	1	3	0	3	2	1	0	1	2	1
CO2	3	3	2	0	1	2	0	1	3	0	3	2	2	1	1	2	2
CO3	3	3	2	0	1	3	0	1	3	0	3	3	2	1	2	3	2
CO4	3	2	1	0	1	3	0	1	3	0	3	2	2	1	2	3	2
CO5	3	3	2	0	1	3	0	1	3	0	3	3	2	1	2	3	3

CSS2L02 – PRACTICAL II

Objectives: To practically implement the theory portions covered in the courses Operating System Concepts (CSS2C07) and Computer Networks (CSS2C08)

Course Outcome

CO1: Discuss and formulate the problems based on the basic principles of networks.

CO2: Implementation of different memory management techniques in OS.

CO3: Implement various system operations of the operating system and also the various process scheduling algorithms.

CO4: Understand the TCP/IP configuration for Windows and Linux.

CO5: Design and implement various network applications such as data transmission between client and server, file transfer, real-time multimedia transmission.

CO6: Understand different Linux/UNIX shell scripts and execute various shell programs.

Course Outline

Unit I: Computer Networks

1. Design a LAN with a given set of requirements. The design should include topology, hardware and software requirements like cable, connectors, hubs/switches/bridges, interface cards along with a budget for the LAN. (Faculty in charge should give the requirements to the students)*.
2. Establish a LAN that consists of at least one server and two clients*.
3. Implementation of TCP Client.
4. Implementation of TCP Server.
5. Write a program to check the Date and Time in TCP Date Time Client.
6. Write a program to check the Date and Time in TCP Date Time Server.
7. Implementation of UDP client and server.
8. Write a program to transfer Files using UDP.
9. Implementation of transferring files using FTP.
10. Write a program to simulate the sliding window protocol.

**These questions are NOT meant for examination purpose. However, Viva questions can be asked based on these experiments.*

Unit II: Operating System Concepts

1. Write programs using the following system calls: fork(), excel() and wait().
2. Write File System Calls to write, append and display.
3. To accept the burst time for a set of processes for FCFS scheduling and create chart consisting of the burst time, turnaround time and wait time of each process.

4. To accept the burst time for a set of processes for SJF scheduling and create chart consisting of the burst time, turnaround time and wait time of each process.
5. To accept the burst time and priority for a set of processes for Priority scheduling and create chart consisting of the burst time, priority, turnaround time and wait time of each process.
6. To create n Fibonacci numbers and prepare a list of prime numbers amongst them (use pipe for IPC).
7. To demonstrate IPC using shared memory.
8. To allocate memory requirements for processes using best fit allocation- Accept n processes with their memory requirements and *n* holes with their sizes. Perform memory allocation using Best Fit algorithm. Display a chart consisting of the process and the allocated hole.
9. To accept n processes with their memory requirements and n holes with their sizes. Perform memory allocation using First Fit algorithm. Display a chart consisting of the process and the allocated hole.
10. To demonstrate the process of contiguous allocation of memory blocks to store files of varying sizes.
11. To implement Producer Consumer problem using semaphores.

CSS2A02 – TERM PAPER

(PROFESSIONAL COMPETENCY AUDIT COURSE)

Objectives:

- To introduce the student to the techniques of literature survey.
- To acquaint him/her with the process of presenting his/her work through seminars and technical reports.

Course Outcome:

CO1: Apply critical thinking skills analytical ability in problem solving.

CO2: Apply foundational research skills to address research problem.

CO3: Innovate, experiment and analyse research findings.

CO4: Demonstrate capacity to lead and manage change through a collaborative environment.

CO5: Innovate, experiment and analyse research findings and practice the process of scientific publishing.

Course Outline

The student is expected to do an extensive literature survey and analysis in an area related to computer science, chosen by him/her, under the supervision of a faculty member from the department. The student has to choose an area for his/her work after due consultation and approval from the guide. The study should preferably result in a critical review of the present works/design ideas/designs/algorithms/theoretical contributions in the form of theorems and proofs/new methods of proof/new techniques or heuristics with analytical studies/implementations and analysis of results.

The student should give a seminar on his/her work, during the semester, and submit a technical report. Technical report should be prepared in TEX in IEEE conference style format.

Course Delivery Mode

Students are given choice to opt for the supervisor according to his/her area of interest. The Department Council will finally decide and distribute the students among the faculty members by accommodating the choice and interest of the students, as far as possible. The faculty in charge must give proper directions and guidance to the students in carrying out the literature review effectively and systematically.

Course Evaluation & Course Credit

The Professional Competency Audit Course has 4 credits which will not be counted for evaluating the overall SGPA & CGPA. The Department shall conduct the final evaluation

of the course based on the following criteria and have to intimate /upload the results of the same to the University on the stipulated date during the III Semester.

Component	Weightage
Publication of the Review Paper in a UGC Listed, Peer Reviewed or other peer reviewed refereed Journals	20% (Maximum weightage be given to UGC listed Journal and weightage be reduced in other cases)
Presentation in an International/ National/ Regional Conference	20% (Maximum weightage be given to International Conferences with Proceeding having ISBN and weightage be reduced in other cases)
Quality of the Technical Report	40%
Quality and Effectiveness of the Report Presentation	20%

Students have to obtain only minimum pass requirements in this Audit Course.

Reference:

Articles from ACM/IEEE/INFLIBNET Journals/Conference Proceedings and/or equivalent documents, standard textbooks and web-based material, approved by the supervisor.

SEMESTER III

CSS3C11 – ADVANCED DATABASE MANAGEMENT SYSTEM

Objectives:

- To understand the relational model, and know how to translate requirements captured in an Entity-Relationship diagram into a relational schema.
- To reason about dependencies in a relational schema.
- To understand normal form schemas, and the decomposition process by which normal forms are obtained.
- To familiarize with advanced SQL' statements.
- To understand advanced features of database technologies.

Course Outcome

CO1: Explain the basics of database management system, concepts of relational data model, entity-relationship model, relational database design, relational algebra and calculus.

CO2: Apply the normalization techniques to improve the database design.

CO3: Describe various database manipulation commands in SQL.

CO4: Understand Transaction Processing & Locking using the concept of Concurrency control.

CO5: Conceptualize advanced features of Object-Oriented Database Management Systems and Distributed databases.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I	**Database Fundamentals**		**15**	**6**
	1	Introduction – purpose of DBMS, views of data (abstraction, instances, schemas), data independence.	3	
	2	Data models – hierarchical, network, relational; ER diagram. DB languages – DDL, DML, transaction & storage management, DBA, users, overall system structure.	3	
	3	Relational data model – concepts, keys, integrity constraints (domain, key, entity, referential).	4	
	4	ER data model – constraints, keys, design issues, ER diagram, weak entities, extended ER features, schema	5	

		reduction to tables. Relational algebra (selection, projection, joins, division) & relational calculus (tuple & domain).		
II	**Relational Database Design**		**15**	**6**
	5	Database anomalies, functional dependency, lossless join, dependency-preserving decomposition.	5	
	6	Normalization – 1NF ,2NF, 3NF and BCNF.	5	
	7	Advanced normalization – multivalued dependencies (4NF), join dependency (5NF), domain-key normal form (DKNF).	5	
III	**Relational Query Languages & SQL Programming**		**15**	**6**
	8	Basics of QBE and SQL – DDL, data types, creation, insertion, deletion, modification, renaming, dropping tables. Data constraints – PK, FK, unique, I/O constraints, ALTER TABLE.	3	
	9	SQL DML – computations, SELECT, logical operators, range search, pattern matching, GROUP BY, HAVING.	3	
	10	Joins – multiple tables, self-joins. DELETE, UPDATE. Views – creation, renaming, destroying. Procedural SQL – SET, SELECT, IF, IF/ELSE, WHILE, GOTO, global variables.	4	
	11	Advanced SQL – Security (locks, levels), cursors, error handling, stored procedures (CREATE, ALTER, DROP), UDFs, triggers, multiple trigger interactions.	5	
IV	**Transactions & Concurrency**		**15**	**6**
	12	Transactions – concept, states, ACID properties.	3	
	13	Concurrency – serializability (conflict, view), recoverability (recoverable schedules, non-cascading, strict).	3	
	14	Concurrency control schemes – locking, two-phase locking, deadlock, granularity.	4	
	15	Timestamp ordering protocol, basics of query processing.	5	
V	**Advanced Databases**		**15**	**6**
	16	OODBMS – concepts, need, composite objects.	3	
	17	Issues in OODBMS, advantages & disadvantages.	3	

	18	Distributed DB – motivation, concepts, types of distribution, architecture.	4	
	19	Distributed DB design, distributed transactions, commit protocols.	5	

References:

1. Elmasri and Navathe, *Fundamentals of Database Systems*, 5th Edition, Pearson, ISBN: 9788131758984.
2. Abraham Silbersehatz, Henry F. Korth and S.Sudarshan, *Database System Concepts*, 6th Edition, Tata McGraw-Hill, ISBN: 0071325220.
3. CJ Date, *An Introduction to Database Systems*, 8th Edition, Addison Wesley, ISBN: 0321197844.
4. Ramakrishnan and Gehrke, *Database Management Systems*, 3rd Edition, McGraw - Hill Education, ISBN: 9339213114.
5. Alexis Leon and Mathews Leon, *Database Management Systems*, 1st Edition, Vikas Publishers, ISBN: 8182092221.
6. Vikram Vaswani, *MySQL The complete Reference*, 1st Edition, Tata Mcgraw Hill Education Private Limited, ISBN: 0070586845.
7. Joel Murach, *Murach's Mysql*, Mike Murach & Associates Inc, ISBN: 9350237695. 8. Paul DuBois, *MySQL Cookbook*, 2nd Edition, O'Reilly Media, ISBN: 8184042809.

Mapping of COs with POs and PSOs

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7
CO1	3	2	1	0	1	2	0	1	3	0	3	2	2	1	1	2	2
CO2	3	3	1	0	1	2	0	1	3	0	3	2	2	1	2	2	2
CO3	3	2	2	0	1	2	0	1	3	0	3	3	2	1	2	3	2
CO4	3	3	2	0	1	3	0	1	3	0	3	3	2	1	2	3	2
CO5	3	2	1	0	2	2	0	1	3	0	3	2	2	1	2	3	3

CSS3C12 – PYTHON PROGRAMMING CONCEPTS

Objectives: To learn object-oriented concepts and programming concepts and methodologies and to learn its implementation using Python .

Course Outcome:

CO1: Recall the object-oriented programming concepts and basics of Python.

CO2: Design and implement object-oriented programs.

CO3: Understand the basic concepts and definition of Data Science, and describe its evolution over time.

CO4: Apply basic data pre-processing techniques such as data cleaning, integration, and transformation.

CO5: Introduction to Unified Modelling Language (UML).

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I		**Python Basics**	**15**	**6**
	1	Review of Python - Interactive Mode, Script Mode, Data Types.	3	
	2	Types, Functions in Python, Sequential Statement, Selective Statements.	3	
	3	Looping Statements, String and String Methods.	4	
	4	List and List Methods, Tuple and Tuple Methods, Dictionary and Dictionary Methods.	5	
II		**Classes and Inheritance in Python**	**15**	**6**
	5	Classes in Python- Name spaces and scope rules, Define classes (attributes, methods), Use _ init_(), Importance of "self".	3	
	6	Create instance objects, distinguish between class attributes and instance attributes, add methods dynamically, Access attributes and methods, Use built-in class attributes (dict, doc, name, module, bases).	3	
	7	Use __del__() and __str__() in a class, Data hiding, Static methods, Destroy objects (garbage collection).	4	
	8	Inheritance-Concept of Inheritance, Types of Inheritance, Single Inheritance, Use super() in derived class to invoke	5	

		init(),Multiple Inheritance, Overriding methods, Override methods of parent class, Learn about abstract methods.		
III	**Introduction to Data Science**		**15**	**6**
	9	Introduction to Data Science – Definition, Evolution, Roles, Applications.	3	
	10	Data Collection and Data Pre-Processing - Data Collection Strategies, Data Pre-Processing Overview.	3	
	11	Data Cleaning, Data Integration and Transformation.	4	
	12	Data Reduction and Discretization.	5	
IV	**Data Model Development & Evaluation**		**15**	**6**
	13	Data Model Development and Evaluation- Simple and Multiple Regression.	3	
	14	Model Evaluation using Visualization, Residual plot and distributional plot.	3	
	15	Model Evaluation using Visualization, Residual plot and distributional plot.	4	
	16	Model Evaluation techniques- Supervised learning techniques, unsupervised learning techniques.	5	
V	**UML and System Analysis**		**15**	**6**
	17	Introduction to Unified Modelling Language (UML), UML diagrams.	5	
	18	Diagrams, class diagrams, object interaction diagrams, state and activity diagrams, component diagrams, deployment diagrams.	5	
	19	Introduction to analysis - object oriented system analysis, design and implementations.	5	

References:

1. Python: The Complete Reference 4th Edition, McGraw Hill Education (2018) 2. E Balaguruswamy, *Programming in Introduction To Computing And* ISBN-10: 9789387572942.
Problem Solving Using Python,Tata Mcgraw Hill India (2016), ISBN: 9789352602582.

3. Grady Booch, James Rumbaugh and Ivar Jacobson, *The Unified Modeling Language User Guide*, 2nd Edition, Pearson, ISBN: 8131715825.

Mapping of COs with POs and PSOs

	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PS O1**	**PS O2**	**PS O3**	**PS O4**	**PS O5**	**PS O6**	**PS O7**
CO 1	3	2	1	1	0	0	0	2	2	0	3	2	2	0	0	2	0
CO 2	3	3	2	2	2	2	0	2	3	2	3	3	3	0	2	3	3
CO 3	2	2	2	0	3	3	3	2	3	0	3	3	2	2	2	3	2
CO 4	2	3	0	0	2	3	0	2	3	0	3	3	2	0	2	3	2
CO 5	3	2	2	2	3	2	2	2	2	1	3	2	3	0	3	3	3

CSS3C13 – PRINCIPLES OF SOFTWARE ENGINEERING

Objectives:

- To develop familiarity with software engineering principles and practices.
- To have an understanding about the process of product/literature survey, techniques of problem definition, and methods of report writing.

Course Outcome

CO1: Understand the software process and development models.
CO2: Understand the software design process and structured analysis of systems.
CO3: Distinguish different types of modelling like DFD and UML.
CO4: Illustrate the knowledge about the design of user interface.
CO5: Apply the skill of project management and report preparation.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I		**Introduction to Software Engineering**	**15**	**6**
	1	Introduction – problem domain, software engineering challenges, approaches.	3	
	2	Software process and development models – waterfall, iterative, spiral, V-model.	3	
	3	Agile models – principles, Scrum, XP, Kanban.	4	
	4	Software Development Life Cycle (SDLC) – phases, importance, comparison of models.	5	
II		**Requirements Analysis & Software Design**	**15**	**6**
	5	Requirements – feasibility study, types of feasibility, problem analysis.	3	
	6	Software requirements – requirement specification, functional specification, metrics.	3	
	7	Software design – outcomes, cohesion & coupling, layered module arrangement, structured analysis (DFD, real-time DFD extension).	4	
	8	Object-Oriented Modelling – use case model, class diagram, interaction diagram, activity diagram, data diagram, state chart diagram, ER diagram.	5	

III	**UI Design, Planning & Testing**		**15**	**6**
	9	User Interface (UI) design – characteristics, concepts, types, component-based GUI development.	3	
	10	UI design methodology, process planning, cost estimation, project scheduling.	3	
	11	Configuration management, risk management, coding practices, review, documentation.	4	
	12	Software testing – basics, test plan steps, strategies (unit, integration, system, acceptance).	5	
IV	**Project Management & Research Preparation**		**15**	**6**
	13	Project management – time management, setting aims and objectives.	3	
	14	Techniques for generating ideas, literature survey basics.	3	
	15	Information sources – primary, secondary, tertiary sources.	4	
	16	Writing a literature survey – structure, critical review, referencing styles.	5	
V	**Project Story & Communication**		**15**	**6**
	17	Project story preparation – key deliverables, milestones.	3	
	18	Communicating with experts – formal and informal forms of communication.	3	
	19	Presenting ideas – effective presentation methods, common problems faced by research scholars.	4	
	20	Report writing – structure of project report, style, formatting, best practices.	5	

References:

1. Pankaj Jalote, *An Integrated Approach to Software Engineering*, 3rd Edition, Narosa Publishing House, ISBN: 9788173197024.
2. Rajib Mall, *Fundamentals of Software Engineering*, 3rd Edition, PHI Learning Pvt Ltd, ISBN: 9788120338197.
3. Rohit Khurana, *Software Engineering: Principles and Practices*, 2nd Edition, Vikas Publishing House Pvt Ltd, ISBN: 8125939466.
4. Andy Hunt, *Your Research Hunt, How to Manage it*, Routledge, ISBN: 0415344085.

5. Michael Jay Polonsky, David S. Waller, *Designing and Managing a Research Project: A Business Student's Guide*, Sage, ISBN: 1412977754.

6. Richard Bullock, Maureen Daly Goggin and Francine Weinberg, *The Norton Field Guide to Writing (with Readings and Handbook)*, 3rd Edition, W. W. Norton & Company, ISBN: 0393919595.

7. Kavadia Garg, Agrawal and Agrawal, *An introduction to Research Methodology*, Rbsa Publishers ISBN: 8176111651.

Mapping of COs with POs and PSOs

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7
CO1	3	2	1	0	1	2	0	1	3	0	3	2	2	1	1	2	2
CO2	3	3	2	0	1	2	0	1	3	0	3	2	2	1	2	2	2
CO3	3	3	2	0	1	2	0	1	3	0	3	2	3	1	2	3	2
CO4	3	2	2	0	1	2	0	1	3	0	3	3	2	1	2	3	2
CO5	3	3	2	0	2	2	0	1	3	0	3	2	2	1	3	3	3

CSS3L03 – PRACTICAL III

Objectives: To practically implement the theoretical aspects covered in Advanced Database Management System (CSS3C11) and Python Programming Concepts (CSS3C12).

Course Outcome

CO1: Design and development of relational database systems.
CO2: Understand various advanced queries execution such as relational constraints, joins, set operations, aggregate functions, trigger and views.
CO3: Apply various software to design and build ER Diagrams, UML, Flowchart for related database systems.
CO4: Design and implement database applications on their own.
CO5: Analyse the irregularities present in the data and perform data cleaning
CO6: Use an integrated development environment to write, compile, run, and test simple object- oriented Python programs.
CO7: Create prediction models using supervised techniques.
CO8: Design and develop Python programs that solve real-world problems.

Course Outline:

Unit I: Advanced Database Management System

1. Creating database tables and using data types (create table, modify table, drop table).
2. Data Manipulation (adding data with INSERT, modify data with UPDATE, deleting records with DELETE).
3. Implementing the Constraints (NULL and NOT NULL, primary key and foreign key constraint, unique, check and default constraint).
4. Retrieving Data Using SELECT (simple SELECT, WHERE, IN, BETWEEN, ORDERED BY, DISTINCT and GROUP BY).
5. Aggregate Functions (AVG, COUNT, MAX, MIN, SUM).
6. String functions.
7. Date and Time Functions.
8. Use of union, intersection, set difference.
9. Implement Nested Queries & JOIN operation.
10. Performing different operations on a view.
11. Stored Procedure Programming - Simple Procedures - decision making - Loops - Error handlers - Cursors - Functions - Triggers - Calling Stored Procedure from Triggers.

Unit II: Python Programming Concepts

12. Simple Python programs like computing formulas expressions.

13. Program to display the function of different operators.
14. Write a Program to print prime number using python.
15. Program to swap two numbers.
16. Find area and perimeter of rectangle , square and triangle.
17. Find sum and average of 5 numbers.
18. Convert degree to Fahrenheit.
19. Program to convert hour to minutes.
20. Program to calculate simple interest and compound interest.
21. Write a Python program that predicts the value of Y based on a given value of X using linear regression. Use some sample data for training.

22.Create a Convolutional Neural Network (CNN) in Python to classify NFNFNNimages of cats and dogs. Train it using a small custom dataset and test BBBCBCthe accuracy.

CSS3E01a – COMPUTER GRAPHICS

Objectives:

- To understand the fundamentals of the modern computer graphics.
- To pipeline the mathematics of affine transformations in three dimensions.
- To understand the common data structures to represent and manipulate geometry, colour and light representation and manipulation in graphics systems. ▪ To have an exposure to programming in Open GL.

Course Outcome

CO1: Understand the basics of computer graphics, different graphics systems and applications of computer graphics.

CO2: Extract scene with different clipping methods and its transformation to graphics display device.

CO3: Explore projections and visible surface detection techniques for display.

CO4: Explore object representations and surface detection methods.

CO5: Understand techniques and OpenGL programming concepts.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I		**Introduction to Computer Graphics**	**15**	**6**
	1	Introduction – Applications of computer graphics.	3	
	2	Video display devices – refresh CRT, raster & random scan display, color CRT.	3	
	3	Flat panel displays – LCD, LED, DVST.	4	
	4	Raster-scan systems – video controller, display processor; Random-scan systems.	5	
II		**2D Graphics & Viewing**	**15**	**6**
	5	Line drawing algorithms – DDA, Bresenham's. Midpoint Circle drawing algorithm.	3	
	6	Filling algorithms – Scan-line polygon fill, boundary fill, flood fill.	3	

	7	2D Transformations – translation, rotation, scaling, shearing, reflection, composite transformations.	4	
	8	2D Viewing – viewing pipeline, window-to-viewport transformation; Clipping – point clipping, Cohen-Sutherland, Sutherland-Hodgeman polygon clipping, text clipping.	5	
III	**3D Graphics**		**15**	**6**
	9	3D transformations – translation, rotation.	3	
	10	scaling, shearing, reflection.	3	
	11	3D Viewing – viewing pipeline, viewing coordinates.	4	
	12	3D Projections – parallel & perspective projections	5	
IV	**3D Object Representation & Surface Detection**		**15**	**6**
	13	3D object representation – wireframe model, curve representation.	3	
	14	Surfaces – spline representation, Bezier curves, cubic splines	3	
	15	Visible surface detection – classification & back-face detection.	4	
	16	Z-buffer algorithm & other methods.	5	
V	**Discrete Techniques & OpenGL**		**15**	**6**
	17	Discrete techniques – texture mapping, bit & pixel operations.	3	
	18	Discrete techniques – compositing, sampling, aliasing techniques.	3	
	19	Introduction to OpenGL – features, operations, abstractions (GL, GLU, GLUT).	4	
	20	OpenGL programming – basic examples & applications.	5	

References:

1. Donald Hearn and M. Pauline Baker, *Computer Graphics*, 2nd Edition, Prentice Hall, ISBN: 0135309247.
2. Donald D. Hearn, M. Pauline Baker and Warren Carithers, *Computer Graphics with Open GL*, 4th Edition, Prentice Hall, ISBN: 9780136053583.
3. Hill, *Computer Graphics using OpenGL*, 3rd Edition, Prentice Hall of India Private Ltd. New Delhi, ISBN: 8120338294.

4. Mason Woo, Jackie Neider, Tom Davis, Dave Shreiner, Dave Shriner and Tom David, *Open GL Programming Guide*, 6th Edition, Person, ISBN: 9780201604580.

5. *The Official Guide to Learning OpenGL*, Version 1.1, Available at http://www.glprogramming.com/red/.
6. Shreiner and Angel, *Interactive Computer Graphics: A Top-Down Approach with Shader-Based OpenGL*, 6th Edition, Pearson Education, ISBN: 0132545233.

Mapping of COs with POs and PSOs

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7
CO1	3	2	1	0	2	2	0	2	2	0	3	2	2	0	1	3	2
CO2	3	3	0	0	2	2	0	2	3	0	3	3	2	0	2	3	2
CO3	3	3	0	0	2	2	0	2	3	0	3	2	2	0	1	3	2
CO4	3	3	0	0	2	2	0	2	3	0	3	2	2	0	1	3	2
CO5	3	3	1	0	2	2	0	2	3	0	3	3	3	0	2	3	3

CSS3E02c – CRYPTOGRAPHY AND NETWORK SECURITY

Objectives:

- To be familiar with classical and modern encryption and decryption techniques and apply in the security system.
- To understand various aspects of network security standards.

Course Outcome

CO1: Understand the fundamentals of cryptography.
CO2: Describe data integrity, authentication, digital signatures.
CO3: Analyse different network security applications
CO4: Familiarize standard algorithms that provide confidentiality, integrity and authenticity.
CO5: Understand network security technologies.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I	**Computer Security Concepts & Cryptography**		**15**	**6**
	1	Computer security concepts – challenges, attacks, services, mechanisms, model for network security.	3	
	2	Cryptography – symmetric encryption principles, cryptanalysis, Feistel cipher structure.	3	
	3	Symmetric block encryption algorithms – DES, Triple DES, AES.	4	
	4	Random & pseudorandom numbers – stream cipher & RC4, cipher block modes of operation.	5	
II	**Message Authentication & Public Key Cryptography**		**15**	**6**
	5	Message authentication – approaches, MAC.	3	
	6	Hash functions – one-way hash, secure hash functions, MACs.	3	
	7	Public key cryptography – principles, algorithms.	4	
	8	Digital signatures – concepts, applications.	5	
III	**Network Security Applications**		**15**	**6**
	9	Symmetric key distribution using symmetric encryption.	3	
	10	Kerberos version 4 – authentication mechanism.	3	
	11	Key distribution using asymmetric encryption – X.509 certificates, PKI.	4	

	12	Federated identity management – concepts & implementation.	5	
IV	**Transport & IP Security**		**15**	**6**
	13	Web security considerations – SSL & TLS overview.	3	
	14	SSL architecture – record protocol, change cipher spec protocol, handshake protocol.	3	
	15	Transport Layer Security – HTTPS, SSH.	4	
	16	IP Security – overview, policy, ESP, combining security associations, Internet Key Exchange (IKE).	5	
V	**Intrusion, Malware & Firewalls**		**15**	**6**
	17	Intruders – types, intrusion detection, password management.	3	
	18	Malicious software – types, viruses, worms, DDoS, countermeasures.	3	
	19	Firewalls – need, characteristics, types, firewall basing, configuration.	4	
	20	Advanced firewall concepts – DMZ networks, VPNs, distributed firewalls.	5	

References:

1. William Stallings, Network Security Essentials Applications and Standards, 4th Edition, Pearson India, ISBN: 8131761754.
2. William Stallings, Cryptography and Network Security: Principles and Practice, 6th Edition, Pearson India, ISBN: 9332518777.
3. Atul Kahate, Cryptography and Network Security, 3rd Edition, Tata McGraw-Hill Publishing, ISBN: 9789332900929.
4. Eric Maiwald, Fundamental of Network Security, 1st Edition, Tata McGraw - Hill Education, 0071070931.
5. Charlie Kaufman, Radia Perlman and Mike Speciner, Network Security: Private Communication in Public WorJd, 2nd Edition, PHI Learning Pvt Ltd, ISBN: 8120322134.

Mapping of COs with POs and PSOs

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**	**PSO7**
CO1	3	2	0	0	2	2	0	2	3	0	3	2	2	0	1	3	2
CO2	3	3	1	2	2	2	0	2	3	1	3	2	2	0	2	3	2
CO3	3	3	1	1	2	3	0	2	3	1	3	3	2	1	2	3	3
CO4	3	3	0	0	2	3	0	2	3	0	3	3	2	0	1	3	2
CO5	3	3	1	1	2	3	1	2	3	1	3	3	2	1	2	3	3

SEMESTER IV

CSS4P01 – PROJECT WORK

Objectives:

- To give a practical exposure to the process of software development life cycle.
- To develop a quality software solution by following the software engineering principles and practices. Students are also encouraged to take up a research-oriented work to formulate a research problem and produce results based on its implementation/simulation/experimental analysis.

Course Outcome:

CO1: Demonstrate a depth of knowledge of modern technology.

CO2: Practice to communicate effectively and to present ideas clearly and coherently to specific audiences in both the written and oral forms.

CO3: Understand the project requirements, reflect on their learning and take appropriate actions to implement it.

CO4: Estimate, plan, calculate, and adjust project variables.

CO5: Understand the importance of iteration, evaluation and prototyping in design of a software system.

Course Outline

Major project work is to be done individually by each student, under the guidance of a faculty member of the concerned department.

Guide has to constantly monitor the works done by the student, imparting him/her the necessary inputs for the successful completion of the project work. Students can either take up real-life application-oriented project work or research and development projects. The student can formulate a project problem with the help of her/his guide and submit the project proposal of the same. Approval of the project proposal is mandatory. If approved, the student can commence working on it, and complete it.

Guidelines for Submission of Report

The distinguishing mark of a dissertation is an original contribution to knowledge. The dissertation is a formal document whose sole purpose is to prove that you have made an original contribution to knowledge. Failure to prove that you have made such a contribution generally leads to failure.

It is a test of the student's ability to undertake and complete a sustained piece of independent research and analysis / application development, and to write up the work in a coherent form according to the rules and conventions of the

academic community. The role of the supervisor too is very crucial in this context.

A satisfactory dissertation should not only be adequate in its methodology, in its analysis and in its argument, and adequately demonstrate its author's familiarity with the relevant literature; it should also be written in correct, coherent language, in an appropriate style, correctly following the conventions of citation. It should, moreover, have a logical and visible structure and development that should at all times assist the reader understand the arguments being presented. The layout and physical appearance of the dissertation should also conform to university standards.

The dissertation is to be prepared in TEX format (either Latex or a suitable Windows TEX variant). The format of the report is included in Appendix A. Students are also encouraged to present their work in IT fest/conference/workshop/journal with the assistance and guidance of the supervisor. This should pave as a good start for the student in the art of publishing/presenting his/her work to the outside world. Due weightage is accommodated for publications out of the project work in the final evaluation.

CSS4E03c – SYSTEM SECURITY

Objectives: To provide an understanding of the differences between various forms of computer security, where they arise, and appropriate tools to achieve them.

Course Outcome

CO1: Familiarize with different types of securities in information systems, security goals and CIA.

CO2: Illustrate computer system threats and various types of system attacks .

CO3: Identify different issues associated with system attacks and how attacking occurs; and various types of attackers.

CO4: Provide knowledge in operating system security, file protections, security assurance .

CO5: Understand important elements of Database security .

CO6: Define security planning, various types of security policies and risk analysis.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I		**Security Concepts & Attacks**	**15**	**6**
	1	Notion of different types of securities – information security, computer security, security goals. Relation between security, confidentiality, integrity, availability, authorization.	3	
	2	Vulnerabilities – principles of adequate protection. Operating, database, program, and network security.	3	
	3	Security attacks – threats, vulnerabilities, controls. Problems – interception, interruption, modification, fabrication.	4	
	4	Computer criminals – amateurs, crackers, career criminals. Defence methods – control, hardware controls, software controls, effectiveness.	5	
II		**Program Security & Malicious Code**	**15**	**6**
	5	Secure programs – fixing faults, unexpected behaviour, types of flaws.	3	
	6	Non-malicious program errors – buffer overflows, incomplete mediation.	3	
	7	Viruses & malicious code – types, attachment, gaining control, prevention, examples (Brain virus, Internet worm, web bugs).	4	

	8	Targeted malicious code – trapdoors, Salami attack. Controls against threats – development controls, peer reviews, hazard analysis.	5	
III	**Operating System Security**		**15**	**6**
	9	Protected objects & protection methods – memory address protection (fence, relocation, base/bounds registers, tagged architecture, segmentation, paging).	3	
	10	Access control – directories, ACLs, file protection mechanisms, permissions.	3	
	11	Authentication – basics, passwords, challenge-response, biometrics.	4	
	12	Trusted OS – security policies, models, multilevel security, access security, limitations, OS design (features, assurance, flaws, methods).	5	
IV	**Database Security**		**15**	**6**
	13	Security requirements – integrity, confidentiality, availability	3	
	14	Reliability and integrity – sensitive data and interfaces.	3	
	15	Multilevel database concepts – challenges and solutions.	4	
	16	Proposals for multilevel security in DBMS.	5	
V	**Security Administration & Policies**		**15**	**6**
	17	Security planning – team members, commitments, business continuity plans.	3	
	18	Risk analysis – nature of risk, steps, arguments for & against risk analysis.	3	
	19	Organizational security policies – purpose, goals, audience, characteristics of good policy.	4	
	20	Nature of security policies – data sensitivity policy, government IT security policy, physical security (natural disasters, vandals, interception).	5	

References

1. C. P. Pfleeger and S. L. Pfleeger, Security in Computing, 4th Edition, Pearson India, ISBN: 9788131727256.
2. Matt Bishop, Computer Security: Art & Science, 1st Edition, Pearson, ISBN: 0201440997.
3. William Stallings, Cryptography and Network Security: Principles and Practice, 6th Edition, Pearson India, ISBN: 9332518777.

4. Michael E. Whitman and Herbert J. Mattord, Principles of Information Security, 4th Edition, Ceneage Learning India Pvt Ltd, ISBN: 8131516458.

Mapping of COs with POs and PSOs

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**	**PSO7**
CO1	3	2	0	2	2	2	0	2	3	0	3	2	2	0	1	3	2
CO2	3	3	0	2	2	2	0	2	3	0	3	2	2	0	2	3	2
CO3	3	3	1	2	2	2	0	2	3	1	3	2	2	1	2	3	2
CO4	3	3	0	2	2	3	0	2	3	0	3	2	2	0	2	3	2
CO5	3	2	0	2	2	2	0	2	3	0	3	2	2	0	2	3	2
CO6	3	3	1	3	2	3	0	2	3	1	3	2	2	1	3	3	3

CSS4E04a – DIGITAL IMAGE PROCESSING

Objectives: To be familiar with processing of the images, recognition of the pattern and their applications.

Course Outcome

CO1: Understand the fundamental concepts of a digital image processing

CO2: Apply various image enhancement techniques

CO3: Describe various image enhancement techniques

CO4: Implement algorithms for handling intensive image restoration problems.

CO5: Identify and compare various image segmentation and representation techniques

CO6: Understand various image compression procedures.

Detailed Syllabus

Module	Unit	Content	Hrs	Weightage
I		**Digital Image Fundamentals**	**15**	**6**
	1	Introduction – digital image representation, fundamental steps in image processing.	3	
	2	Elements of digital image processing systems.	3	
	3	Digital image fundamentals – elements of visual perception, simple image model.	4	
	4	Sampling and quantization, pixel relationships, image geometry.	5	
II		**Image Transforms**	**15**	**6**
	5	Introduction to image transforms – Fourier transform.	3	
	6	Discrete Fourier Transform (DFT) and its properties.	3	
	7	Other separable image transforms – Walsh, Hadamard, Discrete Cosine Transform (DCT).	4	
	8	Hotelling transform.	5	
III		**Image Enhancement**	**15**	**6**
	9	Basic grey-level transformations, histogram equalization, image subtraction, image averaging.	5	
	10	Spatial filtering – smoothing filters, sharpening filters, Laplacian filters.	5	

	11	Frequency domain enhancement – smoothing, sharpening, homomorphic filtering.	5	
IV		**Image Restoration & Segmentation**	**15**	**6**
	12	Image degradation/restoration model, noise models.	3	
	13	Restoration techniques – inverse filtering, least mean square filtering.	3	
	14	Constrained least mean square filtering.	4	
	15	Edge detection, thresholding, region-based segmentation, boundary representation.	5	
V		**Image Compression**	**15**	**6**
	16	Fundamentals of image compression, compression models, information theoretic perspective	3	
	17	Lossless compression techniques – Huffman coding, arithmetic coding.	3	
	18	Lossless methods – bit-plane coding, run-length coding.	4	
	19	Lossy compression – transform coding, image compression standards.	5	

References

1. Richard E Woods and Rafael C Gonzalez, *Digital Image Processing*, 3rd Edition, Pearson Education Singapore Pte Ltd, ISBN: 8131726959.
2. B. Chanda and D.D. Majumder, *Digital Image Processing and Analysis*, 2nd Edition, PHI Learning Pvt Ltd, ISBN: 8120343255.
3. A.K. Jain, *Fundamentals of Digital Image Processing*, 2nd Edition, PHI Learning Pvt Ltd, ISBN: 8120309294.
4. W.K. Pratt, *Digital Image Processing: PIKS Scientific Inside*, 4th Edition, John Wiley, ISBN: 0471767778.
5. Milan Sonka, Vaclav Hlavac and Roger Boyle, *Image Processing Analysis and Machine Vision*, 3rd Edition, Ceneage Learning India Pvt Ltd, ISBN: 8131518833.

Mapping of COs with POs and PSOs

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7
CO1	3	2	1	1	1	2	1	2	3	1	3	2	2	1	1	2	1
CO2	3	3	1	1	1	2	1	2	3	1	3	3	2	1	1	3	2
CO3	2	2	2	1	1	2	1	2	2	1	2	2	2	1	1	2	1
CO4	3	3	1	1	2	3	1	2	3	1	3	3	2	1	2	3	2
CO5	3	3	1	1	2	3	1	2	3	1	3	3	2	1	2	3	2
CO6	3	2	1	1	1	2	1	2	3	1	3	2	2	1	1	3	1

IV

APPENDIX A

Guidelines for Project Report & Layout

Cover Page & First Page

<TITLE> A PROJECT REPORT

SUBMITTED BY

<NAME OF THE STUDENT>

FOR THE AWARD OF THE

DEGREE OF MASTER OF SCIENCE (M.Sc.)

COMPUTER SCIENCE

UNIVERSITY OF CALICUT

<COLLEGE EMBLEM>

<NAME OF THE DEPARTMENT>

<NAME OF THE INSTITUTION>

(AFFILIATED TO THE UNIVERSITY OF CALICUT)

<ADDRESS>

MONTH, YEAR

Acknowledgement

ACKNOWLEDGEMENT

Date: Name of the Student

Declaration by the Student

DECLARATION

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person or material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Date: Signature:

Name :

Reg.

No.:

Certificate from Guide & HoD

CERTIFICATE

This is to certify that the project report entitled <<TITLE HERE>> submitted by <<Name of the Student>> (Register Number: << Reg, No>>) to University of Calicut for the award of the degree of Master of Science (M.Sc.) in Computer Science is a bonafide record of the project work carried out by him/her under my supervision and guidance. The content of the report, in full or parts have not been submitted to any other Institute or University for the award of any other degree or diploma.

Signature	Signature
<Name Project Guide>	<Name of the HOD>
<Designation>	<Designation>

Place:

Date:

PROJECT EVALUATION REPORT OF THE EXAMINERS

Certified that the candidate was examined by us in the Project VivaVoce Examination held on ..... and his/her Register Number is .....

Examiners:

1.

2.

Contents

CONTENTS

Abstract

List of Figures <<Page No>>
List of Tables <<Page No>>

1 <<Chapter Name>> <<Page No>>

1.1 <<Section Name>> <<Page No>>

1.2 <<Section Name>> <<Page No>>

1.2.1 <<Sub-Section Name>> <<Page No>>

Abstract

ABSTRACT

The abstract is a very brief summary of the report's contents. It should be about half a page long. Somebody unfamiliar with your project should have a good idea of what it's about having read the abstract alone and will know whether it will be of interest to them.

An abstract is a section at the beginning of a report, dissertation, thesis or paper summarizing the contents, significant results and conclusions of said document. It allows people to rapidly ascertain the documents purpose and if the document will be useful for them to read.

The abstract is not the same as a summary in the sense you are think of. It is a standalone account of the document giving purpose of the work (objectives), method used, scope of the work, results, conclusions and recommendations.

The abstract, although it comes first logistically, always should be written at the completion of the other chapters of the project report. It needs to be written last because it is the essence of your report, drawing information from all of the other sections of the report. It explains why the experiment was performed and what conclusions were drawn from the results obtained.

A general guideline for an abstract has five sections or areas of focus: why the experiment was conducted; the problem being addressed; what methods were used to solve the problem; the major results obtained; and the overall conclusions from the experiment as a whole.

Do not be misled, however, from this list into thinking that the abstract is a long section. In fact, it should be significantly shorter than all of the others. All of this information should be summarized in a clear but succinct manner if the abstract is going to be successful. An estimated average length for all of this information is only a single paragraph. Although this may seem as though it is a short length to contain all of the required information, it is necessary because it forces you to be accurate and yet compact, two essential qualities.

There are many useful web pages such as http://writing2.richmond.edu/training/proiect/ biology /abslit.html to get few sample abstracts and the common mistakes we make when we write an abstract.

List of Figures

LIST OF FIGURES

Figure 1.1: <<Figure title>> <<Page No>>

Figure 1.2: <<Figure title> <<Page No>>

…

Figure 2.1: <<Figure title>> <<Page No>> Figure 2.2: <<Figure title>> <<Page No>>

…

List of Tables

LIST OF TABLES

Table 1.1: <<Table title>> <<Page No>>
Table 1.2: <<Table title>> <<Page No>>

…

Table 2.1: <<Table title>> <<Page No>>

Table 2.2: <<Table title>> <<Page No>>

CHAPTER 1

INTRODUCTION

This is a general introduction about the project. Briefly summarize the relevance and background information about the proposed work. It should have the following sections.

1. About the proposed work, underlying technologies and techniques – outline briefly the technological/engineering/scientific/socioeconomic/relevance or significance of the project work being reported.
2. Project Profile – Tile, Area and Category and other relevant information.
3. About the Organization – to whom the Project Work is carried out.
4. Major Contributions of the Project Work.

CHAPTER 2

PROBLEM DEFINITION AND METHODOLOGY

This chapter is meant for giving a detailed description about the problem. This chapter includes the following subsections.

1. Problem Definition
2. Objectives
3. Motivation
4. Methodology
5. Scope

CHAPTER 3

REQUIREMENT ANALYSIS AND SPECIFICATION

This chapter includes the following subsections.

1. Requirement Analysis/Literature Review
2. Existing System
3. Proposed System
4. Requirement Specification
    a. Functional Requirements
    b. Non-functional Requirements
    c. Environmental Details (Hardware & Software Requirements)
5. Feasibility Study
    a. Technical Feasibility
    b. Economical Feasibility
    c. Operational Feasibility
6. Project Planning and Scheduling
    a. PERT Chart
    b. GANTT Chart
7. Software Requirement Specifications (IEEE format preferred)

CHAPTER 4

REQUIREMENT ANALYSIS AND SPECIFICATION

This chapter includes the following subsections.

1. Users of the System
2. Modularity Criteria
3. Architecture Diagrams (whichever of the following if applicable)
   a. DFD
   b. UML Diagrams
   c. Flowchart
4. User Interface Layout
5. Structure of Reports Being Created
6. Database Design
   a. List of Entities and Attributes
   b. E R Diagram
   c. Structure of Tables

IMPLEMENTATION

This chapter is about the realization of the concepts and ideas developed earlier. It can also describe any problems that may have arisen during implementation and how you dealt with them.

Do not attempt to describe all the code in the system, and do not include large pieces of code in this section. Instead pick out and describe just the pieces of code which, for example:

- Are especially critical to the operation of the system;
- You feel might be of particular interest to the reader for some reason;
- Illustrate a non-standard or innovative way of implementing an algorithm, data structure, etc.

You should also mention any unforeseen problems you encountered when implementing the system and how and to what extent you overcame them. Common problems are:

- Difficulties involving existing software, because of,

e.g., ○ its complexity, ○ lack of documentation; ○

lack of suitable supporting software; ○ overambitious

project aims.

A seemingly disproportionate amount of project time can be taken up in dealing with such problems. The Implementation section gives you the opportunity to show where that time has gone.

Complete source code should be provided separately as an appendix.

This chapter includes the following subsections.

1. Brief description about the Tools/Scripts for Implementation
2. Module Hierarchy
3. Coding

CHAPTER 6

TESTING

This chapter includes the following subsections.

1. Test Plans
2. Unit Testing
    a. Test Items (Test Cases)
3. Integration Testing
4. System Testing
    a. Test Items (Test Cases)
5. Implementation - Changeover Plans

CONCLUSION

The purpose of this section is to provide a summary of the whole thesis or report. In this context, it is similar to the Abstract, except that the Abstract puts roughly equal weight on all report chapters, whereas the Conclusion chapter focuses primarily on the findings, conclusions and/or recommendations of the project.

There are a couple of rules for this chapter:

- All material presented in this chapter must have appeared already in the report; no new material can be introduced in this chapter (rigid rule of technical writing).
- Usually, you would not present any figures or tables in this chapter (rule of thumb).

Conclusions section can have the following (typical) content. These contents must not be given in bulleted format.

- Re-introduce the project and the need for the work though more briefly than in the introduction.
- Reiterate the purpose and specific objectives of your project.
- Recap the approach taken similar to the road map in the introduction.
  - However, in this case, you are re-capping the data, methodology and results as you go.
- Summarize the major findings and recommendations of your work.

Future Enhancements

Identify further works that can be added to make your system to meet the challenges of tomorrow. You can also include whatever requirements you could not fully due to the scarcity of time/resources.

BIBLIOGRAPHY

Ideas or contents taken from other sources should be properly cited. It is important that you give proper credit to all work that is not strictly your own, and that you do not violate copyright restrictions.

References should be listed in alphabetical order of authors' surname, and should give sufficient and accurate publication details. IEEE format is to be followed while preparing citations.

PUBLICATIONS OUT OF THE PROJECT WORK

A list of publications made or communicated out of the work done in the project is to be included here.

GENERAL INSTRUCTIONS

1. All chapters should contain an introduction and summary (summarizes the entire chapter content in one or two lines) sections.
2. Students have to take care that only chapters/sections relevant to their work are to be included in their report.
3. Instead of merely replicating the definitions for these sections from standard text books of Software Engineering, the student has to describe the information related to his/her work (For eg, Feasibility study should be about how the proposed work is technically/economically/operationally feasible).
4. Figures and tables are to be clear and legible.
5. Citations are to be provided wherever necessary.
6. Important code, screenshots, report formats and glossary of technical terms are to be attached as Appendices A, B, C and D respectively.

****